KAMERA İLE GÖRÜNTÜ İŞLEME TEKNİKLERİYLE MALZEME TANE BÜYÜKLÜĞÜ ANALİZİ

Hikmet BAL

YÜKSEK LİSANS TEZİ
MAKİNE MÜHENDİSLİĞİ

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

EYLÜL 2006
ANKARA

Hikmet BAL tarafından hazırlanan KAMERA İLE GÖRÜNTÜ İŞLEME TEKNİKLERİYLE MALZEME TANE BÜYÜKLÜĞÜ ANALİZİ adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Prof. Dr. Mehmet EROĞLU

Tez Yöneticisi

Bu çalışma, jürimiz tarafından Makine Mühendisliği Anabilim Dalında oy birliği ile Yüksek Lisans tezi olarak kabul edilmiştir.

Başkan : Prof. Dr. Kemal ÖZGÖREN

Üye : Prof. Dr. Yücel ERCAN

Üye : Prof. Dr. Mehmet EROĞLU (Danışman)

Üye : Doç.Dr. Şefaatdin YÜKSEL

Üye : Y. Doç. Dr. Metin Uymaz SALAMCI

Tarih : 11 Eylül 2006

Bu tez, Gazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygundur.

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Hikmet BAL

KAMERA İLE GÖRÜNTÜ İŞLEME TEKNİKLERİYLE MALZEME TANE BÜYÜKLÜĞÜ ANALİZİ

(Yüksek Lisans Tezi)

Hikmet BAL

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
Eylül 2006

ÖZET

Bu tezin amacı bir malzemeye ait görüntülerden, görüntü işleme teknikleri kullanılarak malzeme özelliklerini belirlemektir.

Bu çalışmanın ilk bölümünde hazırlanan yazılım ile model olarak seçilen değişik doğrultularda yerleştirilmiş domino taşlarına ait görüntülerden her bir taşın değeri bilgisayar ortamında belirlendi. Domino taşlarına ait görüntüler sayısal kamera kullanılarak görüntü yakalama yazılımı ile bilgisayar ortamına aktarıldı. Model tanıma için değişik görüntü çözünürlükleri kullanıldı. Uygulamanın ikinci bölümünde ise ilk bölümde hazırlanan yazılım geliştirilerek malzeme görüntüleri için uyarlandı. Malzeme görüntülerinden, hazırlanan yazılım ile malzeme içerisindeki küresel grafit alanı yüzdesi veya istenen fazın piksel alan yüzdesi hesaplandı. Bu işlemler için görüntü iyileştirme, filtreleme, yapısal düzenleme, bölümlendirme görüntü işleme teknikleri uygulandı. Hazırlanan program bu teknikleri uygulayarak elde edilen verilerden görüntü ile ilgili istenen sonuçları üretmektedir.

Sonuçlar: i- muntazam aydınlatmanın bölümlendirmede görüntü için en etken resim özelliği olduğu, ii- görüntü çözünürlüğünün uygulamaya göre önem taşıdığı, iii- görüntü tanıma işleminde yapısal özelliklerin

seçici olduğu, iv- görüntüsü alınan malzemeye uygulanacak örnek hazırlama işlemlerinin belirlenmek istenen özelliklerin doğruluğunu direkt etkilediğini göstermiştir.

Bilim Kodu : 914.1.091

Anahtar Kelimeler : Görüntü sistemi, görüntü işleme, kenar bulma, resim tanıma, tane büyüklüğü

Sayfa Adedi : 230

Tez Yöneticileri : Prof. Dr. Mehmet EROĞLU

GRAIN SIZE ANALYSIS FROM MATERIAL IMAGES BY USING IMAGE PROCESSING TECHNIQUES

(M.Sc. Thesis)

Hikmet BAL

GAZI UNIVERSITY
INSTITUTE OF SCIENCE AND TECHNOLOGY
September 2006

ABSTRACT

The aim of this thesis is to determine the material properties from the material images by using image processing techniques.

With the software which is developed at the first stage of this study, each domino's value was determined from domino's images which are oriented at various directions and selected as a model in the computer environment. Domino's images have been transferred to computer environment by using digital camera within video capture software. It is used various video resolutions to model recognition. As for the second stage of application, previous developed software has been enhanced and adapted to material images. From material images, spheric graphite area percentage or desired phase pixel area percentage is calculated with the developed software. Image enhancing, image filtering, morphological construction and segmentation image processing techniques have been used for this processes. Developed program produces desired results related to images from acquired data by applying those techniques.

Results: i- In segmenting, the most effective image property is regular illumination for the image, ii- image resolution has significance according to the application, iii- structural properties of objects are selective in image recognition processing, iv-it is shown that sample preparation processing which will be applied to the material directly affect the accuracy of properties desired to determine.

Science Code : 914.1.091
Keywords : Vision system, image processing, edge detection, pattern recognition, grain size
Page : 230
Supervisor : Prof. Dr. Mehmet EROĞLU

TEŞEKKÜR

Çalışmalarımın her aşamasında değerli yardım ve engin hoşgörüsünü benden esirgemeyen Sayın Hocam Prof. Dr. Mehmet EROĞLU' na, yine, yabancı kaynaklardaki yardımlarından dolayı kıymetli arkadaşım Mevlüt BÜYÜKKILIÇ' a, Gazi Ünv. Tek.Eğt. Fak. Metal Eğt. Böl.'den, malzeme görüntü işlemedeki verdiği bilgilerden dolayı Arş.Gör. Volkan KILIÇLI'ya , program algoritmaları konusunda bilgilendirmelerinden dolayı kıymetli arkadaşım ve ağabeyim Bilgisayar Mühendisi Murat BAL'a, çalışmamda maddi ve manevi desteğini her zaman yanımda hissettiğim ve çalışmalarım nedeniyle kendisine ayırmam gereken zamanı veremediğim bitmek tükenmek bilmeyen sabır ve desteğiyle her zaman yanımda olan Sevgili Eşim Elektronik Yüksek Mühendisi Tuba BAL'a ve ayrıca bu günlere gelmemi sağlayan çok kıymetli aileme teşekkürlerimi borç bilirim.

İÇİNDEKİLER

Sayfa

ÖZET iv

ABSTRACT vi

TEŞEKKÜR viii

İÇİNDEKİLER ix

ÇİZELGELERİN LİSTESİ xv

ŞEKİLLERİN LİSTESİ xvi

RESİMLERİN LİSTESİ xx

SİMGELER VE KISALTMALAR xxiii

1. GİRİŞ 1

2. GÖRÜNTÜ VE SAYISAL GÖRÜNTÜNÜN YAPISI 11

2.1. Görsel Algılamanın Temeli 11

2.1.1 İnsan gözünün yapısı 11

2.1.2 Gözde görüntünün oluşumu 14

2.1.3 Parlaklık adaptasyonu ve ayrımı 15

2.2 Bilgisayar Ortamında Görüntü Yapısı 17

2.2.1 Kamera Projeksiyon Modeli 17

2.2.2 Işık ve elektromanyetik spektrum 19

2.3 Görüntü Algılama ve Elde Etme 23

2.3.1 Algılayıcı dizileri (sensor arrays) kullanarak görüntü elde etme 24

2.3.2 Kameralar 25

2.4 Basit Bir Görüntü Oluşturma Modeli 27

Sayfa

2.5 Görüntü Örnekleme (Sampling) ve Nicelikleştirme (Quantization) 29

2.5.1 Örnekleme ve nicelikleştirmede temel kavramlar 29

2.6 Sayısal Görüntünün İfade Edilmesi 31

2.7 Pikseller Arasındaki Temel İlişkiler 34

2.7.1 Bir pikselin komşuları 34

2.7.2 Yakınlık, bağlanabilirlik bölgeler ve sınırlar 34

3. GÖRÜNTÜ HAZIRLAMA VE GELİŞTİRME 37

3.1 Giriş 37

3.2 Gri Seviye Dönüşümü 38

3.2.1 Negatif dönüşüm 39

3.2.2 Logaritmik ve kontrast kerme dönüşümü 40

3.2.3 Üssel dönüşüm 42

3.3 Histogram İşleme 43

3.3.1 Histogram eşitleme 45

3.4 Görüntü Tanım Bölgesinde Filtreleme 47

3.4.1 Filtrelemenin temeli 47

3.4.2 Yumuşatma filtreleri 50

3.4.3 Keskinleştirme Filtreleri 52

4. GÖRÜNTÜ YAPISAL İŞLEMLERİ 59

4.1 İkili Düzey (Binary) Görüntü Yapısal İşlemleri 59

4.1.1 Genişleme (dilastion) 60

4.1.2 Aşınma (erosion) 61

Sayfa

4.1.3 Açma ve kapama (opening and alosing) 62
4.1.4 Hit or Miss dönüşümü 63
4.1.5 Yeniden Yapılandırma 63
4.2 Gridüzey Görüntü Yapılandırma İşlemleri 64
4.2.1 Genişleme ve aşınma (Dilation - Erosion) 64
4.2.2 Açma ve kapama (opening – closing) 65
4.2.3 Gri seviye Yeniden Yapılandırma 67
5. BÖLÜMLENDİRME (SEGMENTATION).....68
5.1 Kenar Belirleme 68
5.2 Kenar Birleştirme ve Sınır Belirleme 70
5.2.1 Bölgesel işleme 70
5.2.2 Hough dönüşümü ile genel işleme.....71
5.3 Eşikleme (Thresholding) 72
5.3.1 Eşiklemenin temeli.....72
5.3.2 Eşiklemede aydınlatmanın rolü.....74
5.3.3 Genel eşikleme 75
5.3.4 Uyarlanabilir eşikleme 76
5.3.5 Optimum genel ve uyarlanabilir eşikleme 77
5.4 Bölge Tabanlı Bölümlendirme.....79
5.4.1 Bölge tabanlı bölümlendirmenin temeli 80
5.4.2 Bölge büyütme metodu.....80
5.4.3 Bölge bölme ve birleştirme metodu 81
5.5 WaterShed Bölümlendirme 81

