



İSKENDERUN TEKNİK
ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

**YÜKSEK
LİSANS
TEZİ**

**İNŞAAT DEMİRİ PAKETLERİNDE
GÖRÜNTÜ İŞLEME İLE
YÜKSEK DOĞRULUKTA
DEMİR ÇUBUK SAYIMI**

Nazlı Sümeyra DAĞILGAN

**ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ
ANABİLİM DALI**

HAZİRAN 2021



**İNŞAAT DEMİRİ PAKETLERİNDE GÖRÜNTÜ İŞLEME İLE YÜKSEK
DOĞRULUKTA DEMİR ÇUBUK SAYIMI**

Nazlı Sümeyra DAĞILGAN

**YÜKSEK LİSANS TEZİ
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI**

**İSKENDERUN TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

Haziran 2021

Nazlı Sümeyra DAĞILGAN tarafından hazırlanan “İNŞAAT DEMİRİ PAKETLERİNDE GÖRÜNTÜ İŞLEME İLE YÜKSEK DOĞRULUKTA DEMİR ÇUBUK SAYIMI” adlı tez çalışması aşağıdaki jüri tarafından OY BİRLİĞİ ile İskenderun Teknik Üniversitesi Elektrik-Elektronik Mühendisliği Anabilim Dalında YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Danışman: Dr. Öğr. Üyesi Murat FURAT

Elektrik Elektronik Mühendisliği Anabilim Dalı, İskenderun Teknik Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....
.....

Başkan: Doç. Dr. Alkan ALKAYA

Elektrik-Elektronik Mühendisliği Anabilim Dalı, Mersin Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....
.....

Üye: Dr. Öğr. Üyesi Ersin ÖZDEMİR

Elektrik-Elektronik Mühendisliği Anabilim Dalı, İskenderun Teknik Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

.....
.....

Tez Savunma Tarihi: 22/06/2021

Jüri tarafından kabul edilen bu tezin Yüksek Lisans Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

.....

Doç. Dr. Ersin BAHÇECİ

Lisansüstü Eğitim Enstitüsü Müdürü

ETİK BEYAN

İskenderun Teknik Üniversitesi Mühendislik ve Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez üzerinde Yükseköğretim Kurulu tarafından hiçbir değişiklik yapılamayacağı için tezin bilgisayar ekranında görüntülediğinde asıl nüsha ile aynı olması sorumluluğunun tarafıma ait olduğunu,
 - Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
 - Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
 - Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
 - Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
 - Bu tezde sunduğum çalışmanın özgün olduğunu,
- bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

İmza

Nazlı Sümeyra DAĞILGAN

22 / 06 / 2021

İNŞAAT DEMİRİ PAKETLERİNDE GÖRÜNTÜ İŞLEME İLE YÜKSEK
DOĞRULUKTA DEMİR ÇUBUK SAYIMI
(Yüksek Lisans Tezi)

Nazlı Sümeyra DAĞILGAN

İSKENDERUN TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

Haziran 2021

ÖZET

Demir çelik sektöründe demir çelik ürünlerinin kalite kontrol süreci son derece önemlidir. Döküm sürecinden paketlemeye kadar üretimin her aşamasında çok büyük miktarlarda enerji tüketilir. Demir çelik fabrikaları çevresel ortamın çok tozlu olduğu ve yüksek sıcaklık şartlarının bulunduğu ortamlardır. Bu çalışmada, depolama alanında bulunan çelik inşaat demiri sayımı ele alınmıştır. Demir çelik fabrikalarında çelik inşaat demiri paketinin yeniden sayılması gerektiğinde bu iş için personel görevlendirilir. Yüksek sıcaklık şartlarında insan gözüyle demir çubuk sayımı zorlu bir süreçtir. Bir diğer demir çubuk sayma alternatifi ise çelik inşaat demiri paketinin tartılmak üzere vinç sistemi ile taşınmasıdır. Her ikisi de zaman ve enerji tüketen yöntemlerdir. Bu çalışmada, inşaat demiri paketlerinin akıllı telefon kamerasıyla görüntüleri çekilerek demir çubuk sayımı için algoritma geliştirilmiştir. Boyama eksikliği, uygun olmayan çevresel ortam, bozuk kesim uçları, çelik nervür özellikleri, depolama alanının yetersiz aydınlatılması gibi unsurlara rağmen, geliştirilen algoritma yüksek doğrulukta sayım yapabilmektedir.

Anahtar Kelimeler : Görüntü işleme, demir çelik endüstrisi, demir çubuk sayımı.

Sayfa Adedi : 63

Danışman : Dr. Öğr. Üyesi Murat FURAT

HIGH ACCURATE REBAR COUNTING WITH IMAGE PROCESSING IN REBAR PACKAGES

(M. Sc. Thesis)

Nazlı Sümeyra DAĞILGAN

ISKENDERUN TECHNICAL UNIVERSITY
INSTITUTE OF GRADUATE STUDIES

June 2021

ABSTRACT

Iron and steel quality control is extremely important in the iron and steel industry. From casting process to bundling a large amount of energy are consumed at every stage of production. Iron and steel factories are environments where the surrounding environment is very dusty and where high temperature conditions are present. In this study, it has been considered the counting of steel rebar bundles in the storage area. In iron and steel factories, when it is necessary to recount the rebar in a bundle, staff are assigned for this work. In high temperature conditions, the counting of steel rebar bundles with human eyes is a challenging process. Another the counting of steel rebar bundles alternative is transported the steel rebar package with a crane system for weighting. Both of them are time and energy-consuming methods. In this study, an algorithm for counting of steel rebar bundles is developed by taking photograph of rebar packages with a smartphone camera. Despite the existence defects such as lack of painting, unsuitable environmental environment, distorted cutting edges, steel rib features, insufficient lighting of the storage area, the developed algorithm can count with high accuracy.

Key Words : Image processing, iron and steel industry, counting of steel rebar.

Page Number : 63

Supervisor : Assist. Prof. Dr. Murat FURAT

TEŐEKKÜR

Tez alıőmamın planlanmasında, araştırılmasında, yürütülmesinde ve oluşturulmasında ilgi ve desteęini esirgemeyen, engin bilgi ve tecrübelerinden yararlandığım, yönlendirme ve bilgilendirmeleriyle alıőmamı bilimsel temeller ışığında őekillendiren danıőmanım Dr. Öğr. Üyesi Murat FURAT'a, yüksek lisans öğretim süresince manevi desteęini eksik etmeyen aileme ve eşim Atılgan DAĞILGAN'a, sonsuz teşekkürlerimi sunarım. Bölgemizdeki inőaat demiri üretimi yapan firmalara, tez alıőmasında yer alan fotoęraflar ile sağladıkları katkı için ayrıca teşekkür ederim.



İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ.....	ix
ŞEKİLLERİN LİSTESİ	x
RESİMLERİN LİSTESİ	xi
SİMGELER VE KISALTMALAR.....	xiii
1. GİRİŞ.....	1
2. PROBLEM TANIMI.....	3
2.1. Çevre Ortamı	4
2.2. Depolama Alanının Yetersiz Aydınlatılması	5
2.3. Demir Demet Uçlarının Dairesel Olmayan Şekli.....	5
2.4. Önceki Çalışmalar	6
2.5. Tez Yapısı	12
3. MATERYAL VE YÖNTEM	14
3.1. Görüntü İşleme Yöntemleri.....	14
3.1.1. OpenCV	15
3.1.2. Görüntü işleme ve renk uzayları	15
3.1.3. RGB renk uzayı.....	17
3.1.4. HSV renk uzayı	17
3.1.5. OpenCV bileşenleri.....	18
3.1.6. OpenCV ile okuma, yazma ve gösterme.....	19

	Sayfa
3.1.7. Morfolojik işlemler	19
3.1.8. Filtreler	22
4. DENEYSEL ÇALIŞMALAR	33
4.1. Önerilen Algoritma	33
4.2. İlgi Bölgesi (ROI)	35
4.3. Eşik Değeri Belirleme	36
4.4. Hough Daire Dönüşümü ve Sayma Algoritması	37
4.5. Algoritma Sonuçları	39
4.5.1. 8mm ² 'lik inşaat demiri sayım sonuçları	40
4.5.2. 10 mm ² 'lik inşaat demiri sayım sonuçları	42
4.5.3. 12 mm ² 'lik inşaat demiri sayım sonuçları	44
4.5.4. 14 mm ² 'lik inşaat demiri sayım sonuçları	46
4.5.5. 16 mm ² 'lik inşaat demiri sayım sonuçları	48
4.5.6. 22 mm ² 'lik inşaat demiri sayım sonuçları	50
4.5.7. 25 mm ² 'lik inşaat demiri sayım sonuçları	52
4.6. Arayüz Tasarımı	54
5. DEĞERLENDİRME VE SONUÇ	56
KAYNAKLAR	59
DİZİN	62

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 4.1. 8 mm ² 'lik inşaat demiri için algoritma değerleri.....	42
Çizelge 4.2. Algoritma sayım sonucu (8 mm ²).....	42
Çizelge 4.3. 10 mm ² 'lik inşaat demiri için algoritma değerleri.....	42
Çizelge 4.4. Algoritma sayım sonucu (10 mm ²).....	42
Çizelge 4.5. 12 mm ² 'lik inşaat demiri için algoritma değerleri.....	44
Çizelge 4.6. Algoritma sayım sonucu (12 mm ²).....	44
Çizelge 4.7. 14 mm ² 'lik inşaat demiri için algoritma değerleri.....	46
Çizelge 4.8. Algoritma sayım sonucu (14 mm ²).....	46
Çizelge 4.9. 16 mm ² 'lik inşaat demiri için algoritma değerleri.....	48
Çizelge 4.10. Algoritma sayım sonucu (16 mm ²).....	48
Çizelge 4.11. 22 mm ² 'lik inşaat demiri için algoritma değerleri.....	50
Çizelge 4.12. Algoritma sayım sonucu (22 mm ²).....	50
Çizelge 4.13. 25 mm ² 'lik inşaat demiri için algoritma değerleri.....	52
Çizelge 4.14. Algoritma sayım sonucu (25 mm ²).....	52
Çizelge 5.1. Algoritma sayım sonuçları.....	56
Çizelge 5.2. Karşılaştırmalı hesaplama zamanı sonuçları	57

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 3.1. OpenCV resim okuma ve gösterme işlevi	19
Şekil 3.2. OpenCV resim yazma işlevi	19
Şekil 3.3. 3×3 çekirdek (kernel) matris.....	22
Şekil 3.4. Ortalama filtresinde sıklıkla kullanılan 3x3 matris	23
Şekil 3.5. Gaussian matris modeli.....	25
Şekil 3.6. a) Standart sapma $\sigma=1$ için grafik b) Standart sapma $\sigma=2$ için grafik.....	25
Şekil 3.7. Sobel çekirdek matrisi	28
Şekil 4.1. Önerilen algoritma	34
Şekil 4.2. İlgi bölgesi (ROI) formülü.....	35
Şekil 4.3. Hough daire dönüşümü.....	38
Şekil 4.4. Tasarımı yapılan arayüz.....	54
Şekil 4.5. Demir kesiti ve parametreleri yazılmış arayüz	55
Şekil 4.6. Arayüzün sonuç görünümü.....	55

RESİMLERİN LİSTESİ

Resim	Sayfa
Resim 1.1. Nervür izi veren çelik ringin aşınması.....	1
Resim 1.2. Slab üretiminde çatlak tespiti.....	2
Resim 2.1. Çelik üretim aşaması-sıvı çeliğin soğutulup kütük demir elde edilmesi	4
Resim 2.2. Depolama alanından demir demeti	5
Resim 2.3. 12mm ² kesitli inşaat demiri demeti.....	6
Resim 2.4. 8mm ² kesitli inşaat demiri demeti.....	6
Resim 2.5. Gerçek zamanlı üretim aşamasında lazer ile 3B görüntü alma.....	7
Resim 2.6. Önerilen ekipman düzeni	8
Resim 2.7. Çalışma sonucu	10
Resim 3.1. Görüntüde piksel kavramı.....	16
Resim 3.2. RGB renk uzayı	17
Resim 3.3. HSV renk uzayı.....	18
Resim 3.4. Aşındırma işlemi sonucu	20
Resim 3.5. Yayma işlemi sonucu.....	21
Resim 3.6. Ortalama filtre sonucu	23
Resim 3.7. Orta değer filtresi	24
Resim 3.8. Orijinal görüntü (üst), bulanıklaştırma filtresi uygulaması (alt).....	27
Resim 3.9. Sobel filtresi uygulaması	28
Resim 3.10. Laplasyan filtresi uygulaması	30
Resim 3.11. a) Orijinal görüntü, b) T1=50; T2=60 için Canny kenar bulma filtresi sonucu, c) T1=200 ve T2=230 için Canny kenar bulma filtresi sonucu.....	32
Resim 4.1. a) Orijinal görüntü, b) İlgi bölgesi (ROI) çıkarılmış görüntü	36
Resim 4.2. Nervürsüz demir çubuklar (solda, 57 adet), sayım sonucu (hatasız, sağda).	40

Resim	Sayfa
Resim 4.3. a) (8 mm ²) Orijinal görüntü, b) ROI uygulanmış görüntü, c) Canny kenar bulma filtresi uygulanmış görüntü, d) Sayım sonucu	41
Resim 4.4. a) (10 mm ²) Orijinal görüntü, b) ROI uygulanmış görüntü, c) Canny kenar bulma filtresi ile görüntü sonucu, d) Sayım sonucu.....	43
Resim 4.5. a) (12 mm ²) Orijinal görüntü, b) ROI uygulanmış görüntü, c) Canny kenar bulma filtresi ile görüntü sonucu, d) Sayım sonucu.....	45
Resim 4.6. a) (14 mm ²) Orijinal görüntü, b) ROI uygulanmış görüntü, c) Canny kenar bulma filtresi ile görüntü sonucu, d) Sayım sonucu.....	47
Resim 4.7. a) (16 mm ²) Orijinal görüntü, b) ROI uygulanmış görüntü, c) Canny kenar bulma filtresi ile görüntü sonucu, d) Sayım sonucu.....	49
Resim 4.8. a) (22 mm ²) Orijinal görüntü, b) ROI uygulanmış görüntü, c) Canny kenar bulma filtresi ile görüntü sonucu, d) Sayım sonucu.....	51
Resim 4.9. a) (25 mm ²) Orijinal görüntü, b) ROI uygulanmış görüntü, c) Canny kenar bulma filtresi ile görüntü sonucu, d) Sayım sonucu.....	53

SİMGELER VE KISALTMALAR

Bu çalışmada kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Kısaltmalar	Açıklamalar
3B	3 Boyutlu
I/O	Input/Output (Giriş/Çıkış)
CCD	Charged Coupling Devices (Şarjlı Bağlantı Cihazları)
RGB	Red, Green, Blue (Kırmızı, Yeşil, Mavi)
HSV	Hue, Saturation, Value (Renk özü, Doygunluk, Parlaklık)
ROI	Region of Interest (İlgi Bölgesi)
GUI	Graphical User Interface (Grafiksel Kullanıcı Arayüzü)

1. GİRİŞ

Günümüzde sıkça duyduğumuz görüntü işleme kavramı, endüstriyel üretimin her aşaması için önemli bir unsur haline geldi. Görüntü işlemenin temel amaçlarından biri anlamlı görüntü elde etmek veya görüntüyü dijital ortama dönüştürerek uygulanabilir bilgileri çıkarmaktır. Bu özelliğinden dolayı görüntü işleme yöntemi ile kalite kontrol denetimi; demir çelik ürünleri, gıda, sağlık gibi endüstriyel uygulamalarda çok önemli bir prosedürdür. Ürün kalitesini takip etmek için önemli yöntemlerden biri, ürünü görsel olarak gözlemlemektir. Bu anlamda görüntü işleme yöntemleri demir çelik sektöründe önemli rol oynamaktadır. Mevcut uygulamalarda üretim hatları insan tarafından takip edilemeyecek kadar hızlı olduğundan insan odaklı kalite kontrol süreçleri verimsiz olmaktadır. Üretim hatları şu anda birçok otomatik kontrol sistemi tarafından desteklense de kalite kontrol süreci hala bazı kalite kontrol personeli tarafından gerçekleştirilmektedir.

Çalışma ortamındaki sıcaklık koşullarının normal işletmelere göre yüksek olmasından kaynaklı insan gözüyle demir çubuk sayma işlemi zor ve emek gerektiren bir işlemdir. İnsan gözü, çok karmaşık yapıları analiz etmek için en uygun yöntemdir ancak yüksek sıcaklık ve hız gerektiren ortamlarda yetersiz kalmaktadır. Demir-çelik sektöründe görüntü işleme metotları, endüstriyel uygulamalarda insan gözünden daha elverişli olarak kullanılmaktadır. Bu sektörde görüntü işleme ile ilgili birçok çalışma vardır: Nervür izi veren çelik ringin aşınması veya çatlaması (Resim 1.1), slab üretiminde çatlak tespiti (Resim 1.2), nervür kalitesinin düşüklüğü (şeklinin bozukluğu veya yamukluğu), iş sağlığı ve güvenliği açısından görüntü işleme ile ürün takibi, üretim aşamasında paketleme öncesi demir çubuk sayımı, depolama sahasında tekrar demir çubuk sayımı, demir çubuğun çapının ölçülmesi gibi birçok çalışma yapılabilir.



Resim 1.1. Nervür izi veren çelik ringin aşınması



Resim 1.2. Slab üretiminde çatlak tespiti

Demir-çelik sanayisi diğer sanayi sektörleri için yarı mamul malzemeler ürettiğinden, metalin geri kalan işleme için ürünlerin kalitesi çok önemlidir. Örneğin, demir-çelik sektöründe demir çubukların yanlış sayımı ve paketlenmesi, hatalı ürünlerin yeniden işlenmesi için büyük miktarda enerji ve zaman harcanmaktadır. Otomatik üretim hatları esnek üretim sağlar, böylece üretim hatlarının yetersiz kullanımından kaynaklanan maliyetler önemli ölçüde azaltılabilir. Demir-çelik sektöründe önemli ürünlerden biri de binalarda kullanılan çelik inşaat demiridir. İnşaat demirinin paketleme aşamasında görüntü işleme yöntemlerinden destek alınabilir.

2. PROBLEM TANIMI

Demir-çelik sektörü ağır sanayi koşullarına sahip olmasından dolayı bazı işlemler insanlar açısından zorlayıcı olabilmektedir. Çalışma ortamında yüksek sıcaklık koşullarının olması, aşırı toz, aşırı gürültü gibi insanları fiziksel ve psikolojik olarak etkileyen olumsuz koşullar içermektedir. Demir-çelik sanayi diğer sanayi sektörleri için yarı mamul malzeme ürettiği için, metalin geri kalan işlemleri için ürünlerin bu aşamadaki kalitesi çok önemlidir [1].

Mevcut uygulamalarda, üretim hatları insan tarafından takip edilemeyecek kadar hızlı olduğundan genellikle verimsiz olan birçok insan temelli kalite kontrol süreci vardır. Üretim hatları şu anda birçok otomatik kontrol sistemi ile desteklense de kalite kontrol sürecinin bazı aşamaları kalite kontrol personeli tarafından yürütülmektedir. Bu gibi durumlarda makine görüşü insan incelemesinin yerini alabilir. Üretimin son aşaması olan paketleme işleminde ise istenilen sayıda demir çubuklar paketlenir. Ağır koşullar altındaki üretimin takibi hem zahmetli hem de insandan kaynaklı hatalar içermektedir. Hatalı ürünlerin yeniden işlenmesi için büyük miktarda enerji ve zaman harcanır. İnsan gözünün ayırt etmekte zorlandığı ve insan için fiziksel koşulların uygun olmadığı durumlarda görüntü işleme ile ilgili çözümler kullanılabilir. Bunlar arasında tez konusu olarak seçilen inşaatlarda kullanılan demir çubukların depolarda demir çubuk paketleri üzerinde sayımı önemli bir problemdir.

Doğası gereği çelik ürünlerin kalite kontrolü zordur. Çelik inşaat demiri demeti gibi paketleme aşamasına gelince, otomatik sayma sistemi demetlemenin yapılmasında önemli rol oynar. Geleneksel olarak, inşaat demirini saymanın iki yolu vardır. Biri, paketlemeden hemen önce bir personel tarafından gerçekleştirilir. Diğeri ise inşaat demiri paketinin ağırlığının ölçülmesi ve ardından sonucun bir inşaat demiri birim ağırlığına bölünmesidir. Şu anda teknolojik gelişmeler görüntü işleminin görüntü tanıma tekniklerini kullanarak otomatik olarak saymaya olanak sağlamaktadır [2].

Sürekli çalışan bir fabrikada inşaat demiri üretimi sırasında, bazen depolama alanında bulunan demetlerdeki inşaat demirinin yeniden sayılması gerekir. Bu durumda, inşaat demiri sayısını belirlemenin iki olası çözümü vardır. Biri geleneksel durumla aynıdır: demir demetindeki çubukların teker teker sayımı için bir personel atamaktır. Sayılacak inşaat demiri miktarına bağlı olarak işgücü harcanır. Diğeri bir çözüm ise, demeti tekrar

tartmak için bir vinçle kantara taşımaktır. Paket çok büyük olduğundan ağırlığını yeniden ölçmek için harcanan enerjinin maliyeti büyük olmaktadır. Bu nedenle inşaat demiri demetinin yeniden sayılmasında üretim maliyetini arttıran olumsuz yöntemler kullanılmaktadır [2].

Bu çalışmada, depolama alanında demet halinde bulunan inşaat demirinin görüntü işleme yöntemleri ile sayımı hedef alınmıştır. Ağır ve uzun bir ürün olduğu için, bina inşaatlarında kullanımda demetlerdeki inşaat demiri sayısı çok önemlidir. Çalışmanın zorlukları arasında üretim sahasında görüntüyü etkileyen olumsuzluklar ve çubukların kesimindeki düzensizlikler sayılabilir. Yukarıda sayılan şartlar altında yüksek doğrulukla depolardaki demir çubuk paketlerinin sayımının gerçekleştirilmesi önem arz etmektedir.

2.1. Çevre Ortamı

Demir çelik fabrikaları gürültü ve kirliliğin yoğun olarak bulunduğu yerlerdir (Resim 2.1). Bu nedenle üretim zor koşullar altında gerçekleşmektedir. Çalışma ortamının sıcak olması, yoğun tozun olması personelin doğru sayım yapabilmesini güçleştirmektedir.



Resim 2.1. Çelik üretim aşaması-sıvı çeliğin soğutulup kütük demir elde edilmesi

2.2. Depolama Alanının Yetersiz Aydınlatılması

Demir demetlerinin depolandığı kısmın yeterli ışıklandırılmaması sayma işleminin düzgün yapılması için bir diğer engeldir (Resim 2.2). Kamera kullanılarak uygun dijital görüntü elde etmek için çelik demetin uçlarının aydınlatılması gerekir. Ancak gerçekte çoğu üretim tesisinin inşaat demiri depolama alanı için bu mümkün değildir.

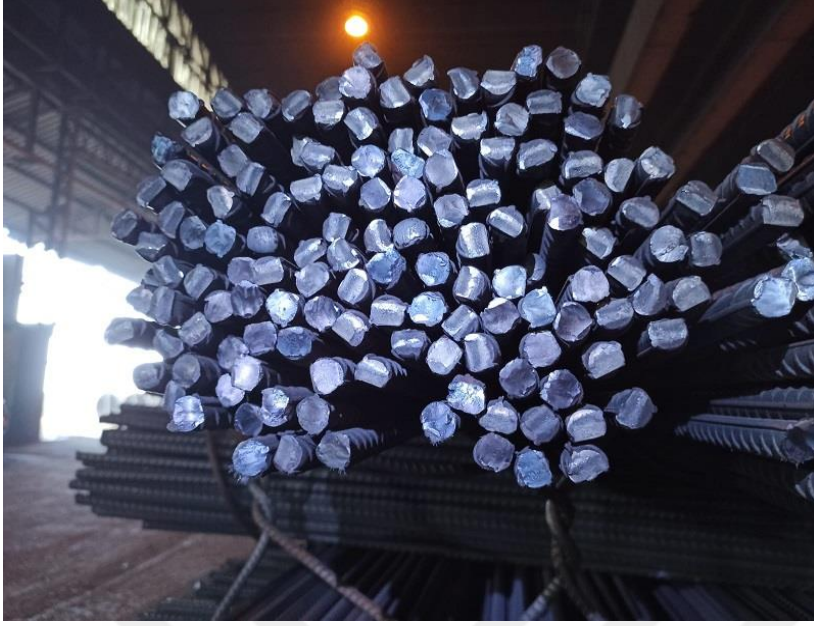


Resim 2.2. Depolama alanından demir demeti

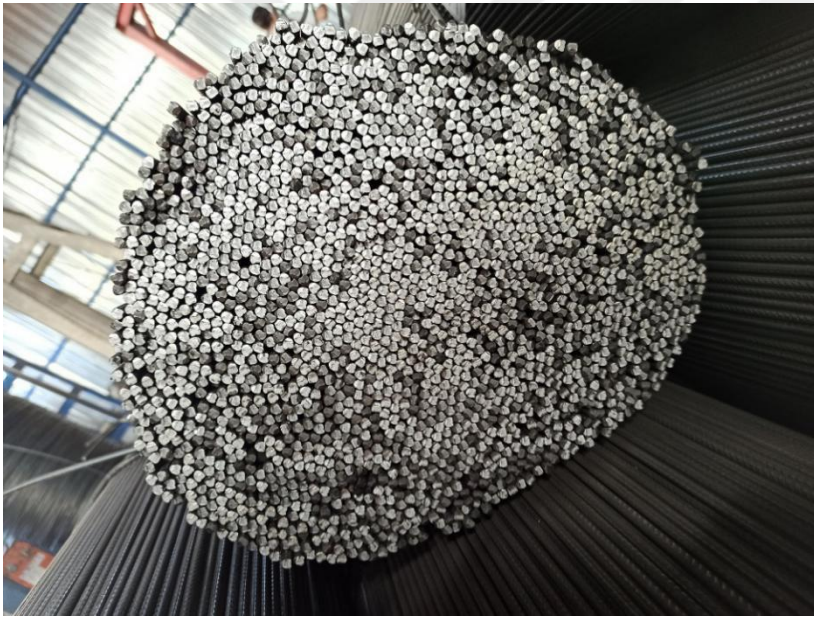
2.3. Demir Demet Uçlarının Dairesel Olmayan Şekli

Çelik inşaat demiri üretimi aşamasında demir çubuk yüzeyinde nervür adı verilen şekiller oluşturulur. Bu nervürler demir çubuğun dairesel yapısını değiştirmektedir. Ayrıca inşaat demirinin uçları kesilirken uçlarında da bozulma meydana gelir. Çelik nervür özellikleri, bozuk kesim ve boyama eksikliği gibi kusurlar, çelik inşaat demir demetinde sayım yapılmasını olumsuz etkileyebilir.

Bu tez çalışmasında tüm bu etmenler ile birlikte demir çubuk sayımı gerçek zamanlı bir fabrikadan alınan görüntüler ile görüntü işleme ile demir çubuğun sayımı gerçekleştirilmiştir. Görüntü işleme yöntemleri sayesinde çelik inşaat demiri sayımı için fazla zaman ve enerji kaybına gerek kalmadan önerilen algoritma ile yapılabileceği anlatılacaktır. Resim 2.3 ve Resim 2.4'te gösterilen resimler demir çubuk demetinin sayımı için birer örnektir.



Resim 2.3. 12mm² kesitli inşaat demiri demeti

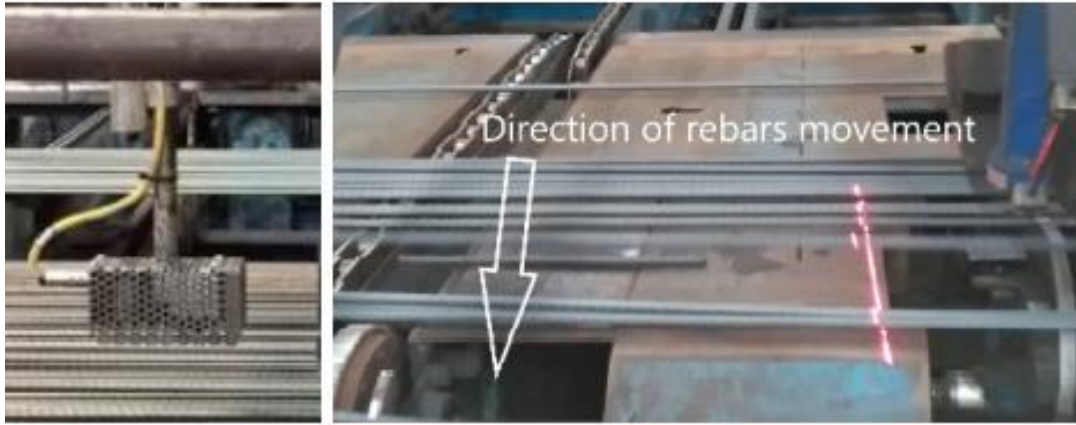


Resim 2.4. 8mm² kesitli inşaat demiri demeti

2.4. Önceki Çalışmalar

Görüntü işleme ile yapılan birçok endüstriyel uygulama mevcuttur. Görüntüyü istenilen amaca uygun ön işlemlerden geçirerek oluşturulan algoritmalar ile kalite kontrol süreci otomatik olarak gerçekleştirilebilir. Demir çubuk sayımı nesnenin doğru sayımı kapsamında oluşturulan görüntü işleme metotları yardımıyla oluşturulmuş bir çalışmadır. Son dönemde bu tür çalışmalar çok yaygınlaşmıştır. Benzer çalışmalara yönelik örnekler verilebilir:

Fernández ve diğerleri (2019) sıcak dönen hat üzerindeki inşaat demirlerinin sayımı için üç boyutlu (3B) görüntüleme yöntemi uygulanır. Bu çalışma, gerçek zamanlı bir ortamda yapılarak sürekli çalışan bir sistem elde etmek amaçlanmıştır. Gerçek üretim aşamasında çeşitli ışık koşulları ile birlikte farklı kesitteki (genellikle 12 mm^2 üstündekiler) inşaat demirlerinin sayımı yapılmıştır. 3B otomatik görüntüleme işleminde sinyalin lazer üçgenleştirilmesine dayanan fotoelektrik sensör teknolojisi kullanılmıştır (Resim 2.5). Sinyallerin sayısının sayılması mantığı ile yaklaşık bir çubuk sayımı gözlenir.



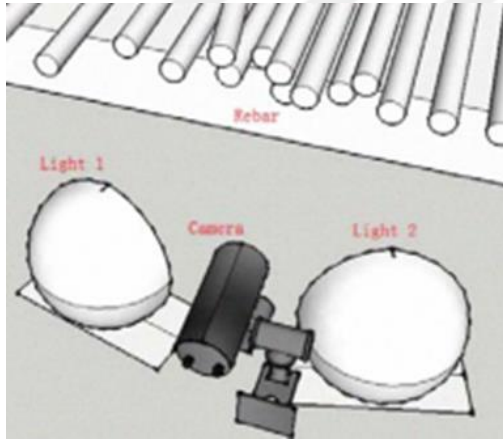
Resim 2.5. Gerçek zamanlı üretim aşamasında lazer ile 3B görüntü alma

Makine görüşünün net ve düzgün görüntü elde edebilmesi için öncelikle donanımsal düzenlemeler ile birlikte lazer ve kamera açısı ayarlanmıştır. 1024×1024 çözünürlüklü 3B kamera, 35 mW gücünde çizgi lazer üretici ve endüstriyel bilgisayar kullanılmıştır. Daha sonra sayma işlemini gerçekleştirmek amacıyla gerekli sayma algoritması oluşturulmuştur. Sonuçlar çizelge şeklinde sunulmuştur. Çizelgede 8 mm^2 , 10 mm^2 ve 12 mm^2 kesitlerindeki inşaat demirlerinin hata oranları verilmiştir. Hata oranları, lazer sistemli yeni kamera ve sayma algoritması ile %0,3'ün altında bulunmuştur [5].

Yang ve diğerleri (2015) yaptıkları çalışmada, demir çubukların görüntülerinden birbirine yapışan demir çubukların kenar konturunu elde etmek için bağlantılı alan analizi kullanılmıştır. Ardından, konturdaki içbükey alanlar taranıp ve içbükey noktalar bulunmuştur. Daha sonra K-seviyesi hata toleransı algoritması aracılığıyla tek çubuğu ayırmak ve saymak için içbükey nokta eşleştirme koşulu kullanılmıştır. Önerilen yöntem, yapışkan demir çubukların segmentasyonu için yüksek bir doğruluğa sahip olduğu ve demir çubukları doğru şekilde bölebildiğini göstermiştir [3].

Nie ve diğerkleri (2015) video analizine dayalı olarak inşaat demirini otomatik sayan bir algoritma sunulmuştur. Endüstriyel bilgisayar, I/O iletişim, bir kamera ve led aydınlatıcı kullanılmıştır. Üretim bandının oldukça hızlı hareket etmesi, demir çubukların üretim bandı üzerinde ileri veya geri hareket etmesine izin verir. Önerilen algoritmada ilk olarak video karesini üç bölüme ayırmak için yöntem geliştirilmiştir. Bitişik çerçeveler arasında tekrarlanan demir çubuk sırasını tanımlamak için bir sıra eşleştirme algoritması geliştirilmiştir ve her çerçevede yeni gelen demir çubuk sayısı hesaplanabileceği anlatılmıştır. Deneysel sonuçlar, önerilen yöntemin yüksek hızlı ve hassas inşaat demiri sayımını destekleyeceği belirtilmiştir [4].

Su ve diğerkleri (2010) demir çubuğun kesitini yakalamak için maksimal teğet daire algoritmasının kullanılması önerilmiştir. Kesit ile demir çubuğu tespit edebilmek için gradyan hough daire dönüşümünde değişiklik yapılmış ve bir dizi karşılık gelen operatör kullanılmıştır. Başarısızlık oranının son derece düşük olduğu (yüzde 0,01) ispat edilerek hızlı bir sayım yapabilen algoritma önerilmiştir [6]. Bir adet endüstriyel kamera ve iki adet ışık kaynağı kullanılmıştır (Resim 2.6).



Resim 2.6. Önerilen ekipman düzeni

Xiaohu ve Jineng (2018) otsu yöntemi (gri seviye histogramı yapılmadan) kullanılarak eşik değeri belirleme yöntemine dayanarak demir çubukların sayımı için görüntü konturlarına dayalı bir algoritma önerilmiştir. Otsu yönteminden sonra, demir çubuk görüntüsü üzerine kontur konumlandırması uygulanmıştır. Otsu metodundan ve eşik değeri belirleme yöntemine göre ikileştirme (binary) yapılan görüntü, konturları çıkarılmış ve tekli çelik çubuk konturları veya çoklu çelik çubuk konturları diye sınıflandırılmıştır. Farklı konturlarla işaretleme yapılmıştır. Donanımsal olarak kamera ve ışık kaynağı

kullanılmıştır. Endüstriyel açıdan pratik bir uygulama olabileceği vurgulanmıştır [7].

Ablidas ve diğerleri (2019) K-mean algoritması ve hough daire dönüşümü kullanılarak bir algoritma ile sayma yöntemi için alternatif bir yol önerilmiştir. Görüntü çekimleri karanlık bir ortamda gerçekleştirilerek görüntünün kontrastı artırılıp gürültüyü azaltmak için filtre kullanılmıştır. Önerilen bu yöntem ile bir pakette beklenen gerçek demir çubuk sayım sonuçları ile sistemin otomatik sayım sonuçları arasında önemli bir fark olmadığı belirtilmiştir. Endüstride, personel sayım süresi ile sistemin otomatik sayım süresinde ciddi fark olduğu ve bu sayede zamandan kazanç sağlandığı vurgulanmıştır [8].

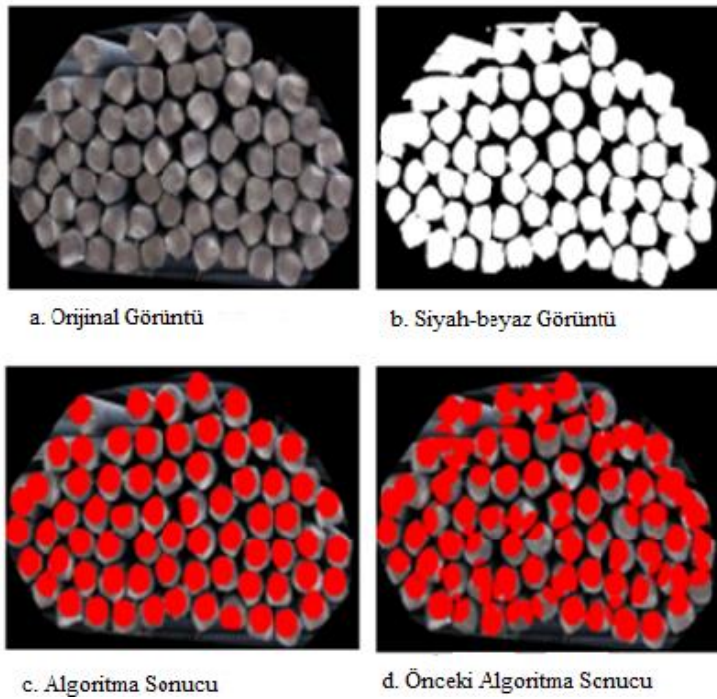
Park ve diğerleri (2020) bu çalışmada derin öğrenmeyle nesne algılama teknolojisini kullanarak denetlemek amaçlı bir yol önerilmiştir. Önerilen çalışma iş gücünden, zamandan ve paradan tasarruf sağlayan bina denetimini kontrol etmeye yarayan bir yöntemdir. Matlab aracılığıyla, YOLOv2 network ağının öğrenilmesini kullanarak bir çember çubuğu algılama ağı oluşturulmuştur. Bu ağ, benzer görüntüler için %92,85 ve belirli mesafedeki demir çubuk demeti için %90 veya daha fazla doğrulukta tespit yapabilmektedir [9].

Xinman ve diğerleri (2017) kare demir slabların sayımı için önerilen yöntemde videodan resim okuma ile toplanan görüntüler önce ortalama filtre ve eşik değeri segmentasyonu uygulanmıştır. Daha sonra canny kenar bulma metodu ile hough dönüşümü kullanılmıştır. Demir çubukların sayısı işaretlenmiş ve görüntülenmiştir. %100 doğrulukla sayım elde edilmiştir [10].

Ying ve diğerleri (2010) demir çubukların görüntüsünün otomatik olarak bölümlendirilmesi için bir yöntem sunulmuştur. Demir çubuğun yaklaşık kesiti zincir kodlamasıyla ölçülmüş. Demir çubukların kenarları sobel operatörü filtresi tarafından tespit edilmiştir ve demir çubukları arka plandan ayırmak ve şeklini belirginleştirmek için otsu metodu kullanılmıştır. Daha sonra hough dönüşüm ve eşik değeri segmentasyonu bulunur. Ardından bir bölgedeki maximum gri değerler oluşturulur. Yaklaşık daire görüntüleri ile doğru şekilde demir sayımının gerçekleştirildiği vurgulanmıştır [11].

Hou ve diğerleri (2011) görüntü işleme uygulamaları ile demir çubuk demet sayımı için şablon kaplama tabanlı yeni bir algoritma önerilmiştir. Öncelikle ikili (binary) görüntü elde

edilir. Yatay şablonla geniş alan bulunan hedef bölge araştırılır ve ardından çapraz şablonla demir çubuğun merkezi aranır ve son olarak sekizgen şablonla görüntü geçici olarak kaplanır. Demir çubuğun yerini en iyi şekilde tespit edebilmek için Freeman zincir kodlarına dayalı sınır takip algoritması ile dairesel bölge tespit edilir. Böylece her demir çubuk doğru şekilde tanımlanabileceği ve sayısının belirlenebileceği anlatılmıştır (Resim 2.7). Ayrıca önerilen algoritmanın diğer algoritmalara göre daha yüksek bir sayma hassasiyeti ve kaplama doğruluğu olduğu öne sürülmüştür [12].



Resim 2.7. Çalışma sonucu

Zhao ve diğerleri (2016) Donanımsal açıdan CCD kamera, endüstriyel bilgisayar ve aydınlatma sistemi kullanılarak düzgün görüntü elde edebilmek için uygun ortam oluşturulmuştur. Öncelikle görüntüdeki gürültüyü azaltmak için ortalama ve orta değer filtreleri görüntü üzerinde deneme yapılarak kullanılmıştır. Eşik değeri segmentasyonuna dayalı otsu yöntemi uygulanmıştır. Eşik değeri, tekrarlamalı (iterative) eşik değeri, otsu gibi yöntemler verilerek eşik değerleri değiştirilerek karşılaştırma yapılmıştır. Otsu yönteminin en iyi sonucu verdiği kararlaştırılmıştır. %95 ve üzeri doğruluk oranında demir çubuk sayımı gerçekleştirilmektedir [13].

Yan ve Chen (2018) demir çubukların uçlarının yapışık olma durumuna göre tekli ve çoklu sınıflandırmaya dayanan hızlı ve doğru bir sayma yöntemi sunulmuştur. Demir çubukların

uçlarının yapışmasını engellemek için bölümlere ayrılmış, morfolojik ve bazı ön işlemlerden geçirilmiştir. Daha sonra, bölgenin alanı, çapı, ağırlık merkezi, şekil faktörü dahil olmak üzere işlenen ikili (binary) görüntünün özellikleri elde edilmiştir. Elde edilen görüntünün alan özelliklerine göre, hedefin tekli ya da çoklu olarak sınıflandırılması yapılmıştır. Ayrıca ağırlık merkezinin özelliklerine göre de demir çubuklardan çoklu demir çubuk şablonu oluşturulmuş ve yüksek verimlilik ile doğru sayım amacına ulaşmak için alan ve diğer faktörler birleştirilerek işlem yapılmıştır [14].

Zhang ve diğerleri (2008) bu çalışmada, çalışan bir demir çubuk sayma ve bilgisayar görüşüne dayalı otomatik ayırma sistemi önerilmiştir. Önerilen sistem, sayma alanı boyunca hareket halinde olan demir çubuk miktarını hesaplar ve ayırma makinesi de sonuca göre ayırma işlemini yapar. Şablon eşleştirme ve değişken eşik değeri segmentasyonu ile demir çubukları ayırt etme işlemini iyileştirmeye çalıştıkları vurgulanmıştır. Demir çubuk fabrikasında uzun süredir bu yöntemi kullandıkları belirtilmiş ve yanlış tespit oranı %0,01'in altına indirildiği ifade edilmiştir [15].