Sayfa

5.6 Özellik Çıkarma....83
5.6.1 Alan....83
5.6.2 Çevre....83
5.6.3 Delik sayısı....84
5.6.4 Delik alanı....84
5.6.5 Yuvarlaklık....84
5.6.6 Yarıçap....85
5.6.7 Kütle merkezi(mentroid)....85
5.6.8 Doğrultu....85
5.6.9 Birbirleriyle ilişkili piksel gruplarının etiketlenmesi (labeling)....87
5.6.10 Zincir kodlama....89
5.7 Nesne Tanıma (Sınıflandırma)....89
5.7.1 Kalıplar ve kalıp sınıfları....90
5.7.2 Karar teorik tabanlı sınıflandırma metotları....90
6. RENKLİ RESİM İŞLEME....98
6.1 Renk Temelleri....98
6.2 Renk Modelleri....103
6.2.1 RGB renk modeli....103
6.2.2 CMY ve CMYK renk modelleri....105
6.2.3 HSI Renk Modeli....106
6.3 Renkleri Modelleri Çevirme....108
6.3.1 Renkleri RGB' den HSI' ya çevirme....108

Sayfa

6.3.2 Renkleri HSI' den RGB' ye çevirme 109
6.3.3 Gri seviyeden renkliye dönüşümler.....111
6.4 Tam-Renk Görüntü İşleme Temelleri 111
6.5 Renk Dönüşümleri 113
6.5.1 Formülasyon.....113
6.5.2 Histogram İşleme.....115
6.5.3 Renkli görüntü yumuşatma.....115
6.5.4 Renkli görüntü keskinleştirme.....116
6.5.5 RGB görüntüde bölümlendirme 116
6.5.6 Renk kenar belirleme.....117
7. GÖRÜNTÜ İŞLEME UYGULAMASI ve SONUÇLARI.....119
7.1 Birinci Bölüm Uygulama (Domino Taşlarını Tanıma) 119
7.1.1 Sistem.....119
7.1.2. Çözünürlük 124
7.1.3. Görüntü yakalama ve görüntü formatı 127
7.1.4. Görüntü işleme 127
7.2 İkinci Bölüm Uygulama (Malzeme Görüntü İşleme) 149
7.2.1 Malzeme görüntü işleme 151
7.2.3 Yeniden yapılandırma 152
7.2.4 Bölümlendirme 154
7.2.5 İkinci bölüm uygulama sonuçları 161
8. SONUÇLAR VE ÖNERİLER.....173
KAYNAKLAR 176

Sayfa

EKLER 179
EK-1 Kullanılan donanımlar ve özellikleri 180
EK-2 Kamera kalibrasyonu 189
EK-3 Birinci bölüm uygulamada kullanılan kaynak kodlar 200
EK-4 İkinci bölüm uygulamada kullanılan kaynak kodlar 213

ÖZGEÇMİŞ 230

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

Çizelge 7.1 Orjinal görüntü ve filtrelenmiş görüntü için eşik değerleri 136

Çizelge 7.2. Durum 1 için elde edilen değerler..... 142

Çizelge 7.3. Durum 2 için elde edilen değerler..... 144

Çizelge 7.4. Durum 3 için elde edilen değerler.....146

Çizelge 7.5. Durum 4 için elde edilen değerler.....148

Çizelge 7.6. Küresel Grafitli Dökme Demir için sonuçlar 164

Çizelge 7.7. Küresel grafitli Martenzit çeliği için sonuçlar 167

Çizelge 7.8. İşlemsiz çelik için sonuçlar 168

Çizelge 7.9. Küresel Grafitli Dökme Demir için sonuçların karşılaştırması 169

Çizelge 7.10 İşlemsiz Çelik için sonuçların karşılaştırması 171

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

Şekil 1.1. Görüntü sistemleri ilişkisi ve uygulama alanları 3

Şekil 2.1. Gözün yapısı 12

Şekil 2.2. Görüntü açısı – koni, çubuk sayısı grafiği 13

Şekil 2.3. Bir nesne görüntüsünün gözde oluşumu. (C göz merceği merkezi) 14

Şekil 2.4. Parlaklık farkını algılamak için kullanılan düzenek 15

Sekil 2.5. Bazı optik yanılsamalar 16

Şekil 2.6. pinhole kamera projeksiyon modeli 18

Şekil 2.7. Tüm elektromanyetik spektrum ve görünür spektrum 19

Şekil 2.8. Elektromanyetik dalga formu ve dalgaboyu 20

Şekil 2.9. Mercek 22

Şekil 2.10. Piksel sensoru ve sensor dizilimleri 23

Şekil 2.11. Algılama elemanı ile sayısal resmin elde edilmesi 24

Şekil 2.12. 256 düzey gri renk 29

Şekil 2.13. Örnekleme ve sayısallaştırma 30

Şekil 2.14. Nesnenin resim düzlemi üzerindeki izdüşümü ve sayısallaştırılmış hali 31

Şekil 2.15. Sayısal resim koordinat yapısı 32

Şekil 2.16. a)rastgele bir piksel yerleşimi, b)8-yakınlık, c)m-yakınlık 35

Şekil 3.1. Görüntü içindeki bir (x,y) noktası ve 3x3 komşuluğu 38

Şekil 3.2. Gri seviye dönüşümleri, (negatif, üssel, logaritmik) 39

Şekil 3.3. Logaritmik ve kontrast germe dönüşümü 41

Şekil **Sayfa**

Şekil 3.4. farklı γ değerleri için muhtemel dönüşüm eğrileri 43

Şekil 3.5. 3X3 lük bir maske ile, maskenin resim üzerinde uygulanma yapısı 49

Şekil 3.6. 3X3 lük bir filtre maskesinin gösterimi 49

Şekil 3.7. iki farklı yumuşatma filtresi 50

Şekil 3.8. 3X3 lük bir resim altbölümü 51

Şekil 3.9. Bir görüntü satırı için birinci ve ikinci türev 53

Şekil 3.10. Farklı laplacian maskeler 55

Şekil 3.11 Resim altbölümü 56

Şekil 3.12. LoG filtre örneği 57

Şekil 3.13. Gradient filtre örneği.....58

Şekil 4.1. İkili resim üzerinde uygulanan mantıksal işlemlerin gösterimi 59

Şekil 4.2. Genişleme işlemi 60

Şekil 4.3. Aşınma işlemi 61

Şekil 4.4. Açma ve Kapama işlemi.....62

Şekil 4.5. Hit or miss işlemi 63

Şekil 4.6. a) resim satırı gri seviye profili, b) açma işlemi, c) açma işlemi sonucu, d) kapama işlemi, e) kapama işlemi sonucu 66

Şekil 4.7. a) yeniden yapılandırma için marker ve maske görüntüleri, b)yeniden yapılandırma işlemi, c) yeniden yapılandırma işlem sonucu 67

Şekil 5.1. Kenar profil modeli 69

Şekil 5.2. Sayısal kenarın birinci ve ikinci türevi 69

Şekil 5.3. Histogram görüntüsü 73

Şekil **Sayfa**

Şekil 5.4. Aydınlatma fonksiyonun etkisi 74

Şekil 5.5. Uniform aydınlatılmamış resmin bölümlendirilmesi 77

Şekil 5.6. Parçalanmış resim ve “quadtree” yapısı 81

Şekil 5.7. a) gri seviye görüntü, b) görüntünün üç boyutlu gösterimi 83

Şekil 5.8. Nesne çevresi 84

Şekil 5.9. Nesnedeki delik 84

Şekil 5.10. Bir doğrunun polar koordinatlarla ifadesi 86

Şekil 5.11. Piksel komşuluk setleri 88

Şekil 5.12. Etiketleme (Labelling) 88

Şekil 5.13. Freeman zincir kodları 89

Şekil 5.14. Minimum mesafe sınıflandırma 92

Şekil 5.15. İki kalıp sınıfı için muhtemel yoğunluk grafiği 95

Şekil 6.1. Görünür ışığı da içeren Elektromanyetik spektrum 98

Şekil 6.2. Toplama renk bileşenleri, çıkarma renk bileşenleri 100

Şekil 6.3. C.I.E. kromatiklik grafiği, kırmızı üçgen renkli monitörlere ait, mavi şekil ise renkli yazıcılara ait renk aralığını göstermektedir 102

Şekil 6.4. RGB birim renk küpü 104

Şekil 6.5. RGB ve HSI renk modelleri arasındaki ilişki, renk özü ve doygunluk bileşenlerinin gösterimi 107

Şekil 6.6. Gri-skala ve RGB renkli görüntüler için uzaysal maskeler 112

Şekil 7.1. Histogram işleme menü seçenekleri 128

Şekil 7.2. Orjinal görüntüye ait histogram ve eşitlenmiş histogram 130

Şekil 7.3 Filtre menüsü 131

Şekil **Sayfa**

Şekil 7.4. Bölümlendirme menusu 135

Şekil 7.5. İkinci bölüm uygulama ana menu görüntüsü 152

Şekil 7.6. Bölümlendirme menüsü..... 156

Şekil 7.7. Küresel grafitli Dökme Demirin diğer yazılım ile işlenmesi 169

Şekil 7.8. (a) Küresel grafitli dökme demir için diğer yazılımın elde ettiği sonuçlar (b) Küresel grafitli dökme demir için diğer yazılımın elde ettiği sonuçlara ait diğer aşamanın gösterimi 170

Şekil 7.9. İşlemsiz çelik görüntüsünün diğer yazılım ile işlenmesi.....171

Şekil 7.10. (a) İşlemsiz çelik için diğer yazılımın elde ettiği sonuçlar (b) Küresel grafitli dökme demir için diğer yazılımın elde ettiği sonuçlarıngösterimi 172

RESİMLERİN LİSTESİ

Resim **Sayfa**

Resim 3.1. RBG ve gri görüntünün negatif dönüşümleri 40

Resim 3.2. Sırasıyla *E*=2,8 ve 50 için log dönüşümü.(m için resim ortalama gri değeri alınmıştır.) 42