Pathan ve Harale (2016) ipekböceği yumurtalarının düzensiz dizilimleri ve üretim süreçlerinin zorluğundan dolayı doğru bir şekilde sayım yapmak oldukça güçtür. Görüntü işleme metotları yardımı ile hem insan gücünden hem de zaman tasarrufundan kâr elde edilebileceği anlatılmıştır. Bu sebeple otomatik sayım yapabilen bir sisteme ihtiyaç duyulmuştur. Önerilen yöntem, ipekböceği yumurtalarının bulunduğu tabakadan yumurta bulunmayan parçasını saymak ve bu kısmı ortadan kaldırmak amacıyla bir yumurta bölümü tespit etmek istenmiştir. Görüntüde oluşan gürültüleri temizleyebilmek için yumurta bulunan kısmı, eşikleme yöntemi ile tespit edebilmek için görüntü bölümlenme tekniği tercih edilmiştir. Nesne sayısını doğru hesaplayabilmek için matlab programında yer alan nesne sayma fonksiyonları kullanılmıştır [16].

Bhaskar ve Yong (2014) trafikte araçların takibini ve tespitini kolaylaştırmak için etkili bir yöntem önerilmiştir. Fakat yine de bu alanda yapılan çalışmalar için iyileştirmelerin gerekli olduğu vurgulanmıştır. Bu çalışmada, gaussian mix modeli ve blob algılama yöntemi ile bir algoritma kullanılmıştır. Arka plan ile ön plan birbirinden ayrılmıştır. Gürültünün azaltılması ve tespit edilmesi gereken nesnenin ayırt edilmesi için kolaylık sağlar. Hareketli nesneyi doğru şekilde algılayabilmek ve gürültüyü azaltmak için filtre kullanılmıştır. Daha sonra nesnelere takip edilerek sayım işlemi gerçekleştirilmiştir.

Gaussian mix model ve blob algılama yöntemi kullanılarak yüzde %91'in üzerinde başarı elde edilmiştir [17].

Han ve Wang (2019) yan yana dizilmiş paralel duran kâğıtlar için sayma işlemi yapılmıştır. Bu çalışma temelde genel bir kâğıt görüntüsü ile bu kâğıdın aslında kenar görüntüsünü saymaktadır. Programda hatalı ya da eksik sayım olmaması için dikdörtgen bir şablon kullanımı önerilmiştir. Görüntü öncelikle gri formata dönüştürülür daha sonra görüntüdeki gürültüyü azaltmak ve sayılacak nesneyi netleştirmek için Gabor filtresi uygulanmıştır. Kâğıdın ikili yan görüntüsü tespit edilir. Sayma algoritması ile sayım işlemi tamamlanır. Geleneksel yöntem ile önerilen yöntem karşılaştırılmış ve önerilen yöntemin daha doğru sayım ile sonuca ulaştığı belirtilmiştir [18].

Kaymak ve diğerleri (2019) görüntü işleme teknolojisi desteği ile bir elma bahçesindeki kırmızı renkli elmaların sayısının belirlenmesi amaçlanmıştır. Dijital fotoğraf makinesi ile görüntü elde edilmiş ve bu görüntü RGB renk uzayı formatında Matlab ile tasarlanmış yazılıma aktarılmıştır. Hedefteki renk kırmızı olduğu için kırmızı renk diğer renklerden ayrıştırılmak istenmiştir. Bu nedenle gri seviyeli görüntüden ayırt edilmiştir. Filtreleme işlemi uygulanıp görüntü siyah beyaz formata dönüştürülmüş ve daha net ayırım yapabilmek için eşik değeri belirlenmiştir. Nesnelerin merkez noktaları ve kenar çizgileri seçilmiştir. Merkez sayımı tamamlanan nesnelere elma olarak kabul edilmiştir. %78,47 oranında başarı elde edildiği belirtilmiştir [19].

Hernández-Ruiz ve diğerleri (2021) SA-CNN-DC (Ölçek uyarlamalı- Evrişimli sinir ağı- Mesafe kümeleme) isimli görüntüden otomatik demir çubuk sayımı yapabilen bir algoritma önerilmiştir. Evrişimli sinir ağı ve kümeleme yöntemleri ile iki adet makine öğrenme tekniği kullanılmıştır. Sistem hem dairesel hem de karesel demir çubukların sayımı için eğitilmiştir. 1091 adet demir paketi için program 70 saniye süre ile sayımı gerçekleştirmiştir. %98,81 ve %98,57 oranında başarı elde edilmiştir [20].

2.5. Tez Yapısı

Bu tez çalışması; giriş, problem tanımı ve önceki çalışmalar, materyal ve yöntem, deneysel çalışmalar, değerlendirme ve sonuç olmak üzere beş kısımdan oluşmaktadır. Giriş bölümünde görüntü işlemenin günümüzde hangi alanlarda kullanıldığından, öneminden ve avantajlarından bahsedilmiştir. Tez çalışmamızın demir çelik endüstrisindeki öneminden

ve endüstride yaşanan başka problemlerden bahsedilerek yapılması muhtemel olan görüntü işleme çalışmalarına da yer verilmiştir. Tezin motivasyonu ve amacı anlatılarak çalışma desteklenmiş ve tez çalışmamıza yönelik literatür çalışmaları anlatılmıştır. Giriş bölümünün son kısmında ise tez çalışmasının aşama aşama taslağından bahsedilmiştir.

Materyal ve yöntem kısmında OpenCV, görüntü işleme yöntemleri ve kullanılan programlar anlatılmıştır. Görüntü işlemenin temel adımları, renk uzayları, OpenCV bileşenleri, OpenCV'nin temel komutları, morfolojik işlemler ve filtreler anlatılmıştır. Bu kısımda tez çalışmamız ile ilgili tüm morfolojik işlemler ve filtreler denenmiştir. Ayrıca çalışmamızın temel materyal ve yöntemi belirlenmiştir.

Deneysel çalışmalar kısmında, inşaat demir çubukları sayımı için oluşturulan program detaylı şekilde anlatılmış ve algoritma önerilmiştir. İnşaat demir çubuklarının görüntüleri üzerinde önerilen algoritmanın yüksek doğrulukla sayım yaptığı çizelgelerde oluşturulan sayım sonuçları ile incelenmiştir. Ara yüz tasarımı ile bilgisayar uygulamasına yönelik çalışma yapılmıştır.

Son bölümde ise tez çalışmasından elde edilen bulgular tartışılmış ve gelecek çalışmalar için öneriler yapılmıştır.

3. MATERYAL VE YÖNTEM

Tezin bu aşamasında görüntü işleme yöntemleri, OpenCV bileşenleri ve temel komutlar incelenmiştir. Morfolojik işlemler ve görüntüdeki gürültüleri elimine etmek için kullanılan önemli filtreler anlatılmıştır. Bu kısımda deneysel çalışmamızın için görüntü işleme yöntemleri, komutları ve uygun filtreler denenmiştir. Materyal ve yöntemler kısmı, tez çalışmamız için temel oluşturmuştur.

Günümüzde artık akıllı cep telefonlarının ve fotoğraf çekme özelliğinin yaygın olarak kullanılması nedeniyle bu çalışmada çekilen fotoğraflar için bir cep telefonundan faydalanılmıştır: Cep telefonunun ve kamerasının özellikleri aşağıda verilmiştir:

- Xiaomi Redmi Note 8 Pro,
- Enstantane: 1/50 saniye,
- Diyafram açıklığı: f/1,89,
- Çözünürlük: 4624x3472 piksel
- Flash (ışık ayarı): kapalı

Görüntüler, aşağıdaki özelliklere sahip bilgisayarda hazırlanan yazılım ile işlenmiştir:

- İşlemci: Intel® Core™ i5-6200U CPU @2,30 GHz
- RAM: 12 GB
- Ekran kartı: NVIDIA GeForce 940 MX
- Ekran çözünürlüğü: 1366x768

3.1. Görüntü İşleme Yöntemleri

Görüntü işleme, görüntü veya videodan elde edilmiş sinyallerin, belirli işlemlerden geçirilerek amaca uygun olarak yapılandırılmasıdır. Temel amacı, piksel değerleri ile işlem yaparak görüntüyü işlevsel hale getirmektir. Görünmesi zor nesnelere gözlemlenme, gürültülü görüntüleri iyileştirme, bir görüntüdeki çeşitli nesnelere tanımlama ve ayırt etme gibi çeşitli amaçları mevcuttur. Görüntü işlemede en çok tercih edilen açık kaynak kodlu kütüphane olan OpenCV'dir. Bu bölümde OpenCV ve bileşenleri, kullanılan yazılım dilleri, renk uzayları, morfolojik işlemler ve filtrelerden bahsedilmiştir.

3.1.1. OpenCV

OpenCV; Intel tarafından 1999 yılında geliştirilen açık kaynak kodlu bir makine öğrenimi yazılımı yani görüntü işleme kütüphanesidir. Hesaplama verimliliğini artırmak amaçlı gerçek zamanlı uygulamalar için tasarlanmıştır [21]. Bilgisayarla görme uygulamaları için ortak bir altyapı sağlamak ve makine algısının kullanımını hızlandırmak için oluşturuldu. OpenCV, görüntü işleme ile ilgili yüzlerce temel ve ileri seviyedeki fonksiyonu, optimize edilmiş halleriyle barındırmaktadır [22].

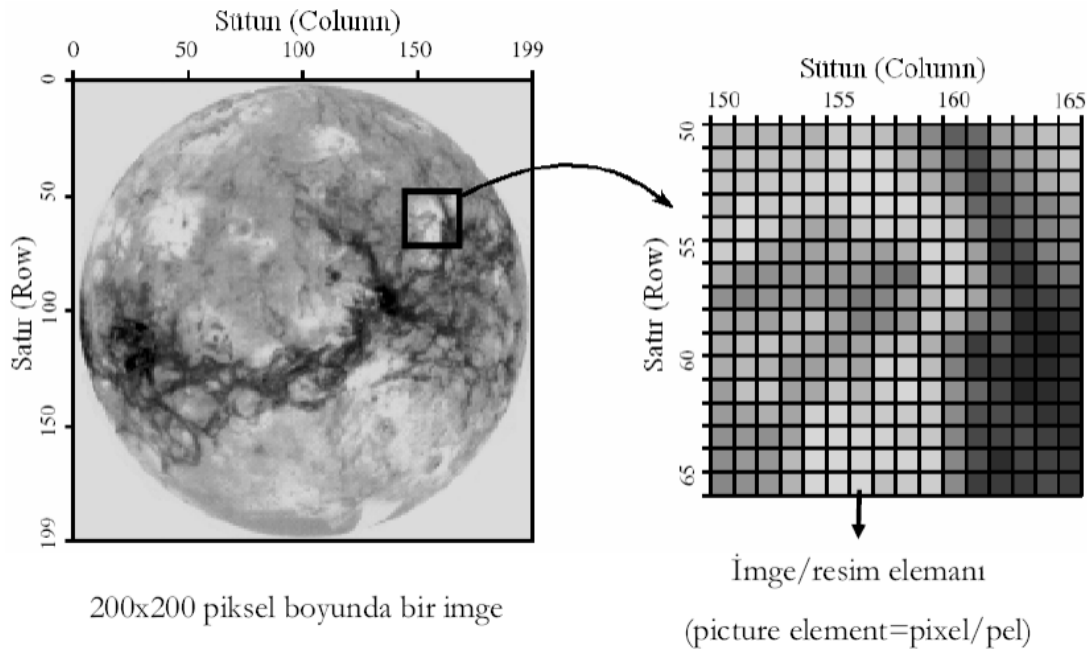
Kütüphanede hem klasik hem de son teknoloji bilgisayarla görme ve makine öğrenimi algoritmalarından oluşan kapsamlı bir set içeren 2500'ten fazla optimize edilmiş algoritma bulunur. Bu algoritmalar; yüzleri algılamak ve tanımak, nesnelere tanımlamak, videolarda insan hareketlerini sınıflandırmak, kamera hareketlerini izlemek, hareketli nesnelere izlemek, nesnelere 3B modellerini çıkarmak, stereo kameralardan 3B nokta bulutları oluşturmak, görüntüleri birleştirerek yüksek bir çözünürlük elde etmek için kullanılabilir. Kütüphane; şirketlerde, araştırma gruplarında ve devlet kurumlarında yaygın olarak kullanılmaktadır [21].

Başlangıçta C dili ile kodlanmaya başlanmış olmasına rağmen C++, Python, Java, Matlab/Octave, C# gibi pek çok dil ile beraber kullanılabilir. OpenCV programlama dillerinde olduğu gibi platform ve Windows, Linux, MacOS gibi farklı işletim sistemleri üzerinde çalışabilmektedir. Bu tez çalışmasında OpenCV açık kaynak kodlu kütüphanesi, Visual Studio Code ve Python dili kullanılmıştır. Visual Studio Code, Windows için entegre geliştirme ortamıdır. Kullanıcıya kod düzenleyici, hata ayıklayıcı ve GUI ara yüz tasarım penceresi gibi tasarım seçenekleri sunar.

3.1.2. Görüntü işleme ve renk uzayları

İnsan algısı çevremizdeki şekiller, renkler, hareketler gibi tüm görsel bilgiyi edinme, bütünleştirme, yorumlama yeteneğine sahiptir. Tüm bu görsel bilgiyi bilgisayarın görmesi görüntü (imge) ile sağlanır. Görüntü, çeşitli yollarla elde edilen bilgilerin görüntüsel olarak saklanmasına ve gösterimine olanak sağlayan resimlerdir. Görüntü işleme, bir disiplin olarak ele alındığında, amaç orijinal şeklinde olmayan bir görüntüyü görsel olarak güçlendirmek ve istatistiksel olarak değerlendirmektir. Bu amaç görüntü üzerinde etkili

olabilecek işlemlerin geliştirilmesi ve uygulanması ile yürütülür. Görüntü işleme üç ana başlıkta incelenebilir. Birincisi optik, ikincisi elektronik olan analog ve üçüncüsü dijital yöntemlerdir. Bu üç yöntemin her biri işlemin uygulanmasındaki en pratik yaklaşımı tanımlayan özel uygulama ile bulunurlar. Sayısal görüntü ise; fiziksel ortamdaki sahnenin bilgisayar uyumlu olarak resim şeklinde sunulmasıdır. Sayısal görüntü iki boyutlu bir fonksiyon $f(x,y)$ ile tanımlanabilir. $f(x,y)$ fonksiyonunun parametreleri olan x ve y konum (düzlem) koordinatlarıdır. Bu koordinatlar pikselin resim üzerindeki konum değeridir. $f(x,y)$ fonksiyonunun dönen sonucu da, resimde o pikseldeki yoğunluk veya gri düzeyidir (Resim 3.1) [24]. Tıp ve biyoloji (x-ışınları, biyomedikal görüntüler), coğrafi bilimler (hava ve uydu görüntülerinden hava tahmini), fizik (elektron mikroskobu görüntüleri), uzay bilimleri (uydu, mikrodalga radar görüntüleri), savunma sanayi (gece görüş, akıllı roket sistemleri), endüstriyel uygulamalar (süreç, ürün kalite kontrolü) gibi pek çok alanda görüntü işleme kullanılır.



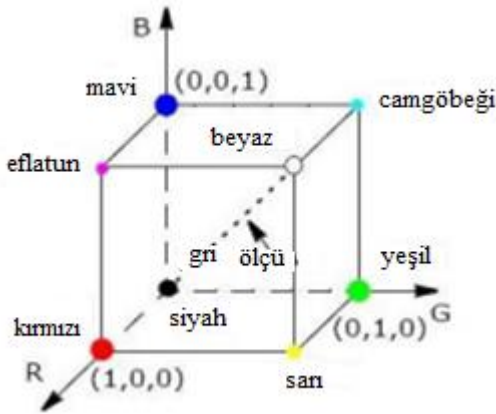
Resim 3.1. Görüntüde piksel kavramı

Görüntü işleme yazılımı belleğin doğru kullanımına bağlı olduğu için bellek modelleri önemlidir. Dikdörtgen şeklindeki görüntü modeli en popüler bellek modelidir. Gri seviyede bir görüntü, her biri 0 ve $2g - 1$ değerleri arasında bulunan gri değerleri arasında bulunan gri değerleri içeren kutular setinden meydana gelmiştir. Her bir kutu piksel ve her bir piksel dizisi M yatay sıralardan ve N düşey sütunlardan meydana gelmiştir.

Böylece her bir piksel dikdörtgensel dizinin elemanı olarak ifade edilebilir [25].

3.1.3. RGB renk uzayı

Görüntülerdeki renkleri ifade etmek, anlamlandırabilmek için renk uzaylarına ihtiyaç duyulur. Renk uzayları renkleri tanımlamak amaçlı kullanılan matematiksel modellerdir ve 3B olarak tasarlanırlar. Çünkü bir rengi belirlemek için birbirinden bağımsız üç değişik renk gereklidir. En yaygın bilinen renk uzayı RGB renk uzayıdır. RGB renk uzayı; Red (kırmızı) – Green (yeşil) – Blue (mavi) renk bileşenlerinden oluşmaktadır. RGB renk uzayında bulunan bu renklere ana renkler de denir. Bu renk uzayında doğadaki tüm renkler, RGB renk bileşenlerinin farklı oranda karışımı ile elde edilir. RGB renk bileşenleri her biri için 0 ile 255 arasında sayısal değerler alarak farklı değerlerde karıştırıldıklarında yeni renk ve tonlarını oluştururlar. RGB renklerinin aynı 255 değerinde karışımlarından beyaz elde edilmektedir. Tüm değerler 0 olduğunda ise siyah görüntü elde edilmektedir. RGB renk uzayındaki renk bileşenleri arasında güçlü bir korelasyon vardır ve görüntü üzerindeki işlemler (bulanıklaştırma vb.) farklı bileşenler arasındaki korelasyonu değiştirecektir [26]. Bu model, Kartezyen koordinat sistemine dayanmaktadır (Resim 3.2).

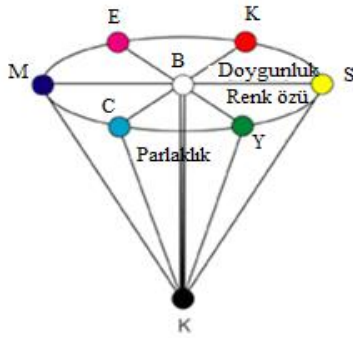


Resim 3.2. RGB renk uzayı

3.1.4. HSV renk uzayı

HSV (Hue – Saturation – Value) yani renk özü (ton), doygunluk ve parlaklık olarak adlandırılmıştır. RGB renk uzayında renklerin karışımı kullanılmasına karşın HSV de renk, doygunluk ve parlaklık değerleri kullanılır. Renk özü (ton) bir renk türüdür.

Doygunluk renk miktarını, parlaklık ise ışık miktarını temsil eder. Ana renkleri kırmızı 0 derece, yeşil ana renkler 120 derece ile 240 derece arasında başlayan renk türünü tanımlar. Mavi renk ise 360 dereceye dönecektir [27]. HSV uzayında siyah renk için renk ve doygunluk değerleri 0 ile 255 arasında herhangi bir değer alabilir olmasına rağmen parlaklık değeri 0'dır (Resim 3.3). Beyaz renkte ise parlaklık değeri 255'tir.



Resim 3.3. HSV renk uzayı

3.1.5. OpenCV bileşenleri

OpenCV kütüphanesi; CORE, HighGUI, IMGPROC, IMGCODES, VIDEOIO bileşenlerinden oluşur. Bunlar [21]:

CORE: Temel veri yapıları ve içeriğini, resim çizme gibi bölümleri içerir.

HighGUI: Video ve görüntülerin saklanması, resim görüntülüne ve GUI için gerekli metotlar barındırır.

IMGPROC: Nesne tespit etme, filtreleme işlemleri gibi fonksiyonları içerir.

IMGCODECS: Opencv'nin temel işlemlerinden olan resim/video okuma yazma kodlarını barındırır.

VIDEOIO: Donanımsal (kamera, video sistemleri) erişim için fonksiyonları içerir.

3.1.6. OpenCV ile okuma, yazma ve gösterme

Open CV ile bir görüntüyü bilgisayar formatında dijital ortama aktarabilmek yani görüntüyü okumak için imread fonksiyonu kullanılır. Açık kaynak kodlu kütüphane ile görüntünün her bir pikseli hafızaya kaydederek görüntüyü okuma işlemi sağlanmış olur. Okunacak görüntü üzerinde işlem yapabilmek için bir adet nesne tanımlanır. Nesne, OpenCV ile dosya sisteminden okunan görüntü dosyasını dijital olarak tutmaktadır. Okunmuş görüntüyü görmek için imshow komutu kullanılır. Aynı zamanda cv2.waitKey() eklenir. Bu komut bir klavye bağlama işlemidir. Milisaniye cinsinden zamanı içerir. Yani herhangi bir klavye işlevi için belirtilen milisaniye kadar bekler ve bu süre içerisinde herhangi bir tuşa basılırsa program devam eder. cv2.destroyAllWindows() komutu, programda oluşturduğumuz tüm pencereleri yok eder. Herhangi bir belirli pencereyi yok etmek için ise cv2.destroyWindow() komutu kullanılır (Şekil 3.1).

```
import numpy as np
import cv2

# Load an color image in grayscale
img = cv2.imread('resim1.jpg',0)

cv2.imshow('image',img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Şekil 3.1. OpenCV resim okuma ve gösterme işlevi

Görüntüyü okuduktan sonra kaydetmek istersek ise cv2.imwrite() komutu kullanılır (Şekil 3.2). Parantez içerisindeki ilk bağımsız değer dosya adı olup, ikinci bağımsız değer ise kaydetmek istediğimiz görüntüdür.

```
cv2.imwrite('messigray.png',img)
```

Şekil 3.2. OpenCV resim yazma işlevi

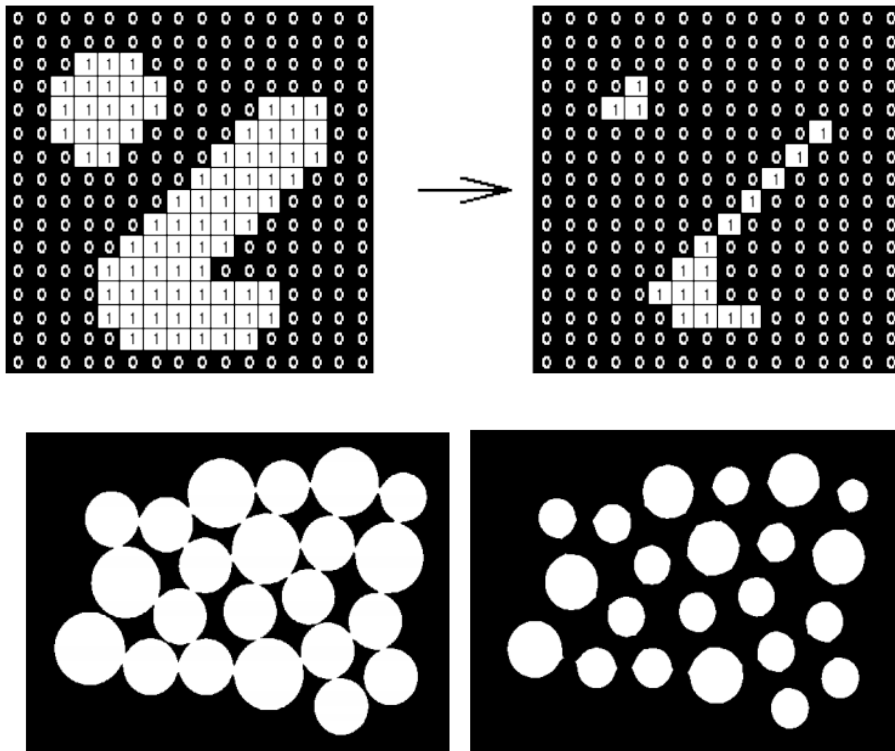
3.1.7. Morfolojik işlemler

Morfoloji, değişik bilim dallarında da yer alan bir tabirdir ve biçim bilimi anlamına gelir. Nesnelerin görüntüdeki biçimleri hem diğer nesnelere hem de arka plandan

ayrıştırılması anlamında önemli bir işlemdir. Morfolojik işlemleri görüntünün objelerini belirgin haline dönüştürmek veya görüntünün objelerini ayırt etmek için kullanılmaktadır. Görüntü işlemede biçimlerin istenilen bölgesinin keskinleştirilip istenmeyen bölgesinin ise silikleştirilmesi gerekmektedir. Morfolojik işlemler, lineer olmayan komşuluk işlemlerinde güçlü bir görüntü işleme analizidir. Morfolojik görüntü işlemede temel olarak kullanılan iki işlem vardır: Yayma ve aşındırma [23]. Gri tonlu imgeler üzerinde yapılabileceği gibi, genellikle ikili imgeler üzerinde yapılan işlemdir.

Aşındırma

Morfolojinin temel operasyonlarından biridir. Görüntü üzerinde belirginleştirme gereken kısma küçültme işlemi yapılması demektir. Görüntüde yer alan gürültü sebebiyle birbiri ile bağlantılı hale gelen alanları ayırtmak yani sınır bölgeleri aşındırmak için kullanılır (Resim 3.4). Aşındırma operatörünün işlevi, çekirdek alanı üzerinde yerel bir minimum hesaplamaya doğru yönelir.

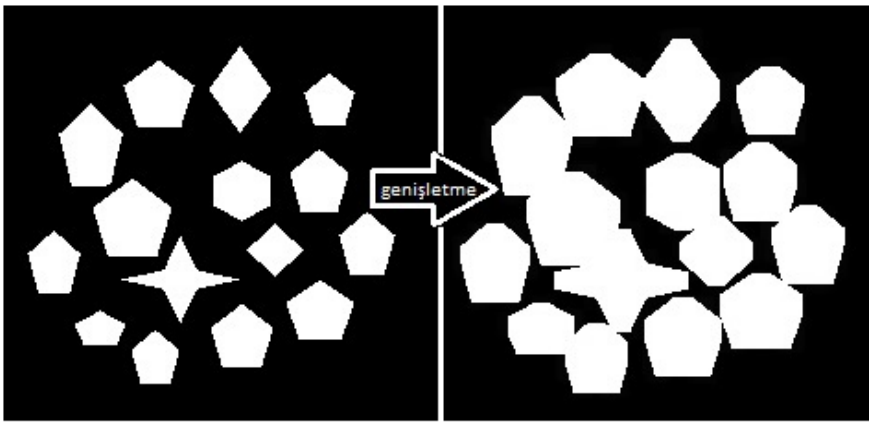
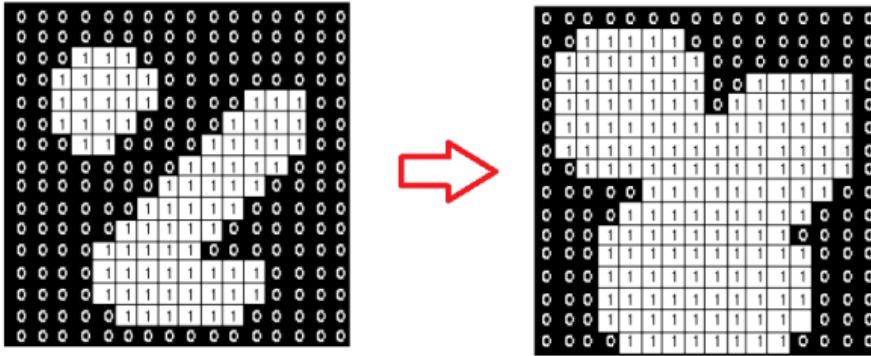


Resim 3.4. Aşındırma işlemi sonucu

Yayma

Ele alınan bölgenin sınırlarının genişletilmesinde kullanılmaktadır. Belirginleştirmek istenilen bölgenin genişletilebilmesi için kullanılır (Resim 3.5). Bu kısım etrafındaki piksellere doğru yayılım işlemi yapar. Piksel grupları genişler ve pikseller arası boş kısımlar daralır.

Açma (open), kapama (close), gradyan (meyil), top hat, siyah (black) hat gibi morfolojik işlem türleri aşındırma ve yayma işlemlerinden türemiştir. Açma önce aşındırma daha sonra yayma işlemi yapar. Bunun tam tersi işlem yani önce yayma sonra aşındırma işlemi ise kapama işlemidir. Yayılmış görüntüden aşındırılmış görüntünün çıkartılması ile elden sonuca gradyan denir. Top hat; orijinal görüntüden açma işlemi uygulanmış görüntünün çıkartılması ile ortaya çıkan görüntüdür. Black hat; kapama işlemi uygulanmış görüntüden orijinal görüntü çıkarılarak elde edilir.



Resim 3.5. Yayma işlemi sonucu

3.1.8. Filtreler

Filtreler genellikle ön işlem veya gürültü temizleme için kullanılırlar. Kaynak alınan görüntünün gürültülü olması veya ışık dengesinin düzgün olmaması gibi pek çok nedenden dolayı filtre kullanımına ihtiyaç duyulur. Kaynak görüntüye öncelikle filtre uygulanır ve görüntünün işleme için en verimli hale getirilmesi sağlanır. Bu durum ön işleme diye adlandırılır. Görüntü filtreleri ayrıntıları keskinleştirme, belirginleştirme, yumuşatma gibi amaçlar için kullanılabilir. Filtreler genellikle 3x3 çekirdek (kernel) matrislerden oluşur (Şekil 3.3). Filtreleme işlemi, bu matrislerin görüntü üzerinde denenir, görüntü üzerinde denk gelen değerlerin çarpımları ve meydana gelen tüm çarpımların toplamalarının filtre elemanlarının toplamına bölünmesi ile gerçekleştirilir. Bu işlem sırasında 3x3 matristeki merkez pikselin değeri sabit değildir değişkendir. Bu işlem sırayla tüm pikseller için yapılır ve yeni görüntü elde edilir. Bu işlemi orijinal görüntünün bütün pikselleri için uygulayıp çekirdek matrisin üzerindeki piksellere kaydırma, aktarma işlemine konvülasyon denir. Şablon matris görüntü üzerinde kaydırılıp uygun piksellerle çarpılır.

k_{11}	k_{12}	k_{13}
k_{21}	k_{22}	k_{23}
k_{31}	k_{32}	k_{33}

Şekil 3.3. 3×3 çekirdek (kernel) matris

Bulanıklaştırma (Blurring) Filtreleri

Görüntü işlemede blurring filtreleri görüntüyü yumuşatmak ya da görüntünün kenarlarını belirginleştirmek amaçlı kullanılır. Mean (ortalama) filtre, median (orta değer) filtre, gaussian blur (gauss bulanıklaştırma) filtre çeşitleri bunlara örnektir.

- *Ortalama Filtre*

Ortalama filtresi, bir piksel ile diğer pikseller arasındaki değişim miktarını azaltarak görüntüyü yumuşatır. Görüntülerdeki gürültüyü azaltmak temel amaçtır. Bir görüntünün her bir piksel değerini komşularının ve kendisinin dahil olduğu ortalama değer ile

değiştirir. Bu durum, çevresinde yer almayan piksel değerlerinin ortadan kalkmasına sebep olur. Genellikle 3x3 çekirdek (kernel) matris şablonu kullanır (Şekil 2.4). Bazı yumuşatma işlemlerinde daha büyük matrislerle işlem yapılır [28].

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Şekil 3.4. Ortalama filtresinde sıklıkla kullanılan 3x3 matris

Gürültü daha az belirgin hale getirilirken, görüntüde yumuşatılmış olmaktadır. Kullanılan çekirdek matrisin boyutu arttırılırsa yumuşatma daha da artacaktır. Ortalama filtre, görüntü üzerinde bazı bozulmalara ve sorunlara sebep vermektedir (Resim 3.6). Bunlardan birincisi; Görüntüyü iyi tanımlamayan değere sahip bir piksel, yakın bölgedeki tüm piksellerin ortalama değerini önemli ölçüde etkiler. Bu durum görüntüde bazı değişikliklere sebep olur. İkincisi; filtre bir kenar üzerinden geçerken, kenarın her iki tarafındaki pikseller için yeni değerler üretecektir ve bu durum kenarın bulanıklaşmasına sebep olacaktır. Eğer keskin kenarların kaybolması istenmiyorsa bu bir sorun olabilir [28].



Resim 3.6. Ortalama filtre sonucu

- *Orta değer filtresi*

Ortalama filtresi gibi resimdeki gürültüleri azaltmak temel amacdır. Resim üzerindeki

detayların kaybolmaması açısından ortalama filtreye göre daha iyi sonuç verdiği düşünülmektedir. Orta değer filtresi, her pikselin değerini hesaplamak için yakınındaki piksellere bakar. Piksel değeri komşu piksel değerlerinin ortalaması ile değiştirmek yerine, komşu pikselleri sıralayıp sıranın ortasındaki değeri alır. Görüntü üzerine incelenen kısımda çift sayıda piksel varsa, ortada bulunan iki pikselin ortalaması kullanılır (Resim 3.7). Orta değer filtre, komşu piksellerin birinin değeri olması gerektiği için, kenar boyunca hareket ettiğinde gerçekçi olmayan piksel değerleri oluşturmaz. Bu nedenle, keskin kenarları bozmadan korur [28].

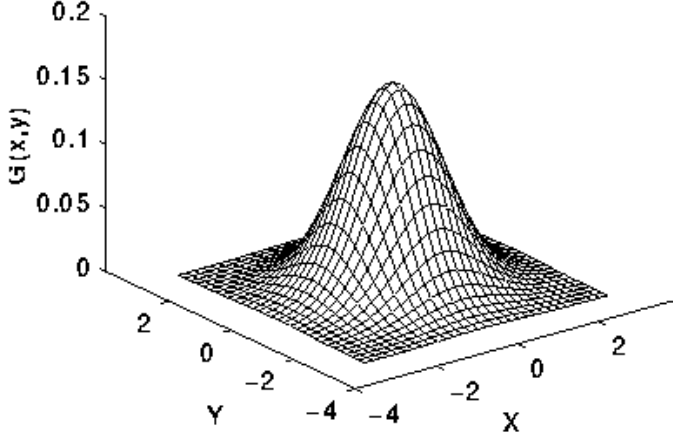


Resim 3.7. Orta değer filtresi

- *Gaussian bulanıklaştırma filtresi*

Gauss bulanıklaştırma filtresi, görüntüleri bulanıklaştırmak ve görüntü üzerinde gerekli görülmeyen kısımların ve gürültülerin giderilmesi amacıyla kullanılan 2B operatördür.

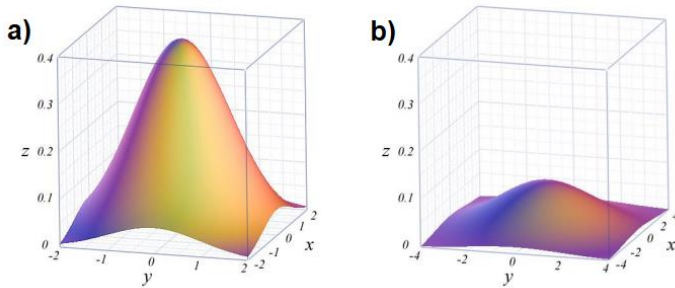
Ortalama filtreye benzer ancak kullandığı matris çekirdek (kernel) matris değildir. Çan şekline benzeyen bir çekirdek modeli kullanır (Şekil 3.5).



Şekil 3.5. Gaussian matris modeli

Teoride, Gauss dağılımı her noktada sıfırdan farklıdır ve bu sonsuz büyüklükte bir evrişim çekirdeği gerektirir. Denklemden gösterilen Gauss denklemi x ve y bileşenlerine ayrılabilir olduğundan, evrişim oldukça hızlı bir şekilde gerçekleştirilebilir. Gauss, aslında bu şekilde ayrıştırılabilen tek operatördür [32]. Burada σ dağılımın standart sapmasıdır. Ayrıca dağılımın ortalamasının sıfır olduğu düşünülmektedir. Yani Şekil 2.6'da görüldüğü gibi $x=0$ çizgisine ortalanmıştır. 2B ve 3B grafik eksenel olarak simetrik değerlere sahiptir [28]. Gaussian filtre için x ve y değerleri verilen formül şu şekildedir:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (2.1)$$

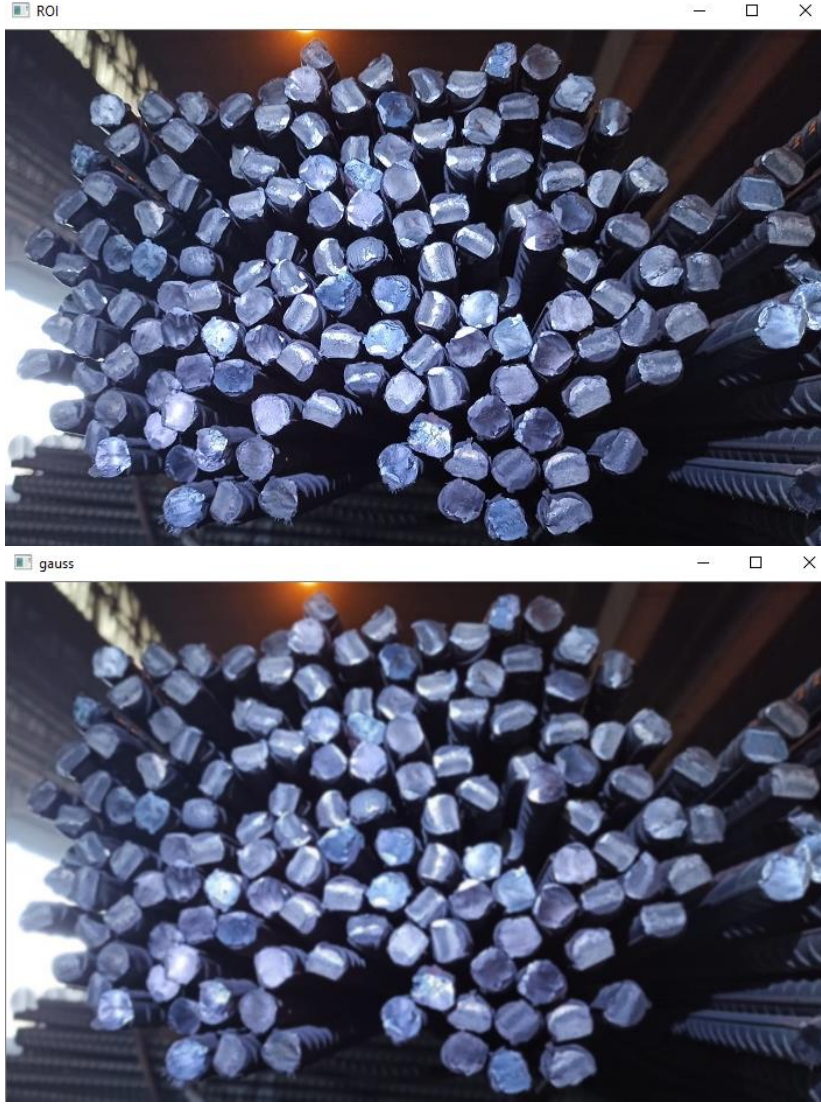


Şekil 3.6. a) Standart sapma $\sigma=1$ için grafik b) Standart sapma $\sigma=2$ için grafik

Standart sapma değeri (σ) arttıkça Şekil 3.6 'da görüldüğü gibi grafik yayvanlaşmaktadır. z değerinin sıfıra yakın bir değerden başlaması için x ve y değerlerinin daha büyük değer aralığını alması gerekmektedir. Dolayısı ile standart sapma değeri (σ) arttıkça onu temsil eden matrisleri de daha büyük seçmek uygun olacaktır [28]. Gaussian'ın resim bulanıklaştırma etkisi, bir görüntüyü ortalama filtreye yakın şekilde bulanıklaştırmaktadır. Bulanıklaştırma (düzgünleştirme) derecesi Gaussian'ın standart sapmasına göre değişmektedir. Daha büyük standart sapma değeri, ortaya çıkan grafiği daha geniş hale getirir. Bundan türetilen çekirdek matrisin de doğru bir temsil yapabilmesi için daha büyük boyutta olması gerekmektedir. Gauss, her piksel bölgesinin ağırlıklı ortalamasını almaktadır. Merkez piksel değerine doğru yaklaştıkça ağırlık artar. Bu durum, ortalama filtrenin aksine daha düzgün düzeltme sağlar (Resim 3.8). Keskin kenarları yani yüksek frekansları zayıflatır ve salınım göstermez. Ortalama filtreye göre kenarların bozulmamasında daha etkindir [28].

Kenar bulma filtreleri

Kenar bilgilerini kullanmak görüntü işleme alanında önemli bir yere sahiptir. Nesneleri birbirinden ayırma işlemlerinden birisi de kenar bilgileri ile işlem yapmaktır. Nesnelerin görünüşlerindeki renk, doku gibi özelliklerin farklılıklarından dolayı nesnelerin sınırlarında yani kenarlarında değişiklik gözlemlenir. Kenar bulma filtreleri bu sınır değişikliklerini değişik yöntemlerle tespit eder. Kenar algılama, çoğunlukla görüntü işleme gibi alanlarda görüntü bölümlenme ve veri çıkarma için kullanılır. OpenCV açık kodlu kaynak kütüphanesinde yaygın olarak kullanılan Sobel, Laplacian, Prewitt ve Canny kenar bulma filtreleri yer almaktadır.



Resim 3.8. Orijinal görüntü (üst), bulanıklaştırma filtresi uygulaması (alt)

- *Sobel kenar bulma filtresi*

Orijinal resim, siyah-beyaz (binary) görüntüye dönüştürüldükten sonra sobel filtresi kullanımı için uygun görüntü oluşturulmuş olur. Yatay, dikey ya da köşegen şeklindeki kenarları tespit etmek için 3x3 çekirdek matrisler kullanılır (Şekil 3.7). Sobel filtresi, bir resmin kenarlarına denk gelen keskin kenarları meydana çıkarır. Bu filtrenin en büyük dezavantajı gürültülü ortamlarda kenarları doğru bulamamasıdır. Resim 3.9'daki gibi sobel filtresine ait yatay ve düşey yönlü çekirdek matrisler, görüntü üzerinde ayrıca gezdirilerek sonuç görüntüsü elde edilir. Görüntünün sol üst köşesinden başlayıp görüntü üzerine yerleşir. Her piksel değerine karşılık gelen kernel katsayısıyla, piksel değeri çarpılır ve tüm sonuçlar toplanır [29].

Orijinal görüntü matrisinin denginde bir matris görüntüye yerleştirildiğinde, görüntünün kenarlarını belirginleştiren bir görüntü meydana gelir.

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Şekil 3.7. Sobel çekirdek matrisi