Resim 3.3. Bazı görüntülere ait histogram grafikleri 44

Resim 3.4. Histogram eşitleme yapılmış görüntüler 46

Resim 7.1. Kullanılan sistem 120

Resim 7.2. Capture yazılımı arayüzü 123

Resim 7.3. Değişik çözünürlükler 125

Resim 7.4. (a) 640*480 lik görüntü,(b) 320*240'lik görüntü, (c) 160* 120'lik görüntülerin 640*480 piksel olarak gösterimi 126

Resim 7.5. (a) Orjinal resim , (b) Histogram işlenmiş resim 129

Resim 7.6. (a) Ortalama filtre uygulanmış görüntü (b) 5*5 boyutlarında sigma=0.2 için Gaussian filtrelemenin sonucu 132

Resim 7. 7. 5*5 boyutlarında sigma=2 için orijinal görüntünün Gaussian filtreleme sonucu (b) Laplacian filtrelenmiş görüntü (alpha=0).....133

Resim 7. 8 (a) Log (laplacian of gaussian) filtrelenmiş görüntü (b) Median filtre sonuçu.....134

Resim 7.9. (a) Sobel operatörü ile elde edilmiş kenarlar. (b) Prewitt operatörüyle elde edilmiş kenarlar. 137

Resim 7.10. (a) Roberts operatörü ile, (b) LoG operatörü ile ,(c) Canny operatörü ile elde edilmiş kenarlar 138

Resim 7.11. (a) Genel eşikleme ile Nesne sınırları (kırmızı) ve iç sınırlar (yeşil). (b) Genel eşikleme ile elde edilmiş ikili görüntü 139

Resim 7.12. Durum1 için elde edilen sonuç 141

Resim **Sayfa**

Resim 7.13. Durum 2 için elde edilen sonuç 143

Resim 7.14. Durum 3 için elde edilen sonuç 145

Resim 7.15. Durum 3 için elde edilen sonuç.....146

Resim 7.16. (a) Küresel grafitli dökme demir. (b) Küresel grafitli martenzit çeliği 150

Resim 7.17.. İşlemsiz 1030 çeliği 151

Resim 7.18. (a) Yeniden yapılandırılmış küresel grafitli dökme demir görüntüsü (b) Yenide yapılandırılmış martenzit çeliği görüntüsü 153

Resim 7.19. Yeniden yapılandırılmış işlemsiz çelik görüntüsü 154

Resim 7.20. Dökme demir görüntüsündeki küresel grafitler 155

Resim 7.21. (a) ikili düzey martenzit çeliği görüntüsü (b) ikili düzey çelik görüntüsü 156

Resim 7.22. Küresel grafitli dökme demire ait gradient genlik görüntüsü....157

Resim 7.23. Küresel grafitli dökme demir için önplan görüntüsü.....158

Resim 7.24. Kürsel grafitli dökme demir için geriplan görüntüsü 158

Resim 7.25. Martenzit çeliği için gradient genliği görüntüsü 159

Resim 7.26. Martenzit çeliği için önplan görüntüsü 160

Resim 7.27. Martenzit çeliği için geriplan görüntüsü 160

Resim 7.28. Eşikleme ile elde edilen tane sınırları beyaz olarak çizili 161

Resim 7.29. Küresel grafitli dökme demir için eşikleme sonrası elde edilen tane boyutları 162

Resim 7.30. Watershed tekniği ile elde edilen tane sınırları beyaz ile gösterilmiştir.....162

Resim **Sayfa**

Resim 7.31. Küresel grafitli dökme demir için watershed sonrası elde edilen tane boyutları 163

Resim 7.32. Watershed ve Eşikleme ile bulunan tane sınırlarının karşılaştırması. (kırmızı watershed ile bulunan sınırlar, sarı Eşikleme ile bulunan sınırlar) 163

Resim 7.33. Eşikleme ile elde edilen tane sınırları beyaz olarak çizilmiştir 164

Resim 7.34. Küresel grafitli martenzit çeliği için eşikleme sonrası elde edilen tane boyutları 165

Resim 7.35. Watershed tekniği ile elde edilen tane sınırları beyaz renkle çizilmiştir 165

Resim 7.36. Küresel grafitli martenzit çeliği için watershed sonrası elde edilen tane boyutları 166

Resim 7.37. Watershed ve Eşikleme ile bulunan tane sınırlarının karşılaştırması. (kırmızı watershed ile, sarı Eşikleme ile bulunan sınırlar) 166

Resim 7.38. Bölümlendirme sonucunun orijinal görüntüde gösterimi 167

Resim 7.39. Belirlenen Siyah faza ait bölge sınırları kırmızı ile çizilmiştir .168

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış bazı simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	Açıklama
I	Aydınlatma şiddeti
$\mathbf{I_c}$	Zemin aydınlatması
$\mathbf{(x_p, y_p)}$	Görüntü düzleminde bir p noktasının koordinatı
$\mathbf{(X_P, Y_P, Z_P)}$	Nesneye ait bir P noktasının gerçek koordinatı
$\boldsymbol{\lambda}$	Dalgaboyu
c	Işık hızı
h	Planck sabiti
v	Frekans
e	Elektromanyetik spektrum enerjisi
f	Odak uzaklığı
f	İki boyutlu görüntü fonksiyonu
y	Aydınlatma fonksiyonu
yan	Yansıma fonksiyonu
(x, y)	Görüntü koordinatları
l	Gri seviye değeri
Lmin	Minimum gri seviye değeri
Lmax	Maksimum gri seviye değeri
(MxN)	Görüntü matrisinin boyutları
A	Özdeş görüntü matrisi
i	Matris satır göstergesi
j	Matris sütun göstergesi
^{k}	Renk sayının bit olarak ifadesi
$\boldsymbol{B_s}$	Bit olarak resim veri boyutu
$\mathbf{N_D}$	Diagonal komşuluk
$\mathbf{N_4}$	4'lü komşuluk

Simgeler	Açıklama
N_8	8'lü komşuluk
q	Görüntü üzerindeki herhangi bir nokta
D	Dönüşüm işlemi
g	İki boyutlu işlenmiş görüntü fonksiyonu
s	Çıkış gri seviyesi
r	Giriş gri seviyesi
c_1	Logaritmik dönüşüm sabiti
m	Kontrast germe eşiği
E	Kontrast germe dönüşümü üssel sabiti
γ	Üssel dönüşüm değişkeni
c_2	Üssel dönüşüm sabiti
n_k	Bir gri seviyedeki piksel sayısı
h	Histogram fonksiyonu
n	Toplam piksel sayısı
p_r	Giriş görüntüsü MYF
p_s	Çıkış görüntüsü MYF
ϑ	İntegral sabiti
R	Filtreleme işlemi merkez piksel değeri
(mxn)	Filtre maskesinin boyutları
z_i	Resim alt bölümü pikseli
w_i	Filtre maskesi pikseli
∇^2	Laplacian operatörü
Γ	Gauss fonksiyonu
σ	Standart sapma
∇	Gradient operatörü
Gx	Yatay yönde gradient vektör
Gy	Dikey yönde gradient vektör
$\lvert G \rvert$	Gradient vektörün genliği

Simgeler	Açıklama
E_T	Gradient vektörün eşik değeri
α	Gradient vektörün açısı
A_T	Gradient vektörün açı eşiği
ρ	Polar koordinatlarda genlik
θ	Polar koordinatlarda açı
a	Doğru denklemi eğimi
b	Doğru denklemi sehim
T	Genel eşik değeri
T_1	Alt eşik değeri
T_2	Üst eşik değeri
c_3	Yansıma yüzeyi sabiti
η	Normalize edilmiş görüntü
G_1	Birinci grup pikseller
G_2	İkinci grup pikseller
E	Yanlış sınıflandırma olasılığı
p_1	Nesne için MYF
p_2	Geriplan için MYF
μ	Ortalama gri seviye değeri
Prop	Bölge özelliği
Im	Resim bölgesi
A_k	Piksel alanı
B	Etiketlenmiş resim
C	Yuvarlaklık
$P_ç$	Çevre
$(\overline{X}, \overline{Y})$	Kütle merkezi koordinatları
x	Kalıp vektörü
di	Karar fonksiyonu
ϖ	Kalıp sınıfı
D_j	Euclidean uzaklık

Simgeler	**Açıklama**
$\mathbf{L_{ij}}$	Sınıflandırma kaybı
$\mathbf{r_j}$	Kalıp sınıfı için ortalama kayıp
$\bar{c}$	Renkli görüntüde noktanın vektörel ifadesi

Kısaltmalar	**Açıklama**
MYF	Muhtemel yoğunluk fonksiyonu
(RGB)	Kırmızı,yeşil,mavi renk modeli
(HSI)	Renk özü, doygunluk, ışık şiddeti renk modeli
(CMYK)	cam göbeği, menekşe, sarı, siyah renk modeli

1. GİRİŞ

Görme yeteneği insanoğlunun çevresini ve olayları algılamasında ve tepki vermesinde tek başına bile merkezi rol oynamaktadır. Görüntü bilgisini işleyip analiz etmeyi konu alan makine görüşü sürekli gelişmekte olan bir araştırma alandır. Akıllı sistemlerin gelişmesinde anahtar bir rol oynar. Akıllı sistemler üzerine olan araştırmaların temel amacı algılayabilen, mantık yürütebilen, hareket edebilen ve öğrenebilen makineler geliştirmektir. Makinalar için ise görme, yeni bir teknolojidir. Uzay ve havacılık uygulamalarına paralel olarak görüntü işleme teknikleri 1970'lerin başlarında kullanılmaya başlanmıştır. Endüstrideki çeşitli kullanıcı grupları görüntü sistemlerini makinaları yönlendirme, kalite kontrol gibi uygulamalarda son on yıldır kullanmaktadırlar. Fakat yakın zamana kadar sistemler yüksek teknik seviye ve programlama uzmanlığı gerektirdiğinden, konuyla ilgili uzmana sahip veya büyük ölçekli projeler yapanlar için uygulama alanı olmuştur. Makine görüşü araştırmaları, resim ve resim sıralarının analiz edilmesine yönelik sistem ve yöntemler üzerine yoğunlaşır. Amaç, ortam değişiklikleri bakımından sağlam, hesaplamada randımanlı, performansta önder yeni yaklaşımlar üretmek ve algılayabilen, mantık yürütebilen, hareket edebilen ve öğrenebilen makineler geliştirmektir.