Sobel işlemi bir görüntüde 2 boyutlu özel bir eğim (gradyan) ölçümü gerektirir. Piksel mutlak değeri ise; şeklindedir. İlk çekirdek Gx matrisi x (sütunlar) içindeki eğimi hesaplarken diğer Gy matrisi y (satırlar) yönündeki eğimi hesaplar [31]. Gx matrisi tek başına kullanılırsa sadece yatayda renk değişimini gösterir ve ortaya çıkan çizgileri dikey olarak gözlemleriz. Gy matrisi tek başına kullanılırsa aşağıdan yukarıya doğru siyah-beyaz renk geçişini gösterir ve yatay çizgileri gözlemleriz. Eğer her iki matris mutlak formülü ile toplanırsa yatay ya da dikey çizgileri görmek yerine doğal duruşları, belirgin kenarları gözlemleriz. Resim 3.9'daki gibi sobel filtrenin görüntü üzerindeki etkisi elde edilir.



Resim 3.9. Sobel filtresi uygulaması

- *Laplasyan kenar bulma filtresi*

Bir görüntü üzerinde keskin geçişleri bulmak amacıyla kullanılan Laplasyan filtreler, ikinci türev tabanlı filtrelerdir. İki boyutlu bir $f(x,y)$ fonksiyonunun ikinci derece türevi ya da

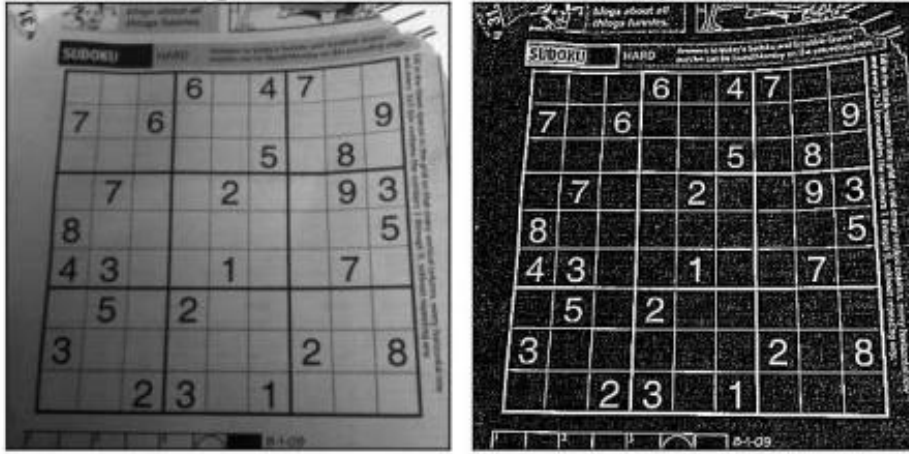
Laplasyan'ı şeklinde ifade edilerek formülü tanımlanır. x ve y yönündeki iki bileşenin toplamından oluşur:

$$L(x, y) = \nabla^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2} \left. \begin{array}{l} (0 \quad 1 \quad 0) \\ (1 \quad -4 \quad 1) \\ (0 \quad 1 \quad 0) \end{array} \right\} \quad (2.2)$$

Laplasyan filtresi, ince ayrıntıları vurgulamak ve kenarlar gibi bulanık görüntüleri iyileştirmek amaçlıdır. Laplasyan filtresinin uygulanma işlemi, giriş görüntüsü ile laplasyan matrisinin konvolüsyonlarının alınmasıdır. Bu işlem sonucunda oluşan görüntü süzgeçlenmiş görüntü olarak adlandırılır. Netleştirilmiş görüntüyü elde edebilmek için laplasyan denklemlerinden faydalanılır. Laplasyan matrisinin merkezindeki katsayı negatif ise (2.3) denklemi kullanılır. Laplasyan matrisinin merkezindeki katsayı pozitif ise (2.4) denklemi kullanılır [30]. Dolayısıyla Laplasyan filtresi, bir görüntüye uygulandığında çıkışta kenar çizgileri, sürekli olmayan gri noktalar ve siyah arka plana sahip bir görüntü elde edilir (Şekil 3.10).

$$g(x, y) = \begin{cases} f(x, y) - \nabla^2 f(x, y) \\ f(x, y) + \nabla^2 f(x, y) \end{cases} \quad (2.3)$$

Laplasyan filtresinin iki önemli dezavantajı vardır. Bunlardan biri görüntüdeki gürültüye fazla hassas olmasıdır. Diğer dezavantajı ise her bir kenar için iki değer üretmesidir. Laplasyan filtresinden önce Gauss kerneli uygulanmasının temel sebebi gürültüden kaynaklı bozulmaları önlemektir. Daha önce belirttiğimiz üzere Gauss kerneli alçak geçiren bir filtredir yani görüntüyü bulanıklaştırır.



Resim 3.10. Laplasyan filtresi uygulaması

- *Prewitt kenar bulma filtresi*

Prewitt kenar bulma filtresi, sobel filtreye benzeyen bir çalışma prensibine sahiptir. Elde edilen görüntüde daha fazla gürültü içermektedir. Dikey ve yatay keskinlik sağlamaktadır. Ancak sobel filtreden farklı iki çekirdek matris kullanır:

$$\underbrace{\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}}_{G_x} \quad \underbrace{\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}}_{G_y} \quad (2.5)$$

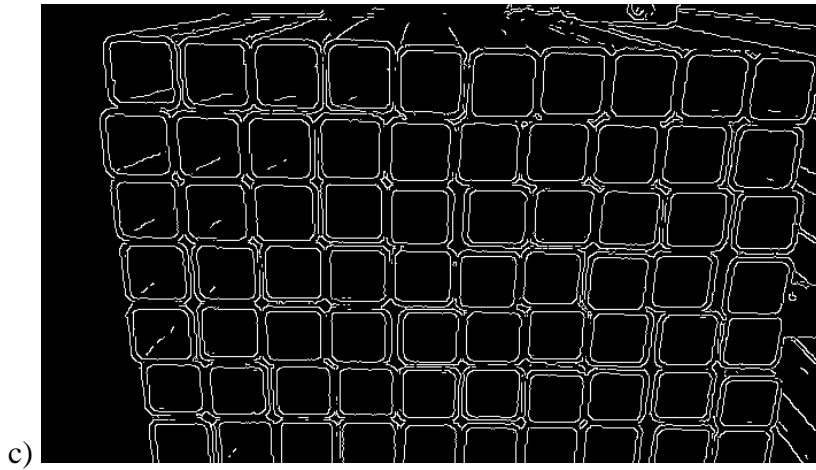
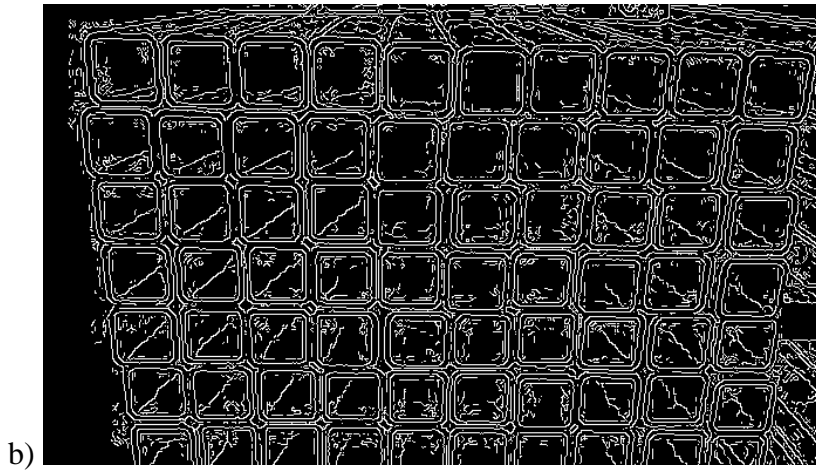
- *Canny kenar bulma filtresi*

Kenar bulma yöntemleri arasında en etkili yöntem olarak görülmektedir. Önce görüntüye bir Gaussian filtresi uygulanarak yumuşatma işlemi yapılır. Ardından, 2 boyutlu birinci türev operatörü birinci yüzeysel türevler ile görüntünün parlak bölümlerini göstermek için yumuşatma işlemi yapılan görüntüye uygulanır. Gradyan büyüklüğü, görüntü üzerindeki kenarlarda tepelere sebep olur. Algoritma bu tepeleri zirveleri boyunca takip eder ve çıkışta ince bir çizgi vermek için aslında tepelerin zirvesinde olmayan bütün pikselleri sıfıra ayarlar, canny kenar bulma maksimum olmayan noktaların kırılması olarak bilinen bir süreçtir. Takip süreci $T1 > T2$ olduğu iki eşik değeri ile kontrol edilen histerize sunar. Takip $T1$ 'den daha büyük bir tepedeki noktada başlayabilir. Takip daha sonra tepe yüksekliği $T2$ 'den aşağıya inene kadar o noktadan dışarıya her iki doğrultuda devam eder [32]. İki eşik değeri canny kenar bulma algoritmasının gürültü oranını optimize etmek için

vardır. Bu metotta uygun eşik değerini bulmak zordur, deneme yoluyla bulmak gerekmektedir. Eşik değerinin çok küçük seçerse, kenarları belirlenmiş görüntüde karşılaşılan yanlış kenarlar fazla olur. Eşik değerini çok büyük seçerse de olması gereken bazı kenarlar yok olabilir [33].

Histerezis Eşiği

Hangi kenarların kenar olduğu hangisinin kenar olmadığına karar verir. Bunun için minimum ve maksimum olmak üzere iki adet eşik değerine ihtiyaç duyulmaktadır. Maksimum değerden daha yüksek yoğunluk gradyanına sahip tüm kenarlar, kenar kabul edilir. Minimum değerinin altında kalan gradyanına sahip tüm kenarlar, kenar kabul edilmez. Bu iki eşik değeri arasında kalanlar, bağlantılarına göre kenarlar veya kenar olmayanlar olarak sınıflandırılır (Resim 3.11).



Resim 3.11. a) Orijinal görüntü, b) $T1=50$; $T2=60$ için Canny kenar bulma filtresi sonucu, c) $T1=200$ ve $T2=230$ için Canny kenar bulma filtresi sonucu

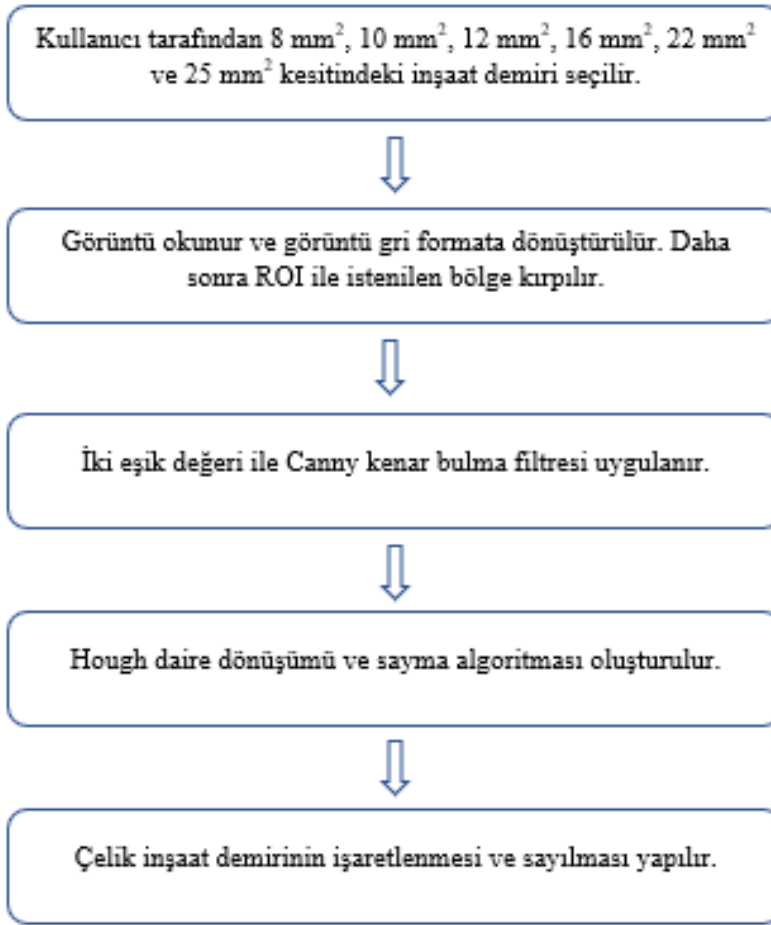
4. DENEYSEL ÇALIŞMALAR

Tez çalışmasının deneysel çalışmalar kısmında kullanılan metotlar, önerilen algoritma ve çalışmanın sonuçları anlatılmıştır. Sonraki bölümde önerilen algoritma ayrıntılı olarak açıklanmıştır. Deneysel çalışmada sayma işlemi; 8 mm², 10 mm², 12 mm², 16 mm², 22 mm² ve 25 mm² kesitindeki demir demetler çevresel şartlar ve demir çubuğun kesim hataları göz önünde bulundurulmadan akıllı telefon kamerası yardımıyla demir çelik fabrikasından alınan görüntüler ile gerçekleştirilmiştir. Gerçek zamanlı görüntülerden elde edilen demir demetleri sayılmış ve yüksek doğruluk elde edilmiştir. Bu kısımda ayrıca ara yüz tasarımı çalışmalarına değinilmiştir. Deneysel sonuçlar hem şekil hem de çizelge olarak gösterilmiştir. Son olarak, sonuçlar ve gelecekteki çalışmalar son bölümde sonuçlandırılmıştır.

4.1. Önerilen Algoritma

Önerilen algoritma, görüntü işleme metotları, OpenCV operatörleri ve sayma algoritmasına dayanmaktadır. Çelik inşaat demiri görüntülerinden gerçek zamanlı sayım yapılarak yüksek oranda doğruluk elde edilmiştir. Bu algoritma, karanlık ortam ve ışık ayarı (demir çubuk yüzeylerinin daha belirgin olabilmesi için lazer ışığı gibi aydınlatma ortamı oluşturulması) olmadan akıllı telefon kamerası ile çekimi yapılan görüntüleri saymak için oluşturulmuştur. Bozuk kesimli demir çubuk uçları, kirli ortam, yetersiz aydınlatma ve görüntü kalitesi pek düzgün olmayan görüntüler için oluşturulmuş bir algoritmadır. Önerilen algoritma için bir yazılım uygulaması gerçekleştirilmiştir ve canny kenar bulma filtresi, ilgi bölgesi, hough daire dönüşümü, sayma algoritmasına dayanır. Demir çelik sektöründe, yüzey kusurları bulunan inşaat demirleri, oksidasyonlu inşaat demirleri, arka planda kalmış inşaat demirleri gibi demet demir çubukların üstesinden gelebilmek için oluşturulmuş bir çalışmadır. İnşaat demiri sayımı sadece üretim aşamasında değil, aynı zamanda bina inşaatlarının denetiminde de önemlidir.

Önerilen algoritma, canny kenar tespiti, eşik değeri ve hough daire dönüşümü ve sayma algoritması için beş değişken parametreye sahiptir. En son aşaması ise ara yüz tasarımıdır. Önerilen algoritma 8 mm², 10 mm², 12 mm², 16 mm², 22 mm² ve 25 mm² kesitindeki çelik inşaat demiri paketlerini saymak için uygulanmış ve çelik inşaat demirinin doğru sayımı için en uygun değer aralığı belirlenmiştir. Bu adımlar Şekil 4.1'de gösterilmiştir.



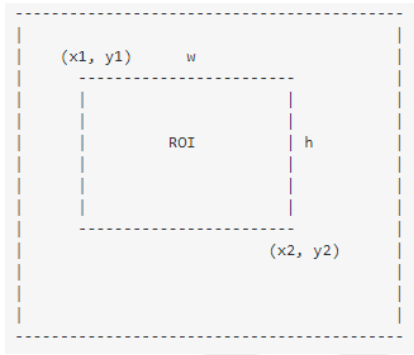
Şekil 4.1. Önerilen algoritma

OpenCV, görüntü işleme ile ilgili yüzlerce süreci içeren açık kaynak kodlu bir kütüphanedir. Video ve fotoğraflardan çeşitli görüntü dosyalarını okur ve işler. Gündüz ve gece ışık kaynakları arasındaki fark, arka plan ışığı ve tozlu ortam gibi faktörlerden dolayı görüntü üzerinde ön işlemler gerekebilir. Görüntüdeki çok sayıda parazit nedeniyle inşaat demiri sayımında “istenilen bölge” diye adlandırılan işlem uygulanır. Görüntüdeki bozuk kesimden kaynaklı demir çubuk uçlarındaki aşırı gürültüler oluşmasına ve çevresel ortamı düzenlenmeden gürültü yoğunluğu fazla bir görüntü elde edilmesine rağmen önerilen algoritma yüksek doğrulukta sayım yapabilmektedir.

Genellikle tercih edilen bulanıklaştırma filtresi, görüntüdeki gürültüyü ve kamera kusurlarını azaltmak amacıyla temel ve sıklıkla tercih edilen bir görüntü işleme adımıdır. Fakat bu filtre genellikle düzgün dairesel şekillerde çok iyi sonuç verirken, bozuk kesimli demir çubuk uçlarına sahip orijinal görüntüde pek başarılı olamamıştır.

4.2. İlgi Bölgesi (ROI)

Bu çalışmada, inşaat demirlerine ait görüntüler ile ilgili çevre düzenlemesinin olmaması, kamera ile ayarlanmaması, görüntünün anlık cep telefonu ile çekim yapılmasından kaynaklı çözünürlük değerinin istenilen düzeyde olmaması, görüntünün aşırı gürültülü olması gibi nedenlerden dolayı istenilen bölge kullanılmıştır. Kısaca, görüntü üzerinde minimum hata ile çalışabilmek için görüntü üzerinde istenilen alan seçilmiştir (Şekil 4.2). Görüntüdeki tüm piksellerin değerinden işlenecek görüntü pikselinin çıkarılmasıyla elde edilen kesit alanı anlamına gelir [2].

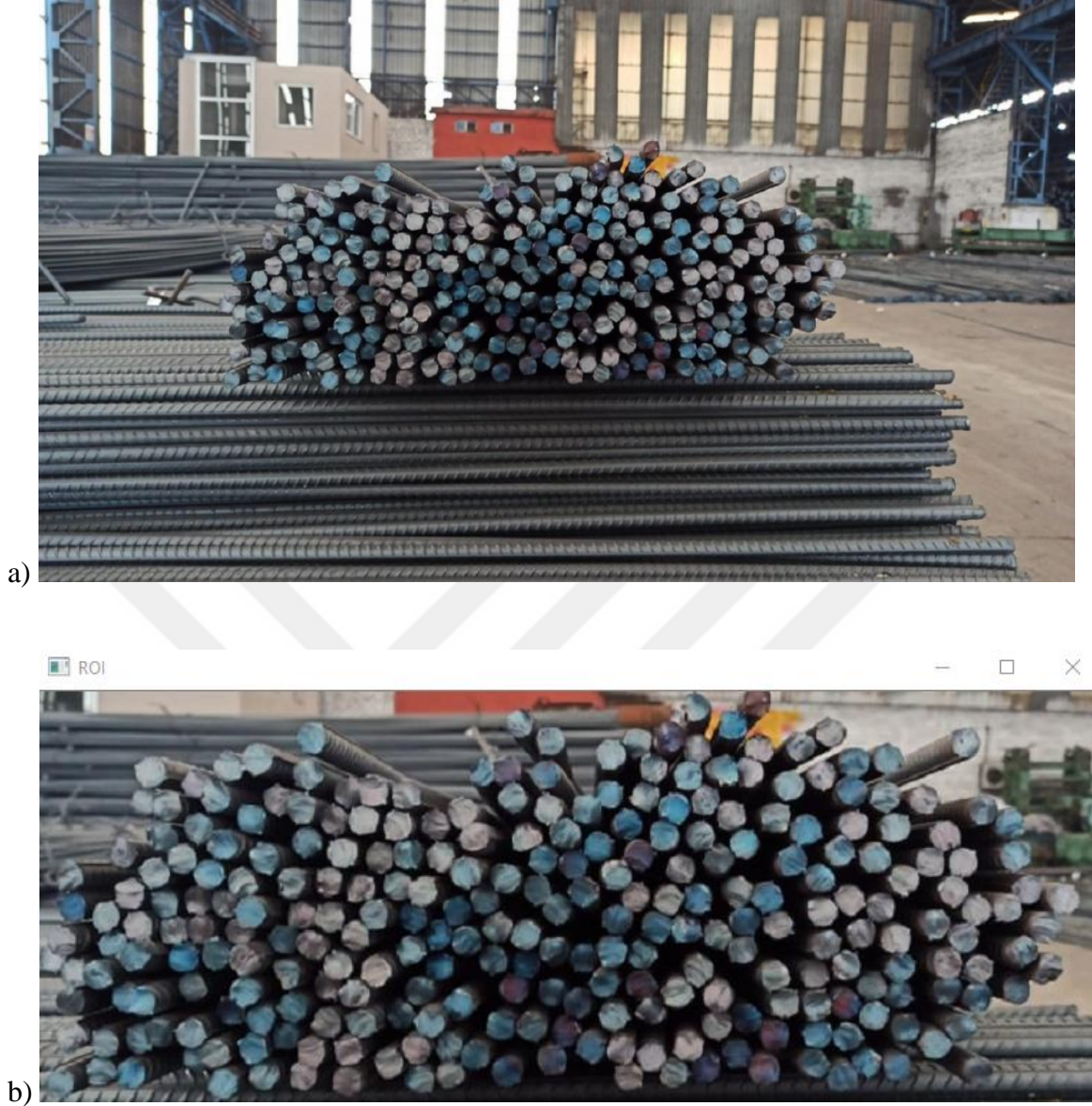


Şekil 4.2. İlgi bölgesi (ROI) formülü

İlgi Bölgesi = görüntü [167:455,208:952]

cv2.imshow ('İlgi Bölgesi', İlgi Bölgesi)

Çelik çubuk sayısının çoğunu elde etmek ve hızlı bir sayım gerçekleştirmek için ilgili bölgenin daha fazla belirlenmesi, ilgili bölgedeki algılama hatasını azaltmak uygundur. Bu çalışmada çelik inşaat demiri ile ilgili kısım, “İlgi bölgesi” kodlaması ile çıkartılmıştır. Daha sonra çıkartılan görüntü ana resim olarak algoritmada işlenmiştir (Resim 4.1).



Resim 4.1. a) Orijinal görüntü, b) İlgi bölgesi (ROI) çıkarılmış görüntü

4.3. Eşik Değeri Belirleme

Eşik değeri belirleme, canny kenar bulma filtresinde kullanılan ve ilgilenilen nesnenin ayırt edilebilmesi amacıyla arka plandan farklılaştırılması, belirginleştirilmesi sürecidir. Elemanların yalıtımının sağlanması, istenilen şeklin gürültülerden arındırılması, piksellerin yorumlanması, analiz, inceltme, görüntü sıkıştırma, köşe analizi, bozuk görüntü onarımı (eksik veya fazla piksellerin çıkarılması veya eklenmesi), dokuların algılanması gibi işlemlerde görüntü üzerine uygun bilgileri çıkarmak için kullanılır [2]. Kenar bulma filtreleri temel olarak görüntüdeki piksellerin renk değerlerinin (0-255) ayrılmasıyla

belirlenir. Sobel, prewitt, laplasyan ve canny kenar bulma filtreleri en yaygın tercih edilen kenar bulma yöntemleridir.

Bu çalışmada; sobel, prewitt, laplasyan ve canny kenar bulma filtreleri denenmiştir ve en doğru sonucun canny kenar bulma filtresi olduğu belirlenmiştir. Canny kenar bulma yönteminde çift eşik değerleri uygulanarak görüntü detaylardan arındırılır. En iyi yaklaşımlardan birisidir. Pikselin iki temel kriteri mevcuttur. Bunlar, radyan cinsinden kenar yönü ve piksel yoğunluğudur. Piksel yoğunluğu, çift eşik olarak tanımlanabilir. Bunlar yüksek ve düşük değerlerdeki eşiklerdir. Yüksek eşik, güçlü pikselleri tanımlar. Düşük eşik ise, görüntüde ilişkili bulunmayan pikselleri ifade eder. Düşük eşikten daha düşük yoğunluğa sahip piksel değerleri için kullanılır. Çelik inşaat demiri sayımında 8 mm², 10 mm², 12 mm², 16 mm², 22 mm² ve 25 mm² kesitteki demir çubukların demet görüntüleri için eşik değerleri belirlenmiştir. Her bir farklı kesitteki demir çubuklar için deneme yanılma yoluyla görüntü üzerindeki gürültüleri en düşük seviyede tutacak ve dairesel şekli en belirgin ortaya çıkaracak şekilde minimum ve maksimum eşik değerleri belirlenmiştir. Gri dağılım histogramından yararlanan eşik, eşik değer üzerindeki değer alanlarının kenar bulma aşamalarından sonra elde edilen işlenmiş görüntüdeki piksellerden belirlenen kenar verilerini içerir ve geri kalan pikseller temizlenemeyecek kenarlar olan gereksiz ayrıntılardır [2]. Nesnelere arka plandan ayırt etmek için histograma göre belirlenen bir T değeri olduğunu varsayalım. Bu T değeri, piksel $f(x,y)$ içindir ve eğer $f(x,y) > T$ için piksel kenar verilerini içerir, $f(x,y) \leq T$ için kenar verisini içermeyen arka plana ait değerleri oluşturur.

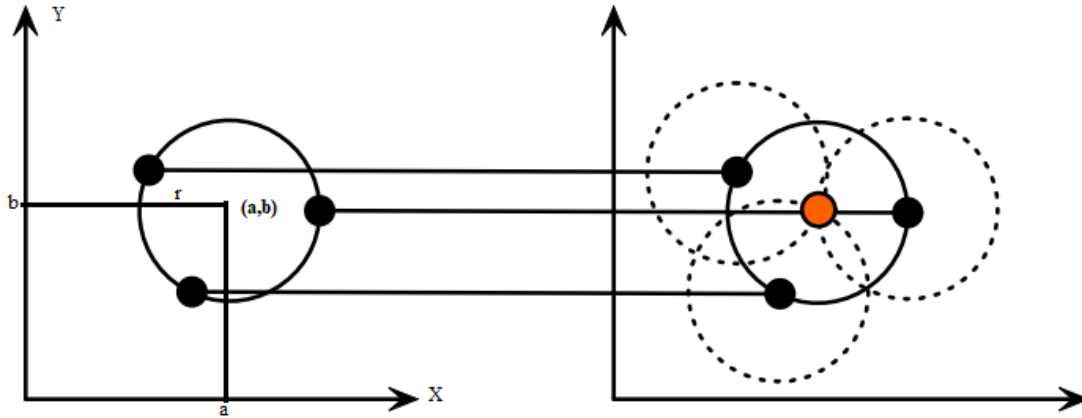
4.4. Hough Daire Dönüşümü ve Sayma Algoritması

Geometrik şekil bulmanın görüntü işlemede pek çok uygulama alanı mevcuttur. Plaka bulma, iris bulma gibi uygulamalar örnek verilebilir. Bir kamera yardımı ile elde edilen görüntülerin her zaman şekillerinin düzgün olması beklenemez. Bu durum da şekillerin tespit edilmesini zorlaştırır. Bu yöntemi kullanmak için bulunması istenilen dairelerin çapları önceden belirtilmelidir. Çalışma için görüntüleri alınan cisimlerin üzerinde bulunan kürelerin ve eğriliklerin kesitlerinin sınırları belli olduğu için, kesit sınırları arasındaki dairelerin bulunması sağlanmıştır [34]. Hough daire dönüşümü ile daire algılamak işleminde ilk adım, görüntünün kenarlarının belirlenmesidir. Hough dönüşümü, kenarın üzerindeki her sıfır olmayan piksel için yapılmaya başlar, kenar üzerindeki nokta, daire

merkezi ve r kesiti kabul edilerek daire çizimi yapılır. Akümülatörde dairenin çevresinden geçen tüm koordinatlar pikseller oylandıkça arttırılır. Akümülatör değeri en yüksek olan daireler en çok oy alan daireler olduklarından görüntü üzerinde belirgin olma olasılıkları yüksektir. Bir sonraki adım maksimum noktaların bulunmasıdır ve böylece daire dönüşümü tamamlanmış olur [35]. Görüntüler üzerinde dairesel şekil arandığından kullanılan matrisi çemberin merkezi olan $[a,b]$ noktaları ve kesit değeri $[r]$ olmak üzere üç boyuttan oluşmaktadır. Belirlenen aralıktaki kesit değerleri için çember merkezi çember denklemi ile bulunur [36]:

$$(x - a)^2 + (y - b)^2 = r^2 \quad (3.1)$$

Dairesel şeklin oluşturulması gereken parametre sayıları arttıkça, R parametre uzayının boyutu büyüdükçe, Hough daire dönüşümünün karmaşıklığı daha da artar. Bu yüzden, bir çemberin parametre uzayı, 'e ait olacaktır. Çizgi sadece 'ye ait olur. Hough daire dönüştürme yöntemiyle, kesitini önceden bildiğimiz bir daire tespit edilir. Şekil 4.3'deki gibi belli metrelerde merkez noktadan uzak noktalar atayarak daire benzeri şekiller elde etmek amaçlanmıştır. Bu yöntem bir daire bulma algoritması oluşturulabilir. Her kenar noktasında istediğimiz kesite sahip bir daire çizilebilir.



Şekil 4.3. Hough daire dönüşümü

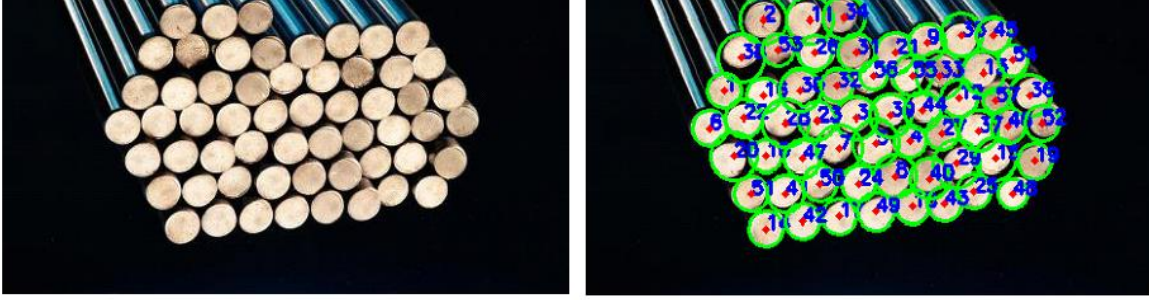
Hough daire dönüşümü uygulamasından önce, gürültülü görüntüler için ön işlemler yapılabilir. Ardından görüntünün kenarlarını tespit etmek amacıyla kenar bulma algoritması oluşturulur. Görüntü üzerindeki dairelerin kenarlarını en düzgün ve keskin olacak şekilde belirginleştirmek için en düşük ve en yüksek değerleri olan iki eşik değeri belirlenir. Sonuçlar, farklı kesit ve merkezde daireler oluşturmak için 3B akümülatör

dizisinde toplanır. Görüntü üzerinde en doğru sonuçları elde edebilmek için bazı önemli parametreler gereklidir:

- **Python Dili Komutu:** `cv2.HOUGH_GRADIENT` komutu Python dilinde parametre girmek amacıyla kullanılan bir komuttur.
 - **Dp:** Görüntü çözünürlüğünün toplayıcıyla orantısıdır. Birbirleriyle ters orantılıdır. $Dp=1$ ise, görüntü ve akümülatör aynı çözünürlüğe sahip anlamına gelmektedir.
 - **minDist:** Görüntü üzerindeki iki dairenin birbirleriyle ile en yakın mesafesinin ölçüsüdür. Bu değer çok büyük olduğunda, görüntü üzerinde var olan daireleri algılamayabilir ya da bu değer çok küçük olduğunda, çevresinde bulunandan daha fazla daire algılayabilir.
 - **Param1:** Yuen ve diğerleri yönteminde kenar algılamayı işlemek için kullanılan gradyan değeridir [37].
 - **Param2:** `Cv2.HOUGH_GRADIENT` yöntemi için akümülatör eşik değeridir. Eşik ne kadar küçükse, o kadar fazla daire algılanacaktır (yanlış daireler dahil). Eşik ne kadar büyükse, potansiyel olarak o kadar fazla daire döndürülecektir [37].
 - **Minimum yarıçap:** Görüntü üzerindeki dairelerden piksel değeri olarak en düşük yarıçap değeridir.
 - **Maximum yarıçap:** Görüntü üzerindeki dairelerden piksel değeri olarak en yüksek yarıçap değeridir.
- Bu parametre değerleri görüntü üzerindeki daire kesitlerine göre değiştirilerek değer verilir. En son olarak da sayma algoritması oluşturulur [2].

4.5. Algoritma Sonuçları

Çelik inşaat demirlerinde kesim bozukluğu, aydınlatma yetersizliği, çevresel şartların uygun olmaması gibi kusurlar bulunmadığında algoritmanın en doğru sonucu verdiği Resim 4.2’de gösterilmiştir. Görüntüde gürültü mevcut olmadığında algoritma için gerekli olan parametrelerin doğru değerlerde seçilmesi ile doğruluk oranı %100 olarak bulunmuştur.



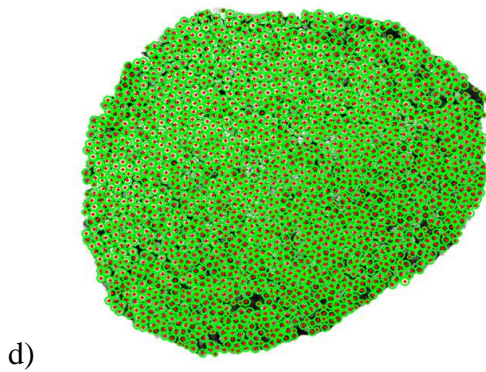
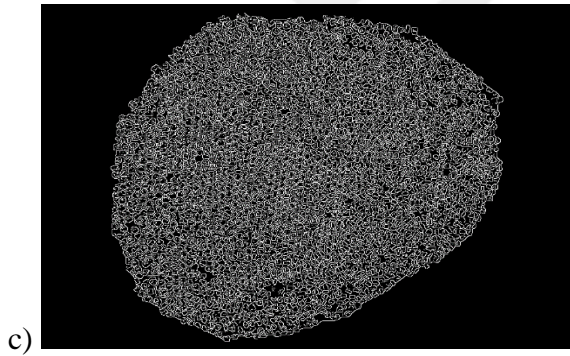
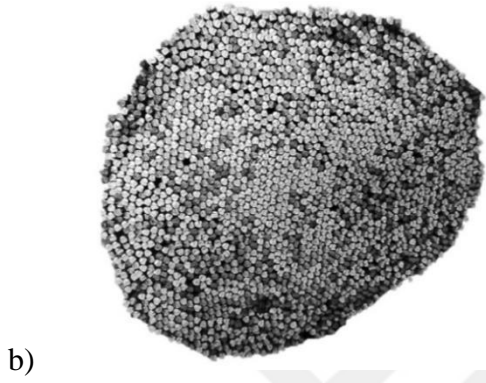
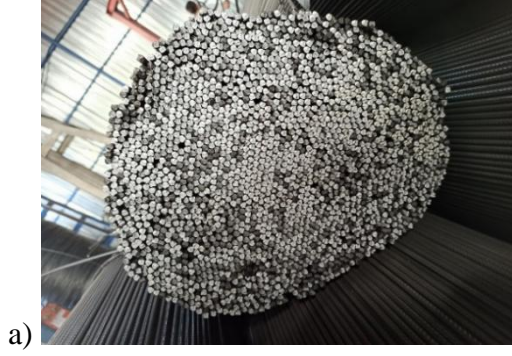
Resim 4.2. Nervürsüz demir çubuklar (solda, 57 adet), sayım sonucu (hatasız, sağda)

Bu çalışmada, fabrikadan alınan gerçek görüntüler üzerinde 8 mm², 10 mm², 12 mm², 16 mm², 22 mm² ve 25 mm² çaplarındaki inşaat demirlerinin sayımında önerilen algoritma kullanılmıştır. Görüntüler bir akıllı bir telefonun kamerası kullanılarak elde edilmiştir. Ortamın yetersiz aydınlatılması, deforme kesimi, çeliğin nervür özelliği, boyama eksikliği, tozlu ve gürültülü fabrika ortamı gibi olumsuz nedenler olmasına rağmen önerilen algoritma ile yüksek doğrulukta sayım elde edilmiştir.

4.5.1. 8mm²'lik inşaat demiri sayım sonuçları

8 mm²'lik demir çubuklar için görüntü gürültüsü en aza indirilmesi ve dairesel şeklin belirginliğinin artırılması amacıyla algoritmada yer alan parametre değerleri girilmiştir. Görüntü üzerinde en düşük ve en yüksek kesitli demir çubuğun kesiti piksel değerinden belirlenir. Ayrıca birbirine en yakın iki demir çubuğun mesafesinin piksel değeri belirlenir. T1 ve T2 eşik değerleri, gradyan değeri, akümülatör eşik değeri algoritma sonucuna göre en düzgün sonucu elde edebilmek için deneme yanılma yoluyla belirlenmiştir. Çizelge 4.1'deki gibi en yüksek verimlilik sağlayan değerler algoritmaya girdi yapılmıştır.

8 mm²'lik demir çubuk insan gözü sayımı ile 1983 adet olarak sayılmıştır. Önerilen algoritma ile %97,6 doğruluk oranı ile 2029 adet olarak sayım yapılmıştır (Çizelge 4.2). Görüntü üzerinde bozuk kesimli demir uçlarının sayısının az olması, tüm inşaat demirlerinin hemen hemen aynı hizada olması nedenlerinden dolayı görüntü üzerinde işlem yapmak verimli olmuştur. Fakat görüntü üzerinde özellikle arka planda gürültü fazlalığının yer alması ve demir çubukların uçlarının çok küçük olması dezavantajlı bir durumdur. Buna rağmen önerilen algoritma ile yüksek doğrulukta sayım elde edilmiştir (Resim 4.3).



Resim 4.3. a) (8 mm²) Orijinal görüntü, b) ROI uygulanmış görüntü, c) Canny kenar bulma filtresi uygulanmış görüntü, d) Sayım sonucu

Çizelge 4.1. 8 mm²'lik inşaat demiri için algoritma değerleri

Düşük eşik değeri (0≤T1≤255)	Yüksek eşik değeri (0≤T2≤255)	Dp	Minimum mesafe	Parametre 1	Parametre 2	Min. yarıçap	Max. yarıçap
125	125	1	10	1	7	3	8

Çizelge 4.2. Algoritma sayım sonucu (8 mm²)

Demir kesiti	Gerçek inşaat demiri sayısı	Algoritma sayım sonucu	Doğruluk oranı	Hata oranı
8 mm ²	1983	2029	%97,6	+%2,31

4.5.2. 10 mm²'lik inşaat demiri sayım sonuçları

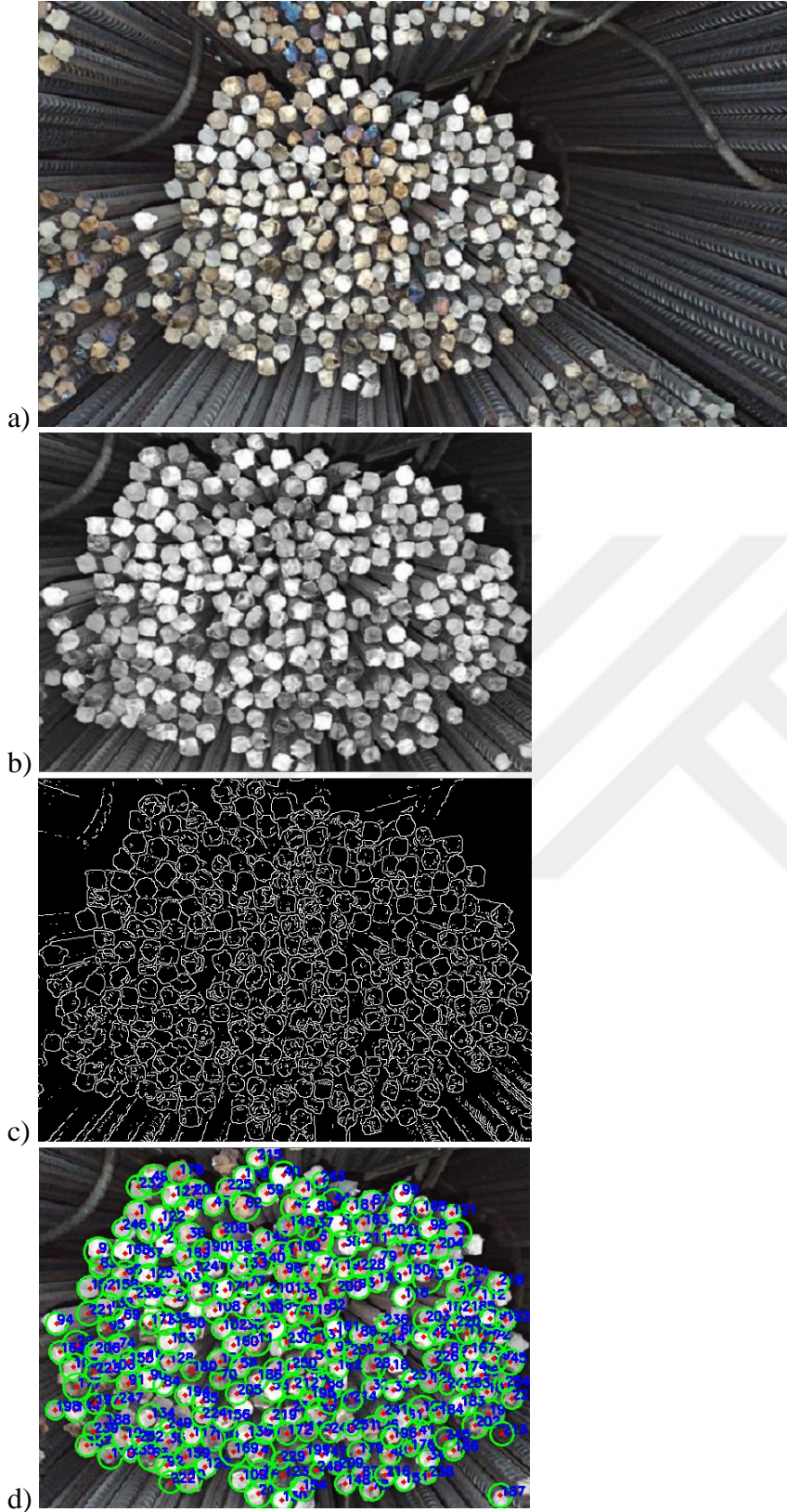
Resim 4.4'te görüldüğü gibi orijinal görüntü üzerinde; arka planın gürültülü olması, bozuk kesim, boyama eksikliği gibi gürültüler mevcuttur. Fakat görüntüde demir çubukların uçlarının aydınlatılmamasına rağmen belirgin olması doğruluk oranını arttırmada avantaj sağlamaktadır. En yüksek doğrulukta sayım yapabilmesi için algoritmanın uygun parametre değerleri girilmiştir (Çizelge 4.3). 10 mm²'lik inşaat demiri için insan gözü ile elde edilen sonuç 272 adettir. Önerilen algoritma ile program 254 adet saymıştır ve Çizelge 4.4'teki gibi doğruluk oranı %93,3'tür.

Çizelge 4.3. 10 mm²'lik inşaat demiri için algoritma değerleri

Düşük eşik değeri (0≤T1≤255)	Yüksek eşik değeri (0≤T2≤255)	Dp	Minimum mesafe	Parametre 1	Parametre 2	Min. yarıçap	Max. yarıçap
140	140	1	16	1	15	7	18

Çizelge 4.4. Algoritma sayım sonucu (10 mm²)

Demir kesiti	Gerçek inşaat demiri sayısı	Algoritma sayım sonucu	Doğruluk oranı	Hata oranı
10 mm ²	272	254	%93,3	-%6,61



Resim 4.4. a) (10 mm²) Orijinal görüntü, b) ROI uygulanmış görüntü, c) Canny kenar bulma filtresi ile görüntü sonucu, d) Sayım sonucu

4.5.3. 12 mm²'lik inşaat demiri sayım sonuçları

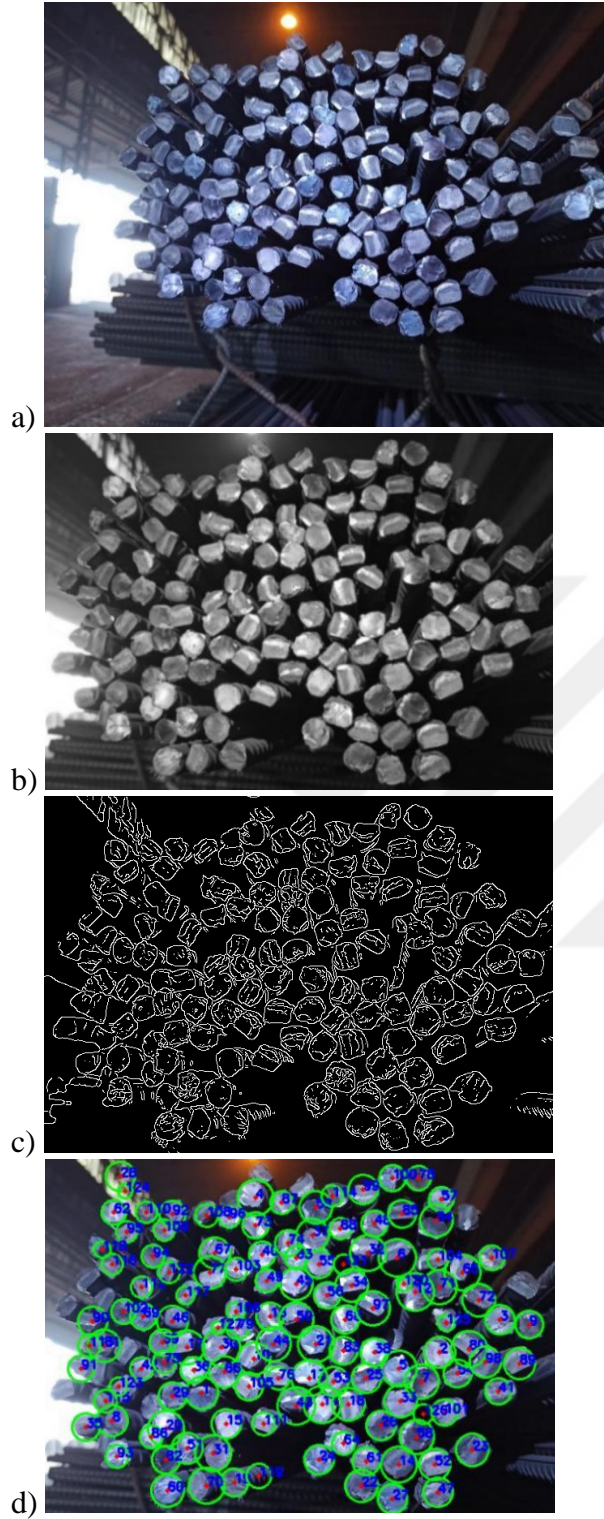
12 mm²'lik inşaat demiri görüntüsü incelendiğinde, bazı demir çubukların ayı hizada olmadığı ve hatta bazılarının çok geri planda kaldığı görülmektedir. Aynı zamanda inşaat demirlerinin depolama alanında çekilen bir görüntü olmasından kaynaklı yeterli aydınlatma bulunmamaktadır (Resim 4.5). Bozuk kesimden kaynaklı dairesellik özelliğini kaybetmiş bazı demir çubuk uçları olmasına rağmen yüksek bir doğruluk ile sayım elde edilmiştir. 12 mm²'lik kesiti olan inşaat demir demeti el ile sayım sonucu 135 adet olarak bulunmuştur. Önerilen algoritma ile oluşturulan program Çizelge 4.5'teki parametreleri ile 127 adet saymıştır ve bu algoritma ile Çizelge 4.6'da görüldüğü gibi %94 başarı elde edilmiştir.