Görüntü sistemlerinde donanım ve makina görüntü yazılımları beraber kullanılarak alınan bir görüntü saniyenin dilimlerinde analiz edilir. Bu bölümde parçanın yeri, doğrultusu, kritik boyutlarının ölçümü, parça kodunun okunması veya doğrulanması veya parçanın doğrulanması yapılabilir. Alınan her görüntü işlendikten sonra elde edilen bilgiler robotlar, bilgi toplama sistemleri veya PLC' lere girdi olarak iletilir. Makina Görüntü sistemlerinin performansını etkileyebilecek iki önemli faktör aydınlatma ve işlenen malzemelerdir. Tipik olarak pek çok hat, parçaları belli bir pozisyonda tutan ve takibi sağlayan konveyör, sıralama tablaları veya robot kolları kullanır. Aydınlatma gereksinimleri uygulamadan uygulamaya basit aydınlatma, çevresel aydınlatma, geriden aydınlatma, yüksek hızlı hatlar için anlık donma etkisi veren aydınlatma gibi farklılıklar gösterir.

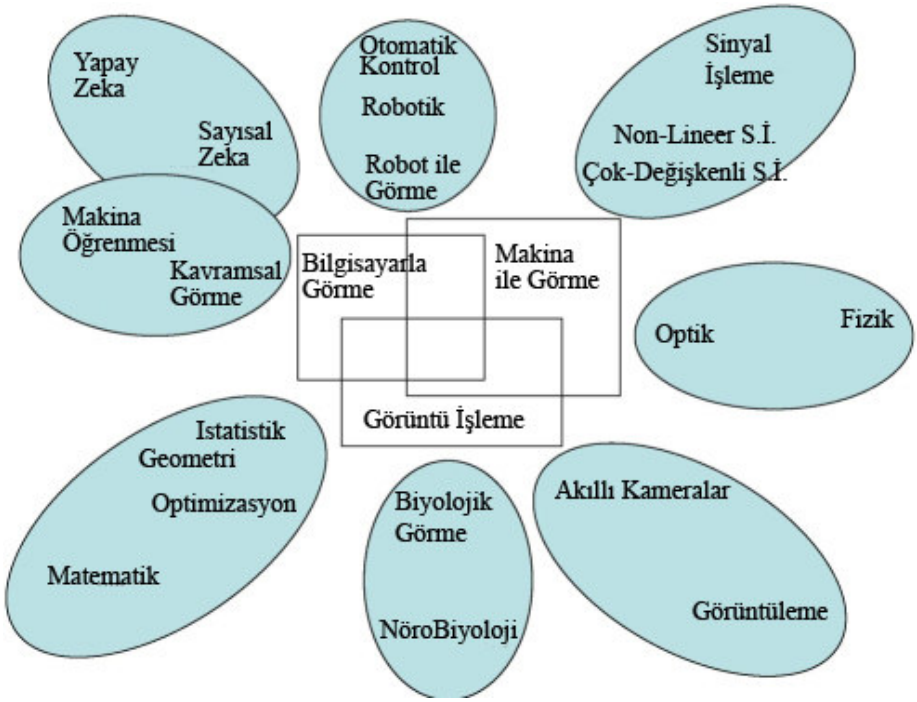
Kalite kontrol uygulamalarında bir parçanın veya ürün alt parçasının kabul edilebilir veya hatalı olduğuna görüntü sistemi ile karar verilerek kontrol düzenekleriyle parçanın kabulü veya iadesi yapılır. Makina yönlendirme uygulamaları robot bileşenlerin veya otomatik malzeme işleme donanımlarının, doğruluğunu ve hızını artırmak amacıyla görüntü sistemleri kullanırlar.

Makina Görüntü Sistemleri, Bilgisayarla Görme sistemlerinin endüstri ve imalat alanlarındaki uygulamasıdır. Bilgisayar görüşü, yapay zeka bir sistemin davranışlarını kontrol etmek üzere resim bilgisinin metin temelli girdilere alternatif olarak bir sisteme beslendiği bir alan olan yapay zekanın bir alt alanıdır. Bilgisayar görüşünde kullanılan bazı öğrenme yöntemleri yapay zeka alanında geliştirilen öğrenme tekniklerine dayanır.

Diğer bir bilgisayar görüşü ile ilgili olan konu sinyal işlemedir. Tipik geçici tek değişkenli sinyali işlemeye yönelik birçok mevcut metot, bilgisayar görüşünde iki veya çoklu değişkenli sinyale doğal bir şekilde uzatılabilir. Fakat görüntünün özel doğasından ötürü, bilgisayar görüşünde tek değişkenli sinyal işlemesinde karşılığı olmayan, geliştirilen bir çok metot vardır. Bu metotların ayırt edici bir karakteri, doğrusal olmayan sinyalin çok boyutluluğu ile birlikte bilgisayar görüşünün bir parçası olan sinyal işlemenin altında bir alt kategori tanımlar. Şekil 1.1'de Görüntü işleme, Bilgisayarla görme ve Makina Görme sistemleri arasındaki ilişki görülmektedir.

Bilgisayar Görme esas olarak makina tabanlı (bilgisayar) görüntü işleme üzerine odaklanırken, makina ile görme, çoğunlukla imalat donanımlarını ve robotik donanımlarını kontrol etmek için sayısal girdi/çıktı donanımlarına ve bilgisayar ağlarına ihtiyaç duyar. Makina Görüntü sistemleri bir konveyör üzerindeki nesneleri sayma, herhangi bir yüzeyde problem olup olmadığını inceleme, seri numarası okuma gibi güç işlemleri yerine getirirler. İnsanların kısa periyotta daha iyi algılama ve sınıflandırma ve arıza, kalite güvence politikalarına adaptasyonda daha büyük esnekliliğe sahip olmasına rağmen, görevli personelin hastalık keyfiyet, dikkat dağılması gibi nedenlerden dolayı daha büyük hata oranına sahip olması, görsel takip

gerektiren 24 saat süreli işlemler, sürekli tekrarlı ölçüm işlemleri, yüksek-hız işlemleri için üreticilerin Makina görüntü sistemlerini tercih etmesinde büyük rol oynamıştır.



Şekil 1.1. Görüntü sistemleri ilişkisi ve uygulama alanları

Makina Görme Sistemlerinin Bileşenleri

Tipik bir Makina Görüntü Sistemi aşağıdaki donanımlar arasından oluşur:

- Bir veya daha fazla dijital veya analog kamera
- Aydınlatma
- Görüntü Yakalama kartı (Görüntüyü sayısallaştırmak için. Çoğunlukla framegrabber kartı olarak bilinir.)
- İşlem ünitesi (Bilgisayar veya sayısal sinyal işleme ünitesi)
- Görüntü işleme yazılımı
- Devindirici (istenen işlemin icrası için)

Görüntü Sistemi Uygulama Alanları

- Büyük ölçekli endüstriyel üretim
- Kısa vadeli özel parça imalatı
- Endüstriyel çevre için güvenlik sistemleri
- İmalat süreç takibi (kalite kontrol, hata belirleme)
- Stok kontrol ve yönetim sistemleri (sayma, barkod okuma)
- Otomatikleştirilmiş Yönlendirilmiş Araçların kontrolü
- Güvenlik ve emniyet için yerleşim yerlerinin otomatikleştirilmiş izlenmesi
- Tarim üretiminin izlenmesi
- Yiyecek ürünlerinin kontrol ve iyileştirilmesi
- Satış otomasyonu
- Tüketici teçhizat kontrolü
- Medikal görüntü işleme
- Medikal çalışmalar için uzaktan inceleme ve işlemleri takip etme

Makina Görüntü Sistemleri yarıiletken donanım fabrikalarında büyük ölçüde kullanılmaktadır. Görüntü sistemleri silikon katmanların, işlemci ciplerinin ve direnç, kondansatör gibi diğer alt parçaların takibinde kullanılmaktadır.

Otomotiv Endüstrisinde makina görüntü sistemleri endüstriyel robotların yönlendirilmesi, metal bölümlerin tam uyuşması, boyanmış yüzeylerde problem olup olmadığının takibi işlemleri için kullanılmaktadır.

Konu ile ilgili Araştırmalar

Görüntü işleme ve nesne tanıma medikalden uzay araçlarına çok çeşitli alanlarda işlenmektedir. Özellikle insanlar için daha zor ve tehlikeli olan, nükleer tesisler, atık madde işleme fabrikaları, medikal alan, tekrar ve süreklilik gereken uygulamalarda görüntü işleme sistemleri tercih edilmekte olduğu bir önceki bölümde ifade edilmişti. Otomotiv endüstrisinde ise malzeme işleme, kaynak yapma, boyama gibi işlemlerde

değişen koşullar ve hatalar karşısında endüstriyel robotların yönlendirilmesi ve hataların giderilmesi işlemlerinde kullanılmaktadır. Görüntü işleme, sadece görünebilir ışık bandında sınırlı kalmayıp, tüm elektromanyetik banttan elde edilen görüntüler (X-ışınları, morötesi, kızılötesi, ultrasound) için de tıp, malzeme, biyoloji, üretim, kalite kontrol, tekstil, uzay ve havacılık, insansız araçlar gibi uygulama alanlarında da kullanılmaktadır.

Xiang Zhang [1] zımparalanmış ve parlatılmış ürün yüzeylerindeki bozuklukların otomatik sınıflandırması üzerine çalışmıştır. Çalışmasında fonksiyonellik ve estetik açıdan yüksek kalitede parçalar imal etmek için parça üzerindeki bozuk bölümü belirleyip bozukluk sınıflandırması yapmaktadır. Çalışmasında bozukluk sınıfları için biçim özellikleri yanında Laws filtreleri, Ayrık Kosinüs dönüşümü (Discrete Cosine Transformation), Gabor filtreleri ve beraber oluşma matrisi tabanlı istatistiksel özellikler denenmiştir. Çoklu-sınıf sınıflandırıcısı olarak Destek vektör makinası (Support Vector Machine) kullanılmıştır. SVM çekirdeği olarak radyal tabanlı fonksiyon kullanmıştır. Gabor fitre özelliklerini ve istatistiksel özellikleri birleştirerek sınıflandırma doğruluğunda % 82 seviyesini yakalamıştır.

Sunil S., PaulW F. [2] yeraltı borularına ait görüntülerin biçimsel-yapısal (morfolojik) bölümlendirme ile sınıflandırmasını yapmışlardır.

Li-Chen Fu Cheng-Yi Liu [3] Otonom araç sürüşü için Bilgisayar görme tabanlı nesne belirleme ve tanıma uygulaması yapmışlardır. Sürüşe kapalı alanda, yaya, araç ve diğer olmak üzere üç farklı engel türü tanımlaması yapılmıştır. Çift görüş sitemi kullanılmıştır. Karşılaştırma işlemi için Kenar belirleme detektörü, Hausdorff mesafe dönüşümü, koordinat belirleme, hiyerarşik kalıp karşılaştırma işlemlerini kullanmıştır.

Mohammed B., ve Boualem B., [4] uygulamalarında otomatik nesne tanıma için yapısal tanımlayıcı tabanlı görüntü sistemi kullanmışlardır. Öncelikle eşdüzey gri bölgeleri çıkarmak için birinci ve ikinci türev kullanan kenar dedektörleri uygulamıştır. İkinci aşamada parça bölümlendirme algoritması kullanılarak nesne

kendini oluşturan eşdüzey çizgileri ile tanımlanmıştır. Sonra modellere ait parametreler elde edilmiştir. Nesneler nesneyi oluşturan bileşenlerin ifadesi için kullanılan ve bu bileşenler arasındaki ilişkiyi (piksel olarak) belirleyen yapısal veri seviyeleri olan yapısal tanımlayıcıyla ifade edilmiştir. Bu ifade şekli nesneleri, onların pozisyon, oryantasyon, büyüklük bilgilerinden bağımsız olarak tanımlanabilmesini sağlamıştır.

Brzakovic ve Vujovic [5] çalışmalarında aynı hastaya ait değişik mamografi görüntülerini detaylı karşılaştırma yaparak kanserli dokudaki değişimleri gösteren bilgisayar tabanlı bir uygulama parçası gerçekleştirmişlerdir. Odak noktası iki mamogram görüntüsünde bir kontrol noktası belirlemektir. Çalışmalarının detayında potansiyel kontrol noktalarını belirleme ve ilgili kontrol noktası setleri arasında ilişki kurma vardır.

Thamsanqa [6] değişik memeli türlerine ait örnek kıl görüntüleri üzerinde, örnek tanıma tekniklerini kullanıp kıl türlerine göre memelilerin sınıflandırmasına çalışmıştır. Çalışmalarında memeliye ait kılın kesit ve boy görüntüsü üzerinde iki boyutlu gabor filtreleri kullararak % 72 doğrulukta sınıflandırma yapmıştır. Sınıflandırma için dışmerkezlilik, yoğunluk, istatistiksel moment özellikleri kullanılmıştır.

Min Young Kim [7] mobil robotlar için insanoğlunun görsel dikkat modelini taklit ederek çevre haritasını kendi kendine çıkaran (seyrüsefer sistemi için pozisyon ve oryantasyon bilgisinin elde edilmesi) bir çalışma yürütmüştür. İnsanoğlunun, çevresi ile ilgili bilgi alırken çevre ile ilgili özellikleri ve dikkat edilmesi gereken şeyleri seçme yeteneğini model olarak kullanmıştır. Robotun en iyi seyrüseferi seçebilmesi için bulanık karar verme metodunu kullanmıştır. Uygulamasında iki CCD kamera ve yapısal lazer ışık kaynağı ile triocular görüntü teorisini kullanmıştır. Christophe D., Gabriel A., Eric O. [8] robotik işlemler için tek gözlü (monocular) görüntü sistemi ile durağan malzemelerin tanınması ve pozisyonlarının belirlenmesi üzerine çalışmışlardır.

Adnan A. Y. Mustafa [9] çalışmalarında nesne tanıma işleminde görüntü, gölgeler veya başka nedenlerle seçilebilir olmadığında, nesnenin sınırı işaretlerini kullanarak karşılaştırma işlemi geliştirmişlerdir. Sınır şekli olarak doğrultu işareti, kavisliliği, uzaklık işareti ve parametre sınır işareti tipleri kullanılarak rasgele doğrultu ve şekillerdeki örnekler için karşılaştırma yapılarak nesneye ait görüntünün en az % 70 belirginse nesnenin orjinal şeklini elde etmişlerdir.

Alessandro L., Cosimo D., Francesco B. [10] takip için video görüntüsündeki gölgelerin elenmesi üzerine bir yaklaşım sunmuşlardır. Hareketli nesnelerin takibinde kullanılan video görüntüsü içerisindeki beklenmedik gölgeler nesnelerin pozisyonlarının yanlış belirlenmesine, bölümlendirilmesinde, ölçümünde takibinde hatalara yol açmaktadır. Uygulamada önce görüntüden geriplan görüntüsü çıkarılmış, kalan görüntü üzerinde doku analizi yaklaşımı uygulanmıştır. Hareketli görüntülerden gölgeleri ayrıştırmak için kullanılan metodunda kovalama (takip) planı baz alınmıştır. Doku kaplamasını büyük bir setin elementlerinin lineer bileşimi gibi ifade etmek için temel fikir, basit fakat güçlü karşılaştırmalı takip algoritması kullanmaktır.

Bayraktar [11] Taşıyıcı bant üzerinde hareket eden nesneleri Bayes ve en yakın komşu tanımlayıcıları kullanarak konumlandıracak ve sınıflandıracak bir çift görüş sisteminin geliştirme uygulaması gerçekleştirmişlerdir. Nesnelerin konumlarını tahmin edebilmek için, farklı konumlara yerleştirilmiş iki kamera ve çift eşleştirme algoritması kullanılmıştır.

Arıcan [12] Yer belirleme için görme tabanlı iki robot yeri belirleme algoritması hayata geçirilmiştir. Birinci algoritma, yapay yer işaretlerine göre pozisyonun ölçülüp, bulunduğu ortamdaki yerini bulmaya yöneliktir. Bu yapay yer işaretlerinin yerleri bilinmektedir. İkinci algoritma ise yapay yer işaretleri kullanmadan ortamda doğal olarak bulunan nesnelerin robota göre pozisyonu, robota stereo kamera sistemi ile bildirilmiştir.

Fumi H., Masayoshi H., Shinji K., Tsugito M. [13] uygulamalarında üzerinde robot manuplator bulunan yemek tablası taşıyan otonom bir robot için pozisyon belirleme ve tanıma işlemlerinde çift görüş sistemi ve kenar belirleme algoritmalarını kullanmışlarıdır.

Önder [14] yüz saptama ve izleme yeteneğine sahip bir robot uygulaması yapıştır. Bu yüzden tez iki temel çalışma içermektedir: İlk olarak robot üzerindeki kameradan alınan görüntü üzerindeki insan yüzü içeren noktaların belirlenmesi için gerçek-zamanlı bir yüz saptama algoritması kullanılmıştır. İkinci iş olarak da robotun bulunan bu yüze doğru ilerlemesi gerçekleştirilmiştir.

Şeran [15] görüntü yönlendirmeli bir robotik montaj sisteminin temel geometrilere sahip parçaları, uygun şablonlara monte üzerine tasarımı ve uygulama çalışma gerçekleştirmiştir. Çalışmasında Stereo görüntüleme, üç boyutlu konum algılama ve nesne tanımlama tekniklerini kullanmıştır. Bu tez çalışmasında, "elde göz" kurulumunu kullanan bir sistem ele alınmıştır. Robot açık-döngü kontrolü prensiplerini kullanan bir görüntü işleme sistemi tarafından kontrol edilmiştir.

Ünal [16] seri üretim hatlarında görüntü işlemeyle kalite kontrolü uygulaması gerçekleştirmiştir. Kameradan alınan görüntüler, görüntü işleme yazılımı ile düzenlendikten sonra değişik geometrilere sahip parçaların otomatik veya manuel ölçümleri alınmıştır. Bu ölçümlere göre malzeme kalite kontrolü uygulanmıştır.

Yaman [17] görüntü işleme tekniklerini kullanarak Ankara hızlı raylı sistem güzergahında sefer aralıklarının optimizasyonuna yönelik olarak incelenmesi çalışması yapmışlardır. Uygulamasında kameralardan alınan görüntülerden istasyondaki yolcu yoğunluğunu belirleyip, bu bilgiyi sefer araklıklarını düzenlemek için kontrol bilgisi olarak kullanmışlardır.

Bu çalışmanın amacı; Bir malzemeye ait görüntüleri üzerinde görüntü işleme teknikleri kullanılarak malzeme özelliklerinin elde edilmesidir. Örnek nesne olarak domino taşları kullanılmıştır ve bu domino taşlarının değişik düzendeki dizilişleri ve

değişik çözünürlükteki görüntüleri için tanıma işlemi gerçekleştirilmiştir. Bu işlemleri yapan uygulama yazılımı daha sonrasında malzeme görüntüleri için yeniden düzenlenerek malzeme içindeki grafit tane büyüklükleri elde edilmiştir. İşlemler gerçek zamanlı yapılmamıştır.