Çizelge 4.5. 12 mm²'lik inşaat demiri için algoritma değerleri

Düşük eşik değeri (0≤T1≤255)	Yüksek eşik değeri (0≤T2≤255)	Dp	Minimum mesafe	Parametre 1	Parametre 2	Min. yarıçap	Max. yarıçap
130	130	1	22	1	18	7	22

Çizelge 4.6. Algoritma sayım sonucu (12 mm²)

Demir kesiti	Gerçek inşaat demiri sayısı	Algoritma sayım sonucu	Doğruluk oranı	Hata oranı
12 mm ²	135	127	%94	-%5,92



Resim 4.5. a) (12 mm²) Orijinal görüntü, b) ROI uygulanmış görüntü, c) Canny kenar bulma filtresi ile görüntü sonucu, d) Sayım sonucu

4.5.4. 14 mm²'lik inşaat demiri sayım sonuçları

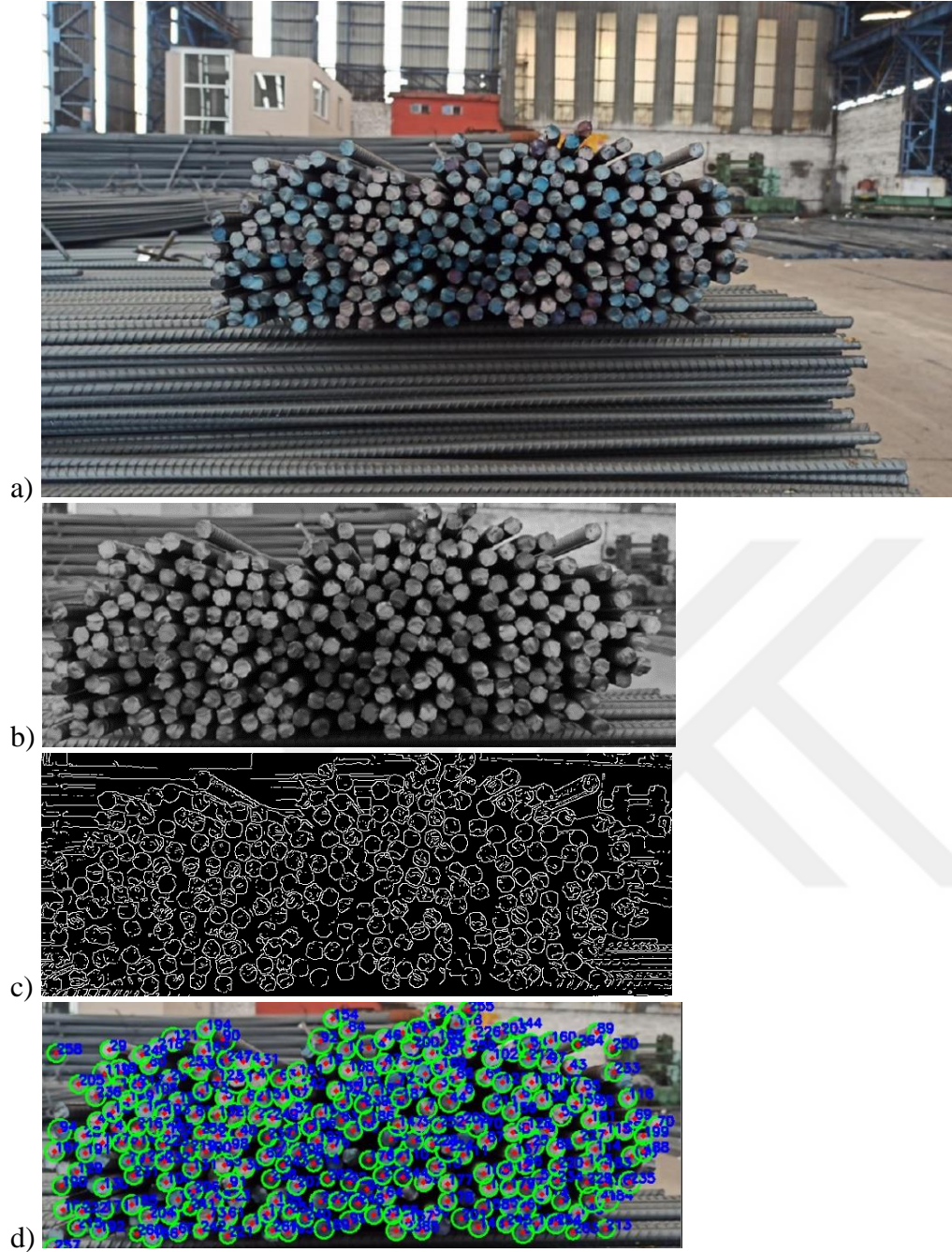
14mm²'lik demir demet görüntüsü analiz edildiğinde, karanlık ortam mevcut olmadığından arka planda çok fazla gürültü olduğu görülmektedir. Özellikle görüntünün altında ve sol üstünde yer alan demir nervürlerinden kaynaklı bozukluklar, canny kenar bulma filtresi uygulanmış görüntüde daha net olarak görülmektedir (Resim 4.6). Fakat 12 mm²'lik demir çubuk görüntüsüne göre bozuk kesimli demir uçlarının daha az olduğu dikkat çekmektedir. 14 mm²'lik kesite sahip inşaat demiri insan gözüyle 272 adet olarak sayılmıştır. Çizelge 4.7'deki parametreler kullanılarak 264 adet sayılmış, Çizelge 4.8'deki gibi %97,05 doğruluk oranı elde edilmiştir.

Çizelge 4.7. 14 mm²'lik inşaat demiri için algoritma değerleri

Düşük eşik değeri ($0 \leq T1 \leq 255$)	Yüksek eşik değeri ($0 \leq T2 \leq 255$)	Dp	Minimum mesafe	Parametre 1	Parametre 2	Min. yarıçap	Max. yarıçap
135	135	1	14	1	12	7	15

Çizelge 4.8. Algoritma sayım sonucu (14 mm²)

Demir kesiti	Gerçek inşaat demiri sayısı	Algoritma sayım sonucu	Doğruluk oranı	Hata oranı
14 mm ²	272	264	%97,05	-%2,94



Resim 4.6. a) (14 mm²) Orijinal görüntü, b) ROI uygulanmış görüntü, c) Canny kenar bulma filtresi ile görüntü sonucu, d) Sayım sonucu

4.5.5. 16 mm²'lik inşaat demiri sayım sonuçları

16 mm²'lik inşaat demiri görüntüsünde algoritmanın doğru sayımını engelleyecek pek çok unsur yer almaktadır. Bunlardan en önemlisi Resim 4.7'deki görüntülerde görüldüğü gibi demir çubukların hizalarının bozuk olmasıdır. Aynı zamanda görüntünün sol alt köşesinde yer alan demir çubukların, kalan kısmındaki demir çubuklara oranla çok daha dağınık durduğu ve aydınlatılmasının yetersiz olduğu dikkat çekmektedir. Bu durum canny kenar bulma filtresi uygulandığında daha net gözlemlenebilmektedir. Önerilen algoritma ile görüntüde mevcut olan bozuk kesim, demir çubukların hizalarının bozukluğu ve demir çubuk yüzeyindeki nervürlerden kaynaklı yüzey kusurlarına rağmen yüksek bir doğrulukla sayım yapılmıştır.

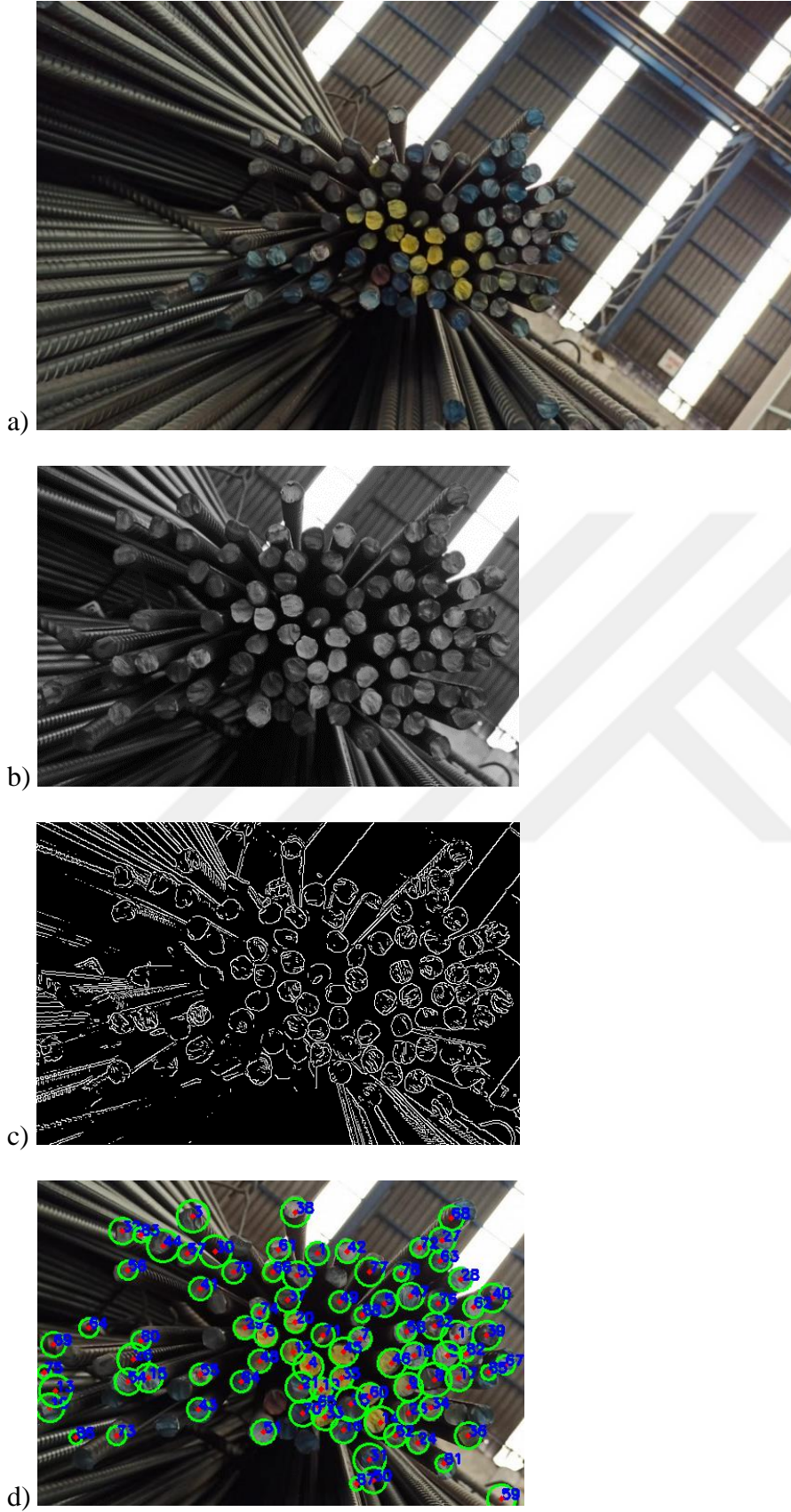
16 mm²'lik demir çubuk sayımında gözle sayım yöntemi ile 93 adet demir çubuk sayılmıştır. Çizelge 4.9'daki parametreleri ile 88 adet sayılmıştır ve Çizelge 4.10'da görüldüğü gibi %94,6 yüksek doğruluk oranı ile sayım gerçekleştirilmiştir.

Çizelge 4.9. 16 mm²'lik inşaat demiri için algoritma değerleri

Düşük eşik değeri ($0 \leq T1 \leq 255$)	Yüksek eşik değeri ($0 \leq T2 \leq 255$)	Dp	Minimum mesafe	Parametre 1	Parametre 2	Min. yarıçap	Max. yarıçap
120	120	1	15	1	15	6	19

Çizelge 4.10. Algoritma sayım sonucu (16 mm²)

Demir kesiti	Gerçek inşaat demiri sayısı	Algoritma sayım sonucu	Doğruluk oranı	Hata oranı
16 mm ²	93	88	%94,6	-%5,37



Resim 4.7. a) (16 mm^2) Orijinal görüntü, b) ROI uygulanmış görüntü, c) Canny kenar bulma filtresi ile görüntü sonucu, d) Sayım sonucu

4.5.6. 22 mm²'lik inşaat demiri sayım sonuçları

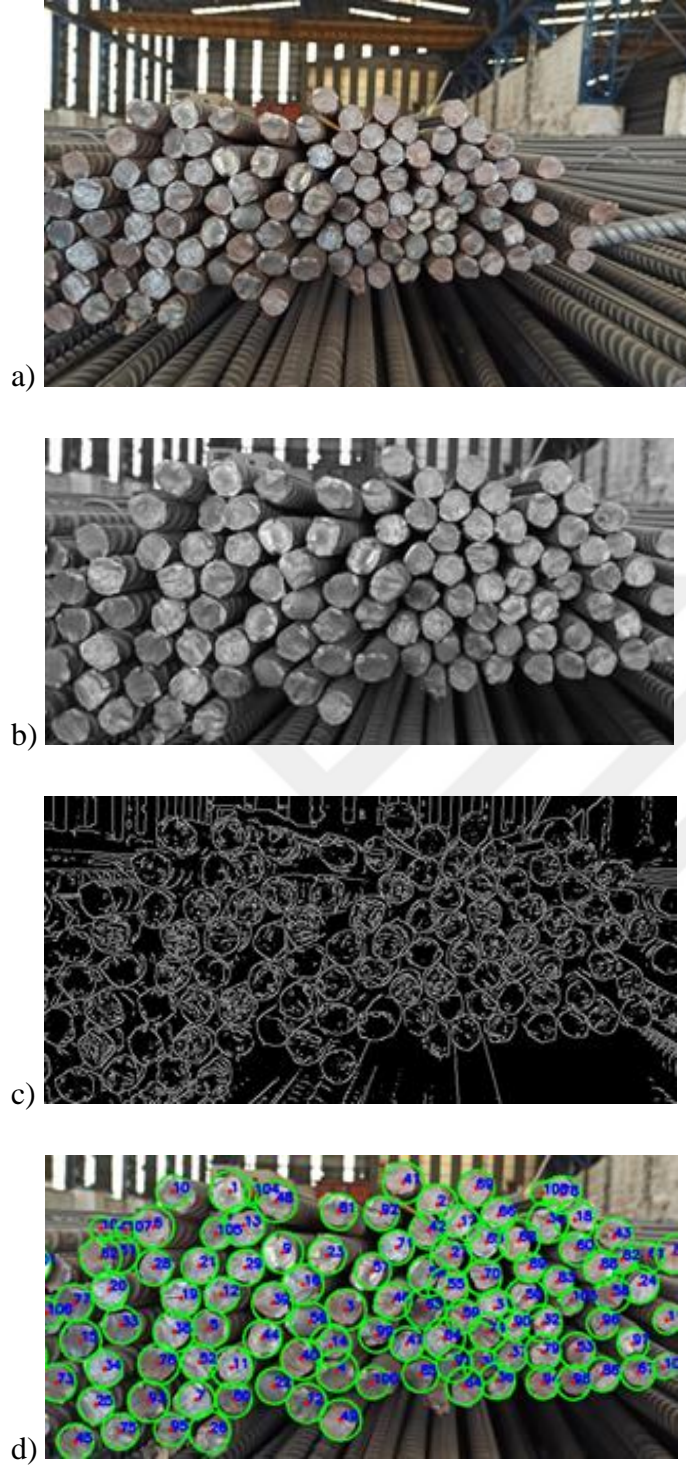
22 mm²'lik inşaat demiri görüntüsünde arka planda demir çubuk sayımını engelleyecek gereksiz unsurların fazla olması ve demir çubukların nervür özelliğinden kaynaklı yüzey kusurlarının bulunması gibi gürültüler yer almaktadır (Resim 4.8). Bu gürültüler olmasına rağmen yüksek doğrulukla sayım yapılmıştır. 22 mm²'lik demir çubuk sayımında gözle sayım yöntemi ile 103 adet demir çubuk sayılmıştır. Çizelge 4.11'deki parametreler ile 107 adet sayılmıştır ve Çizelge 4.12'de görüldüğü gibi %96,1 yüksek doğruluk oranı ile sayım gerçekleştirilmiştir.

Çizelge 4.11. 22 mm²'lik inşaat demiri için algoritma değerleri

Düşük eşik değeri (0≤T1≤255)	Yüksek eşik değeri (0≤T2≤255)	Dp	Minimum mesafe	Parametre 1	Parametre 2	Min. yarıçap	Max. yarıçap
135	135	1	22	1	20	10	25

Çizelge 4.12. Algoritma sayım sonucu (22 mm²)

Demir Kesiti	Gerçek inşaat demiri sayısı	Algoritma sayım sonucu	Doğruluk oranı	Hata oranı
22 mm ²	103	107	%96,1	+%3,88



Resim 4.8. a) (22 mm²) Orijinal görüntü, b) ROI uygulanmış görüntü, c) Canny kenar bulma filtresi ile görüntü sonucu, d) Sayım sonucu

4.5.7. 25 mm²'lik inşaat demiri sayım sonuçları

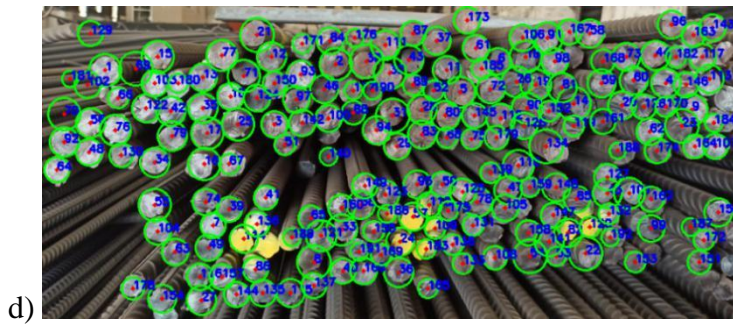
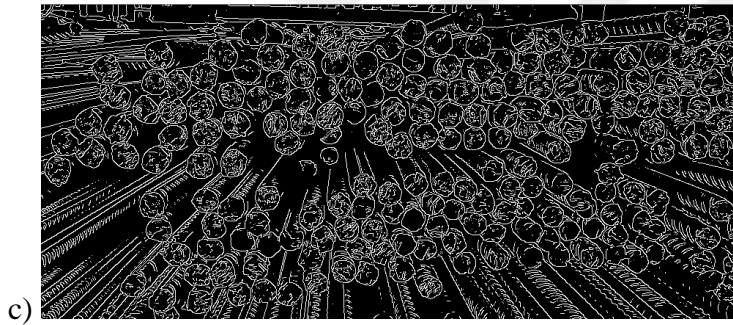
25 mm²'lik inşaat demiri görüntüsü, çevresel ortamın uygun olmaması gibi 22 mm²'lik inşaat demiri görüntüsüne benzer özelliklere sahiptir (Resim 4.9). Gözle sayılarak toplam 200 adet bulunan 25 mm²'lik demir çubuk demeti, Çizelge 4.13'deki parametreler ile 192 adet olarak %96 doğrulukla hesaplanmıştır (Çizelge 4.14).

Çizelge 4.13. 25 mm²'lik inşaat demiri için algoritma değerleri

Düşük eşik değeri (0≤T1≤255)	Yüksek eşik değeri (0≤T2≤255)	Dp	Minimum mesafe	Parametre 1	Parametre 2	Min. yarıçap	Max. yarıçap
135	135	1	22	1	19	10	24

Çizelge 4.14. Algoritma sayım sonucu (25 mm²)

Demir kesiti	Gerçek inşaat demiri sayısı	Algoritma sayım sonucu	Doğruluk oranı	Hata oranı
25 mm ²	200	192	%96	-%4



Resim 4.9. a) (25 mm²) Orijinal görüntü, b) ROI uygulanmış görüntü, c) Canny kenar bulma filtresi ile görüntü sonucu, d) Sayım sonucu

4.6. Arayüz Tasarımı

Bilgisayar ve mobil uygulamalarında gelişme çalışmaları için ara yüz tasarımı yapılmasının uygun olduğu değerlendirilmiştir. Kullanıcının konforu için grafik kullanıcı ara yüzü (GUI) tasarlandı. Tkinter, Python programlama diliyle birlikte gelen bir GUI aracıdır. Bu ara yüz tasarımı için bir Python modülü olan Tkinter kullanılacaktır. Ara yüz tasarımı için öncelikle ara yüz için gerekli olan butonlar ve değerler oluşturulup ara yüzün pencere görüntüsü ayarlanmıştır (Şekil 4.4). Ardından inşaat demirlerinin kesit bilgisi ve parametre bilgilerini içeren verilerinin olduğu veri seti oluşturulmuştur. Programda inşaat demiri sayımı için gerekli yazılım oluşturulmuştur. Geliştirilen ara yüz algoritmaya uyarlanmış ve otomatik sayma sisteminin temeli oluşturulmuştur. Demirin kesiti ve önerilen algoritma için gerekli parametreler GUI ile kullanıcı tarafından seçilir (Şekil 4.5). Daha sonra “fotoğraf seç” butonuna basılarak sayma işleminin sonucu hem ara yüz göstergesinde hem de fotoğraf üzerinde görülebilir (Şekil 4.6).

İnşaat Demiri Sayma

Kesit seç

8 mm²
 10 mm²
 12 mm²
 14 mm²
 16 mm²
 22 mm²
 25 mm²

Eşik değeri (T1) (0-255) aralığında en düşük eşik değerini gir

Eşik değeri (T2) (0-255) aralığında en büyük eşik değerini gir

min.Mesafe İki dairenin birbiri ile en yakın mesafesini gir

param1 Gradyan değeri

param2 Akümülatör eşik değeri

min.Yarıçap En küçük kesit değerini gir

max.Yarıçap En büyük kesit değerini gir

Sayım sonucu

Şekil 4.4. Tasarımı yapılan arayüz

İnşaat Demiri Sayma

Kesit seç

8 mm²
 10 mm²
 12 mm²
 14 mm²
 16 mm²
 22 mm²
 25 mm²

Eşik değeri (T1) 140 (0-255) aralığında en düşük eşik değerini gir
Eşik değeri (T2) 140 (0-255) aralığında en büyük eşik değerini gir
min.Mesafe 16 İki dairenin birbiri ile en yakın mesafesini gir
param1 5 Gradyan değeri
param2 15 Akümülatör eşik değeri
min.Yarıçap 7 En küçük kesit değerini gir
max.Yarıçap 18 En büyük kesit değerini gir

Sayım sonucu

Fotoğraf seç Çıkış

Şekil 4.5. Demir kesiti ve parametreleri yazılmış arayüz

İnşaat Demiri Sayma

Kesit seç

8 mm²
 10 mm²
 12 mm²
 14 mm²
 16 mm²
 22 mm²
 25 mm²

Eşik değeri (T1) 140 (0-255) aralığında en düşük eşik değerini gir
Eşik değeri (T2) 140 (0-255) aralığında en büyük eşik değerini gir
min.Mesafe 16 İki dairenin birbiri ile en yakın mesafesini gir
param1 5 Gradyan değeri
param2 15 Akümülatör eşik değeri
min.Yarıçap 7 En küçük kesit değerini gir
max.Yarıçap 18 En büyük kesit değerini gir

Sayım sonucu 254

Fotoğraf seç Çıkış

Detected Circle

Şekil 4.6. Arayüzün sonuç görünümü

5. DEĞERLENDİRME VE SONUÇ

Bu çalışmada, demir çelik fabrikalarından alınan ve farklı kesitlerdeki inşaat demirlerinin akıllı telefon kamerası ile çekilmiş görüntüleri kullanılmıştır. Önerilen algoritma ile ışık kaynağı ayarı ve karanlık ortam gibi herhangi bir donanımsal düzenleme yapılmadan inşaat demiri sayımı gerçekleştirilmiştir. Görüntünün gürültülerden arındırılması ve daha yüksek doğrulukta sayım yapılabilmesi amacıyla istenilen bölge tüm görüntüden ayrılmıştır. Canny kenar bulma filtresi uygulanarak iki eşik değeri belirlenmiştir. Ardından hough daire dönüşümü uygulanıp dairesel şekiller tespit edilmiştir. Sayma algoritması ile de demir çubukların sayımı yapılmıştır.

İnşaat demiri nervür özelliği, mat inşaat demiri uçları, bozuk kesimli inşaat demiri uçları, çubukların aynı hizada yer almaması, yetersiz aydınlatma gibi çevresel faktörler eksik veya fazla saymaya neden olur. İnşaat demiri aynı hizada olmamasından kaynaklı demir çubuklar arasındaki boşluk, demir çubuk sayımını olumsuz etkilemektedir. Bu nedenle, gerçek değer in altında veya üstünde sayım sonuçları tüm çevresel faktörlerin etkisinden kaynaklanmaktadır. Yapılan sayımların sonuçları Çizelge 5.1’de toplanmıştır.

Çizelge 5.1. Algoritma sayım sonuçları

Demir kesiti	Gerçek inşaat demiri sayısı	Algoritma sayım sonucu	Doğruluk oranı	Hata oranı
8 mm ²	1983	2029	%97,6	+%2,31
10 mm ²	272	254	%93,3	-%6,61
12 mm ²	135	127	%94,0	-%5,92
14 mm ²	272	264	%97,05	-%2,94
16 mm ²	93	88	%94,6	-%5,37
22 mm ²	103	107	%96,1	+%3,88
25 mm ²	200	192	%96,0	-%4

Önerilen algoritma ile donanımsal düzenlemeler olmadan demir çubuklarının depolama alanındaki görüntülerinden sayım gerçekleştirilmiştir. Tüm çevresel faktörlerin etkisine

rağmen önerilen algoritma yüksek doğrulukla sonuçlar vermiştir. Ayrıca depolama alanında en fazla 2000-2500 adet demir çubuk paketleri yer almaktadır. Doğruluk oranına bakılarak yapılan çalışma ile en fazla demir paketinde de yüksek doğrulukla sayım yapıldığı değerlendirilmiştir.

Literatür çalışmasında yer alan bazı çalışmalar ile önerdiğimiz algoritma sonuçları Çizelge 5.2’de karşılaştırılmıştır.