Bu çalışmanın kapsamı; Bu çalışmada kamera ile alınan domino taşları görüntüsü üzerinde görüntü işleme teknikleri kullanılarak taşlar ve geriplan ayrıştırılıp sonra her bir taş teker teker otomatik olarak bilgisayar tarafından tanımlanmıştır. İşlemler gerçek zamanlı yürütülmediğinden her resim ayrı ayrı alınıp işlenmiştir. Taşların düzenli ve rasgele yerleşim durumları için değişik çözünürlükte görüntüler alındı. Sonrasında taşların üst ve alt bölümleri belirlenerek üstteki ve alttaki nokta sayısı ayrı ayrı elde edilmiştir. Sonraki aşamada ise işlemler malzeme görüntüleri üzerine uygulanarak görüntü içerisindeki grafit tanelerinin alanları ve resim içerisindeki yüzdesi belirlenmiştir. Sonuçların hepsi tekrar çağırılabilecek şekilde tüm aşamalarıyla bilgisayar ortamına kaydettirilmiştir.

Bu kapsamda ilk bölümde konu genel olarak tanımlanıp ve konuyla ilgili çalışmaların özeti verilmiştir.

İkinci bölümde görüntü ve resim formatı başlığı altında, görmenin fizyolojisi, bilgisayarla görme ve görüntünün elde edilmesi, örnekleme ve miktarlamanın yapısı, CCD kamera ve elde edilen görüntüdeki yapı elemanları (pikseller) arasındaki komşuluk ilişkisi anlatılmıştır.

Üçüncü bölümde resim tanım bölgesinde resmin iyileştirilmesi ve hazırlanması işlemleri olan, gri seviye dönüşümleri, histogram işleme, görüntü tanım bölgesinde filtreleme işlemleri ve bu işlemlerin geriplanı anlatılmış örnek görüntüler üzerinde işlemler gösterilmiştir.

Dördüncü bölümde görüntü biçimsel-yapısal işleme teknikleri ikili düzey ve gri düzey görüntü için verilmiştir.

Beşinci bölümde görüntü analizi için uygulanan teknikler sunulmuştur. Bu kapsamda bölümlendirme algoritmalarından kenar belirleme, eşikleme, bölgesel bölümlendirme,watershed bölümlendirme, nesne özellikleri çıkarma, sınıflandırma metotlarından karar tabanlı ve optimum istatistik tabanlı olanlardan, karşılaştırmalı ve bayes sınıflandırıcı anlatılmıştır. İşlemler örnek görüntüler üzerinde gösterilmiştir.

Altıncı bölüm renkli görüntü işleme tekniklerini içermektedir. Bu uygulamada renkli görüntü işleme teknikleri kullanılmamıştır.

Yedinci bölümde kullanılan sistem, yapılan uygulama ve oluşturulan yazılım hakkında bilgiler verilmişdir. Birinci ve ikinci bölüm uygulama çalışmasına ait işlemler domino taşları ve küresel grafitli malzeme görüntüsü üzerinde gerçekleştirilip elde edilen sonuçlar sunulmuştur.

Son bölümde ise yapılan çalışmanın analizi ve sonuçları irdelenerek yapılabilecek işlemlerden bahsedilmiştir.

2. GÖRÜNTÜ VE SAYISAL GÖRÜNTÜNÜN YAPISI

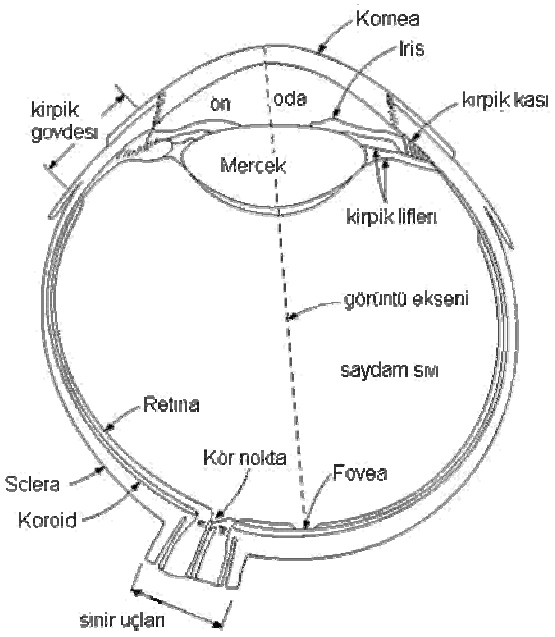
2.1 Görsel Algılamanın Temeli

Sayısal görüntü işleme alanı, matematiksel ve olasılık formülasyon temelleri üstüne kurulmuştur. Bir tekniğin diğerine karşı seçiminde insan sezgisi ve analizi önemli rol oynamaktadır ve bu seçim çoğu defa kişisel, görsel hükümlere dayanmaktadır. Bu nedenle, ilk adım olarak insan görsel algısının basit bir anlayışını tanımlamak uygundur. Bu konunun karmaşık ve geniş olmasından dolayı özellikle görüntünün gözde nasıl oluştuğu ile ilgili mekanizma ve parametreler üzerinde durulmuştur. Aynı zamanda sayısal görüntü ile birlikte bu çalışmada kullanılan faktörler açısından insan görüşünün fiziksel kısıtlamaları verilmiştir.

2.1.1 İnsan gözünün yapısı

Şekil 2.1 insan gözünün basitleştirilmiş yatay kesitini göstermektedir. Göz, ortalama çapı yaklaşık olarak 20mm olan bir küre şeklindedir. Gözü 3 zar kuşatmaktadır: kornea ve göz akı dış kısım; koroid; ve retina. Kornea gözün ön yüzeyini kuşatan sert saydam bir dokudur. Göz akı optik kürenin kalanını kuşatan saydam olmayan bir zardır.

Koroid doğrudan göz akının altında yer alır. Bu zar, gözün en önemli beslenme kaynağı olan kan damarları ağını içerir. Koroidde oluşacak çoğu zaman önemli olmadığı düşünülen yüzeysel hasar bile, kan akışını sınırlayan iltihaplanmaya neden olarak ciddi göz hasarlarına yol açabilir. Koroidin tabakası çok pigmentlidir, böylece dışardan göze gelen ve optik kürede yayılan ışığın miktarını azaltmaya yardımcı olur. En uç noktasında, koroid kirpiksi bölüm ve iris diyaframı olarak iki kısma ayrılır. Sonraki büzülmeler veya genişlemeler göze giren ışığın miktarını kontrol etmek içindir. İrisin merkez açıklığı (göz bebeği) yaklaşık olarak 2-8 mm arasında değişir. Arka tarafı siyah pigment içerirken, irisin ön tarafı gözün görünen pigmentini içerir.

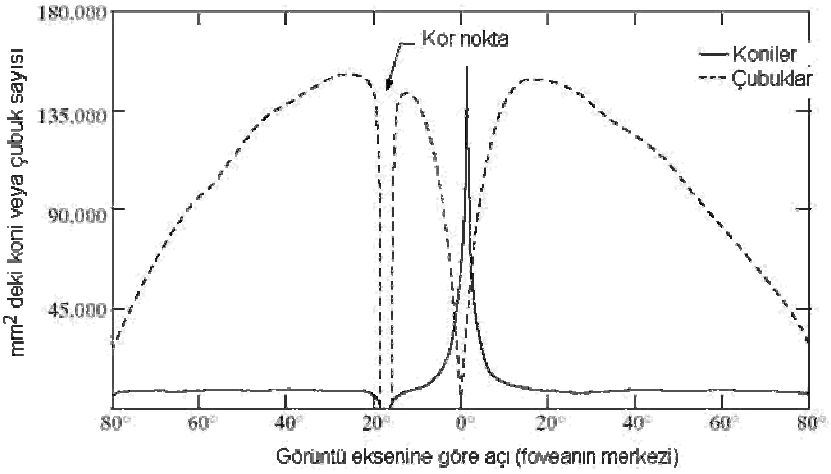


Şekil 2.1. Gözün yapısı [18]

Göz merceği lifli hücrelerin eşmerkezli katmanlarından oluşmaktadır ve kirpiksi kısma bağlanan lifler tarafından geçici olarak kapatılır. Göz merceği %60-70 oranında su, yaklaşık olarak %6 oranında yağ ve gözdeki diğer dokulardan daha fazla protein içerir. Göz merceği daha kısa dalga boyunda daha yüksek emilim oranına nispeten yaklaşık olarak görünebilir ışık spektrumunun %8'ini emer. Hem infrared hem de ultraviyole ışıklar mercek yapısındaki proteinler tarafından emilir. Fazla miktardaki infrared ve ultraviyole ışıklar göze zarar verebilir.

Gözün en içteki zarı duvarın gerideki tüm kısmında uzanan retinadır. Göz uygun şekilde odaklanlığında, gözün dışındaki nesneden gelen ışık retina üzerinde görünür. Model görüşü ayrı ışık alıcılarının retina yüzeyine dağılması ile yapılır. İki tür alıcı vardır; koni ve çubuk. Herbir gözdeki konilerin sayısı 6-7 milyon arasındadır. Retinanın öncelikle fovea denen merkez kısmında yerleşmişlerdir ve renge çok duyarlıdırlar. İnsanlar büyük ölçüde bu konilerle ince ayrıntıları çözebilirler, çünkü her biri kendi sinir ucuna bağlıdır. Gözü kontrol eden kaslar ilgili nesnenin görüntüsü foveaya düşene kadar göz küresini döndürür. Koni görüşüne fotopik veya parlak-ışık görüşü denir.

Çubukların sayısı daha fazladır: Yaklaşık olarak 70-150 milyon retina yüzeyi boyunca dağılmıştır. Dağılımın geniş bölgede olması ve bazı çubukların tek bir sinir ucuna bağlı olması gerçeği bu alıcılar tarafından görülebilir ayrıntı miktarını azaltmaktadır. Çubuklar görüş alanının genel, tüm bir resmini verme işini yaparlar. Renk görüşüne karışmazlar ve düşük seviyedeki aydınlanmalara karşı duyarlıdırlar. Örneğin, gün ışığında parlak renkli gözüken nesneler ayışığında renksiz gözükürler çünkü sadece çubuk alıcılar uyarılmıştır. Bu durum skotopik veya loş-ışık görüşü olarak bilinir[18].



Şekil 2.2. Görüntü açısı – koni, çubuk sayısı grafiği

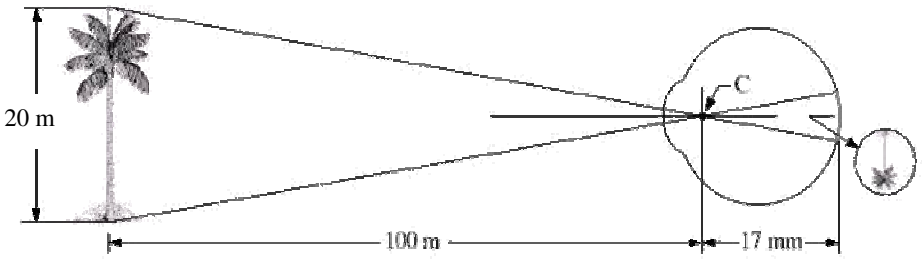
Şekil 2.2 optik sinirin çıkış alanı boyunca geçen çubukların ve konilerin yoğunluğunu göstermektedir [20]. Bu bölgede alıcıların yokluğu kör nokta denen kısımları meydana getirir. Bu bölgenin dışında, alıcıların dağılımı foveanın etrafında radyal olarak simetriktir. Alıcı yoğunluğu, foveadan derece cinsinden ölçülür (bu, eksenler arasındaki derece, göz merceğinin merkezinden geçen ve retinayı kesen bir çizgi ile görüş ekseni arasındaki açının ölçülmesidir.) Şekil 2.2'den konilerin retinanın merkezinde (foveanın merkez bölgesinde) daha yoğun oldukları görülmektedir. Ayrıca çubuk alıcıların merkezden yaklaşık 20º dışarda yoğunluklarının arttığı ve retinanın sınır çevresinde yoğunluklarının azaldığı görülmektedir.

Fovea, retina içerisindeki yaklaşık 1.5mm çapındaki dairesel bir girintidir. Böylece, foveayı 1.5mm x 1.5mm boyutunda bir kare algılayıcı dizi olarak görmek mümkündür. Retina alanındaki konilerin yoğunluğu yaklaşık olarak 150000 element/mm^2'dir. Gözdeki bu bölgede bulunan konilerin sayısı yaklaşık olarak 337000 elementtir. Şimdi, ham çözme gücü cinsinden orta çözünürlükte cip görüntüleme yapan şarj-birleşik cihazı (charge-coupled device) alıcı dizisindeki elemanların sayısını 5mm x 5mm'den daha fazla elde edemez.

2.1.2 Gözde görüntünün oluşumu

Göz merceği ile sıradan bir optik mercek arasındaki en temel fark göz merceğinin esnek olmasıdır. Şekil 2.1'de göz merceğinin ön yüzündeki kavis yarıçapı gerideki yarıçapından daha büyük olduğu görülmektedir [18]. Göz merceğinin yapısı, kirpiksi bölgedeki liflerin gerilimi ile kontrol edilir. Uzak nesnelere odaklanmak için, kontrol kasları göz merceğinin nispeten yassılaşmasına sebep olurken, gözün yakınındaki nesnelere odaklanda, bu kaslar göz merceğinin kalınlaşmasına sebep olur.

Göz merceklerinin merkezi ile retina arasındaki mesafe (odaksal uzaklık denir) göz merceğinin kırılma kapasitesi en küçükten en büyüğe artarken yaklaşık olarak 17mm ile 14mm arasında değişir. Göz, yaklaşık 3m'den daha uzak bir nesneye odaklanırken, göz merceği en düşük kırılma kuvvetini gösterir. Göz, yakınındaki bir nesneye odaklanırken, göz merceği en kuvvetli kırılmasını sergiler. Bu bilgi bir nesnenin retinal görüntüsünün büyüklüğünü hesaplamayı kolaylaştırır.



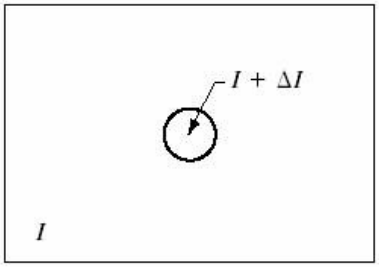
Şekil 2.3. Bir nesne görüntüsünün gözde oluşumu. (C göz merceği merkezi)

Şekil 2.3'te örneğin, gözlemci 20 m yüksekliğindeki ve 100 m uzaklıktaki bir ağaca baktığında eğer nesnenin retinal görüntüsü mm cinsinden h yüksekliği ile ifade edilirse, Şekil 2.3'ün geometrisi, 20/100=h/17 ve h=3,4 mm vermektedir. Retinal görüntü öncelikle fovea alanına yansır. Daha sonra algılama, ışın enerjisini beyin tarafından çözülen elektrik darbe sinyallerini çeviren ışık algılayıcıların bağıl uyarımı ile meydana gelir.

2.1.3 Parlaklık adaptasyonu ve ayrımı

Sayısal görüntünün ayrı ışık şiddeti kümeleri olarak görüntülenmesi sebebiyle, gözün farklı ışık şiddeti seviyelerini ayırma yeteneği görüntü işleme sonuçlarını sunmak açısından önemli bir etkendir. İnsan görüş sisteminin adapte olabildiği ışık yoğunluk seviyeleri aralığı düşük ışık eşiğinden parlak seviyeye kadar çok geniştir**.** Deneysel tecrübeler öznel parlaklığın (insan görüş sistemi tarafından algılanan yoğunluk), göze düşen ışık yoğunluğunun logaritmik bir fonksiyonu olduğunu göstermiştir.

Gözün herhangi bir özel adaptasyon seviyesindeki ışık yoğunluğu değişimleri arasındaki ayırımı yapma yeteneği de önemli bir ilgi alanıdır. İnsan görüş sisteminin parlaklık ayrımı kapasitesini tespit etmek için yapılan klasik deney, tüm görüş alanını kaplayacak şekilde aynı oranda aydınlatılmış yeterince büyük düz bir alana öznel bakış yapmaktır. Bunun için gerekli kurulum şekil 2.4 de görülmektedir [18].

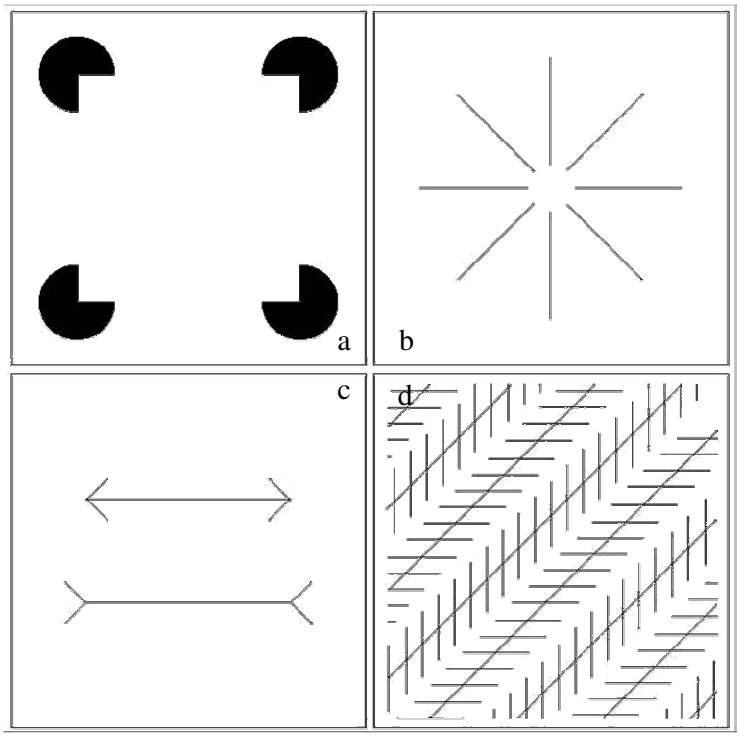


Şekil 2.4. Parlaklık farkını algılamak için kullanılan düzenek

Bu alan tipik olarak, yoğunluğu (I) değişebilen bir ışık kaynağının bakanın arka tarafından aydınlatıldığı donuk cam gibi bir yayıcıdır. Uniform bir şekilde I

aydınlatma şiddeti ile aydınlatılmış alanın merkezinde daire olarak görünen ışıltıya kısa-süreli ΔI kadar aydınlatma artışı eklendiğinde eğer ΔI yeterince parlak değilse, algılanabilir değişiklik farkedilmez. Eğer ΔI daha artırılırsa değişiklik algılanabilir seviyeye gelecektir. Sonunda, ΔI yeterince kuvvetli olursa, her zaman fark edilir halde kalır. $\Delta I_c/I$ büyüklüğüne *Weber oranı* denir, burada ΔI_c zemin aydınlanması I'nın %50 katı ayırt edilebilir aydınlanma artışıdır. Küçük bir $\Delta I_c/I$ değeri, aydınlanma şiddeti yoğunluğunda küçük orandaki değişimin ayırt edilebildiği anlamına gelmektedir. Bu "iyi" parlaklık ayrımını gösterir. Tersine, büyük $\Delta I_c/I$ değeri yoğunlukta büyük oranda değişim gerektiği anlamına gelmektedir. Bu da "zayıf" parlaklık ayrımını gösterir [18].

İnsan algılama olağanüstülüğünün diğer örnekleri, gözün olmayan bilgileri doldurduğu veya nesnelerin geometrik şekillerinin yanlış algılandığı optik yanılsamalardır. Bazı örnekler Şekil 2.5'da verilmiştir. Şekil 2.5(a)' da kareyi



Sekil 2.5. Bazı optik yanılsamalar

tanımlayan hiçbir çizgi görüntünün bir parçası olmadığı halde, karenin çerçevesi net olarak görülür. Aynı etki, Şekil 2.5(b)'de daire olarak görülebilir, sadece bir kaç çizginin nasıl tüm daire yanılsamasını vermekte yeterli olduğu belirgindir. Şekil 2.5(c)'de görülen iki yatay çizgi parçası uzunluk olarak aynıdır, ancak bir tanesi diğerinden daha kısa gözükür. Son olarak, Şekil 2.5(d)'deki 45º açıyla yerleşmiş tüm çizgiler eş uzaklıkta ve paraleldir. Yine kesişen diğer kısa çizgiler, çizgilerin paralel olmadığı ve birbirlerinden uzaklaşıyormuş gibi bir yanılsama yaratır. Optik yanılsamalar insan görüş sisteminin tam olarak anlaşılamamış bir karakteristiğidir.