Çizelge 5.2. Karşılaştırmalı hesaplama zamanı sonuçları

Yöntem	Çevresel iyileştirme	Gerçek inşaat demiri sayısı	Algoritma sayım sonucu	Hesaplama zamanı (s)
Önerilen	<ul style="list-style-type: none"> • Uçlar hizalı • Uçlar aydınlatılmış • Düzgün kesim • Nervürsüz 	57	57	8,00
	<ul style="list-style-type: none"> • Uçlar hizalı • Doğal ışık ortamı • Nervürlü • Bozuk kesim 	1983	2029	12,00
	<ul style="list-style-type: none"> • Uçlar hizalı değil • Doğal ışık ortamı • Nervürlü • Bozuk kesim 	272	254	10,00
		135	127	8,63
		272	264	9,00
		93	88	9,61
		103	107	9,00
200	192	8,59		
Hernández-Ruiz ve diğerleri [20]	<ul style="list-style-type: none"> • Uçlar hizalı • Doğal ışık ortamı • Nervürsüz • Bozuk kesim 	1091	1088	70,00

Öncelikle önceki çalışmaların genelinde çevresel ortamın düzenlendiği, karanlık ortam ve ışık ayarı kullanıldığı görülmüştür. Donanımsal düzenlemelerin sonrasında çekilen

görüntülerdeki demir çubuk uçlarının belirginleştirildiği, demir çubukların aynı hizada konumlandırıldığı ve karanlık ortam ile görüntünün arka planındaki gürültüler yok edildiği gözlemlenmektedir. Çalışmamızın görüntüleri ise çevresel düzenleme yapılmadan çekilmiştir. Demir-çelik fabrikasında her zaman donanımsal düzenleme yapılmasının pek mümkün olmayacağından dolayı önerdiğimiz algoritma daha gerçekçi ve kullanışlıdır.

Sürekli üretimin devam ettiği fabrikalarda paketlerin yeniden sayımının depolama alanında gerçekleştirilmesi zaman ve enerji maliyeti açısından oldukça önemlidir. Önerilen algoritma ile elde edilen sonuçlar, görüntüden sayımı etkileyecek birçok olumsuz faktöre rağmen hem süre hem de doğruluk bakımından tatmin edici düzeydedir. Bu nedenle, önerilen sayım algoritması üretim sahalarının olumsuz şartları altında kullanılması önerilebilir.

KAYNAKLAR

1. Yılmaz, F., Dağilgan, Nazlı S., Furat, M. (2019). Image Processing Applications for Sustainable Production in Iron and Steel Industry. *7.EFRS Uluslararası Demir Çelik Sempozyumu*.
2. Dağilgan, Nazlı S., Furat, M. High Accurate Counting of Steel Rebar with Defected Tips. *Avrupa Bilim ve Teknoloji Dergisi*. Ocak 2021.
3. Wu, Y., Zhou, X. & Zhang, Y. Steel bars counting and splitting method based on machine vision. *IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER) 2015*, Shenyang, China, pp. 420-425.
4. Nie, Z., Hung, M. & Huang, J. Rebar Counting on Production Line Based on Machine Vision. *3rd International Conference on Robot, Vision and Signal Processing (RVSP), Kaohsiung*, 2015, pp. 39-42.
5. Fernández, A., Souto, M. Á. & Guerra, L. Automatic steel bar counting in production line based on laser triangulation. *IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society*, Lisbon, Portugal, 2019.
6. Z. Su, K. Fang, Z. Peng, and Z. Feng, "Rebar automatically counting on the product line. *IEEE International Conference on Progress in Informatics and Computing, Shanghai, China*, 2010, Vol. 2, pp. 756-760, doi: 10.1109/PIC.2010.5688012.
7. L. Xiaohu and O. Jineng, "Research on steel bar detection and counting method based on contours," *2018 International Conference on Electronics Technology (ICET), Chengdu*, 2018, pp. 294-297, doi: 10.1109/ELTECH.2018.8401470.
8. M. R. Ablidas, A. Monsura, L. A. Ablidas, and J. D. Cruz, "An Application of Image Processing Technology in Counting Rebars as an Alternative to Manual Counting Process," *International Journal of Simulation -- Systems, Science & Technology*, 2019, 20(5), pp. 1–9. doi: 10.5013/IJSSST.a.20.05.02.
9. J.-H., Park, T.-H., Kim and S.-Y. Choo, "Deep learning-based rebar reinforcement detection technology to improve supervision work efficiency," *Journal of the Architectural Institute of Korea: Planning Section*, 2020, Vol. 36 (5), pp. 93–103, doi: 10.5659/JAIK_PD.2020.36.5.93
10. Z. Xinman, M. Mei, H. Tingting and X. Xuebin, "Steel bars counting method based on image and video processing," *2017 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), Xiamen*, 2017, pp. 304-309, doi: 10.1109/ISPACS.2017.8266493.

11. Ying, X., Wei, X., Pei-xin, Y., Qing-da, H., & Chang-hai, C. (2010). Research on an Automatic Counting Method for Steel Bars' Image. *In 2010 International Conference on Electrical and Control Engineering* (pp. 1644-1647). IEEE.
12. Hou, W., Duan, Z., & Liu, X. (2011). A template-covering based algorithm to count the bundled steel bars. *In 2011 4th International Congress on Image and Signal Processing* (Vol. 4, pp. 1813-1816). IEEE.
13. Zhao, J., Xia, X., Wang, H., & Kong, S. (2016). Design of Real-Time Steel Bars Recognition System Based on Machine Vision. *2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*. doi:10.1109/ihmsc.2016.75.
14. Yan, X., & Chen, X. (2018). Research on the Counting Algorithm of Bundled Steel Bars Based on the Features Matching of Connected Regions. *2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC)*. doi:10.1109/icivc.2018.8492784.
15. Zhang, D., Xie, Z., & Wang, C. (2008). Bar Section Image Enhancement and Positioning Method in On-Line Steel Bar Counting and Automatic Separating System. *2008 Congress on Image and Signal Processing*. doi:10.1109/cisp.2008.664.
16. Pathan, S. S., & Harale, A. D. (2016). Automated silkworm eggs count. *2016 IEEE International Conference on Advances in Electronics, Communication and Computer Technology (ICAECCT)*. doi:10.1109/icaecct.2016.7942551.
17. Bhaskar, P. K., & Yong, S.-P. (2014). Image processing based vehicle detection and tracking method. *2014 International Conference on Computer and Information Sciences (ICCOINS)*. doi:10.1109/iccoins.2014.6868357.
18. Han, X., & Wang, J. (2019). Design of Paper Counting Algorithm based on Texture Image. *2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*. doi:10.1109/iaeac47372.2019.8997773.
19. Kaymak, A. M. M., Örnek, M. N. N., & Kahramanlı, H. (2019). Görüntü İşleme Teknolojilerinin Elma Bahçelerine Yönelik Kullanım Örneği. *Uluborlu Mesleki Bilimler Dergisi*, 2(1), 17-26.
20. Hernández-Ruiz, A. C., Martínez-Nieto, J. A., & Buldain-Pérez, J. D. (2021). Steel Bar Counting from Images with Machine Learning. *Electronics*, 10(4), 402.
21. Eldem, A., Eldem, H., Palalı, A. (2017). Görüntü İşleme Teknikleriyle Yüz Algılama Sistemi Geliştirme. *BEÜ Fen Bilimleri Dergisi*, 6(2), 44-48.
22. Kuyumcu, B. (2018). *OpenCv Görüntü İşleme ve Yapay Öğrenme*. İstanbul: Umuttepe Yayınları, pp. 12-13.
23. Ergün, H. (2019). Yapraklı ve İğne Yapraklı Ağaçlarda Homojen Öz Işınlardan Morfolojik İşlemlerle Belirlenmesi. *Mühendislik Bilimleri ve Tasarım Dergisi*. 7(1), 52 – 59.
24. Aydın, E.D. (1994). Bilgisayar Vizyon ve Görüntü İşleme. *Marmara İletişim Dergisi*.

25. Kısa, M., (2019). Grafik Tasarım ve Baskı Ortamında Kullanılan Görsellerin RGB Renk Uzayından CMYK Renk Uzayına Dönüşümü Esnasında Oluşan Renk ve Ton Kayıplarının Önlenmesi. *Humanities Sciences*.
26. Pengsen, K., & Zhenming, Y. (2017). Image Blurred Region Detection Based on RGB Color Space Information and Local Standard Deviation. *2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*. doi:10.1109/iaeac.2017.8054403.
27. Kartika, D. S. Y., & Herumurti, D. (2016). Koi fish classification based on HSV color space. *2016 International Conference on Information & Communication Technology and Systems (ICTS)*. doi:10.1109/icts.2016.7910280
28. İnternet Kaynağı: Görüntü İşleme Filtreleri
http://www.ibrahimcayiroglu.com/Dokumanlar/GoruntuIsleme/Goruntu_Isleme_Ders_Notlari-5.Hafta.pdf
29. Aybar, E. (2008). Sobel İşleci Kullanılarak Renkli Görüntülerde Kenar Bulma. *Afyon Kocatepe Üniversitesi Fen Bilimleri Dergisi*, pp. 208-209.
30. Aslan, M. F. OpenCL Ortamında Görüntü İyileştirme İşlemlerinin Paralel Programlama Yöntemiyle Gerçekleştirilmesi. Yüksek Lisans Tezi, Isparta Süleyman Demirel Üniversitesi, 2018.
31. Sanida, T., Sideris, A., & Dasygenis, M. (2020). A Heterogeneous Implementation of the Sobel Edge Detection Filter Using OpenCL. *2020 9th International Conference on Modern Circuits and Systems Technologies (MOCASST)*. doi:10.1109/mocast49295.2020.9200249.
32. Dinçer, S. Dijital Görüntü İşleme Teknikleri ile Matlab ve Mikrokontrolör Kullanılarak İki Boyutlu Görüntünün Çizdirilmesi. Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, 2009.
33. Turan, B. Görüntü İşleme Algoritmalarının Eş Zamanlı Süreçlere Ayrılarak Kablosuz Ağ Üzerinden Gerçeklenmesi ve Performans Analizleri. Yüksek Lisans Tezi, Hacettepe Üniversitesi, 2017.
34. Uçar, K. Silindirik Cisimlerin Görüntü İşleme ile Hata Tespiti. Yüksek Lisans Tezi, Konya Selçuk Üniversitesi, 2018.
35. Rezai, S. Elektrostatik Tabanlı Çok Tepeli Parçacık Sürü Optimizasyon Algoritması ile Çoklu Daire Algılama. Yüksek Lisans Tezi, Gazi Üniversitesi, 2017.
36. Çelik, A., Tekin, E. Hough Transform Görüntü İşleme Yöntemiyle Ekim Makineleri için Tohum Sayma Uygulaması. *Avrupa Bilim ve Teknoloji Dergisi Özel Sayı*, S. 260-270, Nisan 2020.
37. İnternet Kaynağı: Hough Circle Transform
<https://www.pyimagesearch.com/2014/07/21/detecting-circles-images-using-opencv-hough-circles/>

DİZİN

A

Akümülatör · 39
algoritma · iv, 5, 8, 9, 10, 12, 13, 15, 34, 40, 41, 43, 45,
47, 49, 51, 53, 57
ara yüz · 16, 34, 55

B

bilgisayar · 3
bozuk kesim · iv, 5, 43, 49

C

canny · 27, 31, 33, 38, 42, 44, 46, 48, 50, 52, 54, 57
canny kenar bulma · 10, 31, 34, 37, 38, 47, 49

D

daire · 8, 9, 10, 34, 38, 39, 40, 57
demet · 3, 6, 10, 45, 53
demir çelik endüstrisi · iv
demir çubuk · iv, 1, 5, 8, 9, 10, 11, 12, 34, 35, 41, 45,
47, 49, 51, 53, 57
Demir çubuk sayımı · 6

E

eşik değeri · 9, 10, 11, 12, 13, 31, 32, 34, 39, 41, 57

F

filtreler · 11, 13, 15, 30

G

Görüntü · iv, 1, 5, 6, 9, 12, 13, 15, 16, 17, 20, 21, 23, 24,
39, 40, 41, 59, 60
Görüntü işleme · iv, 5, 6, 12, 13, 15, 16, 17
gradyan · 8, 22, 29, 40, 41
GUI · xiii, 16, 19, 55
gürültülü · 15, 22, 28, 36, 39, 41, 43

I

ışık ayarı · 34

K

kalite kontrol · iv, 1, 2, 6
Kamera · 4
karanlık ortam · 34, 47, 57

L

literatür · 13

M

matris · 23, 24, 25, 29, 31
morfolojik · 11, 13, 15, 22

N

nervür · iv, 1, 5, 41, 51, 57

O

okuma · iv
Open CV · 15, 19, 20, 27, 34, 35

P

parametre · 39, 40, 41, 55

Piksel · 21, 24, 29, 38

Python · 16, 40, 55

R

renk uzayı · 13, 17, 18, 19

S

sayım · iv, 4, 5, 8, 9, 10, 11, 12, 13, 34, 36, 41, 42, 43,
44, 45, 47, 49, 51, 53, 57

T

Tkinter · 55

Y

yazılım · 15, 34, 55

yüksek sıcaklık · iv, 1, 2, 4



TEKNOVERSITE



teknoversite **AYRICALIĞINDASINIZ**

İSTE