2.2 Bilgisayar Ortamında Görüntü Yapısı

Bilgisayar ortamında bir görüntü, ışık şiddetin iki boyutlu ifadesidir. Bir görüntü elde etmede yapılan işlem, üç boyutlu ortamın iki boyutlu resim düzlemi üzerine izdüşümünü almaktır. Resim düzlemindeki her nokta, bir renk değerine ve resim düzlemi koordinatına sahiptir. Bir resmin yapısını anlamak için üç boyutlu ortam koordinatları ile resim düzlemi koordinatlarındaki ilişki ile bir noktadaki ışık seviyesinin nasıl belirlendiğini bilmek gerekir.

2.2.1 Kamera projeksiyon modeli

Kamera Projeksiyon Modelleri üç boyutlu ortam noktasının, resim düzlemi üzerindeki yerini belirlemek için kullanılır. Perspektif izdüşümde, resim düzleminin sabit bir mesafe önünde bir küçük-delik (pinhole) olduğu ve ışık ışını sadece bu delik den geçerek resim düzlemine ulaşabileceği kabul edilir. Böylece ışık bu noktadan düz bir çizgi şekilde geçerek, resim düzlemindeki nokta ile ortam noktası arasında ilişki belirlenir. Burada öncelikle Görüntü işlemede kullanılan koordinat sistemlerini sunmak faydalıdır. Bunlar:

Görüntü (piksel) Koordinatları : resim elementleri için kullanılan grid koordinattır (2-boyutludur).
Görüntü düzlemi koordinatları: Görüntülenen ortam noktalarını resim düzlemi üzerine (CCD) 2 boyutlu izdüşüm koordinatlarıdır.

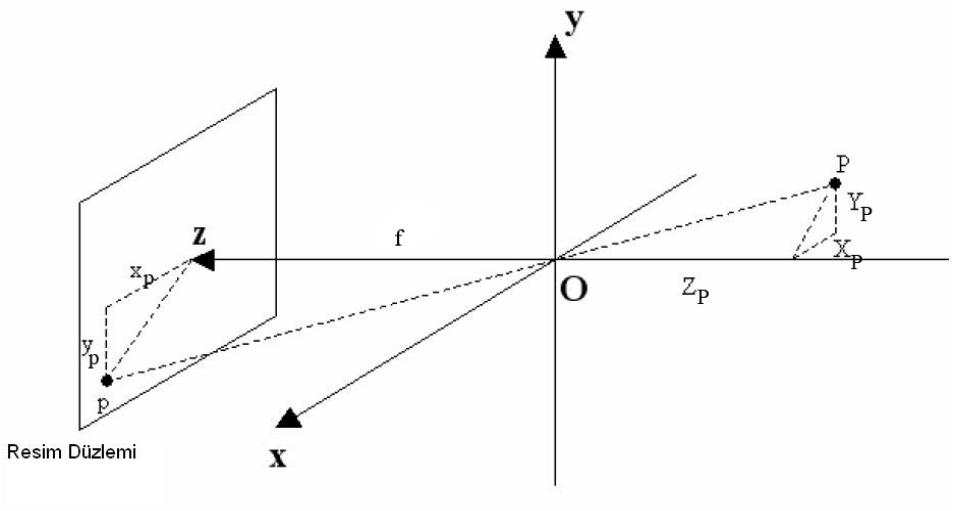
Kamera Koordinatları :Ortam noktalarının görüntüleyen merkezli ifadesidir.
Mutlak (Gerçek) Koordinatları: ortam noktalarının ortam merkezli 3 boyutlu ifadesidir.

Şekil. 2.6'da orjini pinhole olan kartezyen koordinat sistemi görülmektedir [11]. Burada z- ekseni resim düzlemine diktir ve optik eksen olarak da bilinir. Burada amaç P noktasının resim içindeki yerini bulmaktır. Şekil 2.6'da üçgenlerin benzerliğinden faydalanılarak resimde p noktasının yeri Eş. 2.1 ve Eş. 2.2 ile hesaplanabilir.

$$\frac{x_p}{f} = \frac{X_P}{Z_P} \tag{2.1}$$

$$\frac{y_p}{f} = \frac{Y_P}{Z_P} \tag{2.2}$$

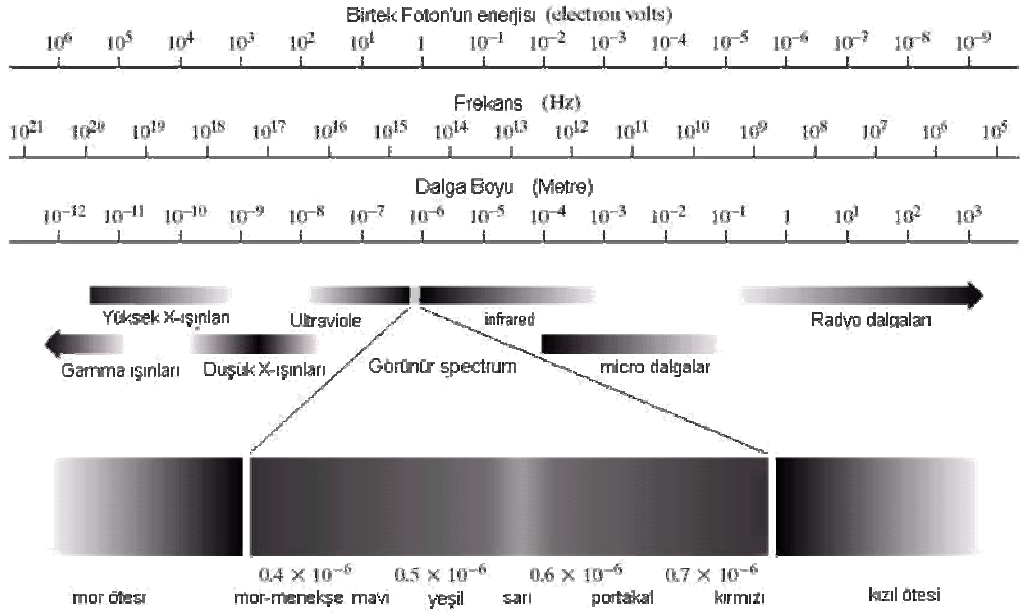
Burada **f** resim düzleminin orjinden (pinhole) uzaklığı ve (x_p, y_p) resim düzleminde p noktasının koordinatlarıdır.



Şekil 2.6. Pinhole kamera projeksiyon modeli

2.2.2 Işık ve elektromanyetik spektrum

1666 yılında Newton, güneş ışını cam prizmadan geçtiğinde görünen ışının beyaz olmadığını fakat bunun yerine bir ucu mor bir ucu kırmızı olan aralıksız bir spektrum içerdiğini keşfetmiştir. Şekil 2.7'de görüldüğü gibi görünen ışıkta algılanabilen renk aralığı elektromanyetik spektrumun küçük bir kısmını göstermektedir.



Şekil 2.7. Tüm elektromanyetik spektrum ve görünür spektrum

Spektrumun bir ucunda dalga boyu görünen ışıktan milyar kadar daha uzun olan radyo dalgaları vardır. Spektrumun diğer ucunda dalga boyu görünen ışıktan milyon kadar daha küçük olan gama ışınları vardır. Elektromanyetik spektrum dalga boyu, frekans ve enerji cinsinden ifade edilebilir. Dalga boyu (λ) ve frekans (ν) arasında ilişki aşağıdaki gibidir:

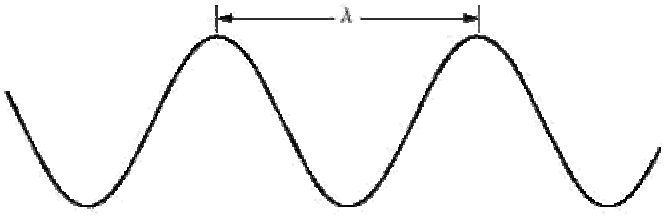
$$\lambda = \frac{c}{\nu} \tag{2.3}$$

Burada c ışık hızını ifade etmektedir (2.998×10^8 m/s). Elektromanyetik spektrumun enerjisi aşağıdaki ifadede verilmiştir:

$$e=hv \tag{2.4}$$

Burada *h* Planck sabitidir. Dalgaboyu metre cinsindendir, micron (μm'yi gösterir ve 10^{-6}'ya eşittir) ve nanometre sıkça kullanılır. Frekans Hertz (Hz) cinsinden ölçülür, bir Hertz sinüs dalganın saniyedeki döngüsüne eşittir. Elektrik Enerji için kullanılan en genel birim elektron-volt'tur.

Elektromanyetik dalgalar, Şekil 2.8'de görüldüğü gibi dalga boyu λ olan sinüs dalgalar şeklinde canlandırılabilir veya dalgaya benzer yapıda ve ışık hızında hareket eden kütlesiz parçacıkların akışı olarak düşünülebilir.



Şekil 2.8. Elektromanyetik dalga formu ve dalgaboyu

Her bir kütlesiz parçacık belirli miktarda (veya yığın) enerji içerir. Her bir yığının enerjisine photon denir. Eş. 2.4'den görüldüğü gibi enerji frekans ile orantılıdır, bundan dolayı yüksek frekanslı (kısa dalga boylu) elektromanyetik olgu her bir photonda daha fazla enerji taşır. Bu yüzden, radyo dalgaları düşük enerjili photonlara sahiptir, mikrodalgalar radyo dalgalarından daha fazla enerjiye sahiptir, kızılötesi hala daha fazla, sonra, görülebilir, ultraviyole, X-ışınları ve son olarak gama ışınları hepsiden daha fazla enerjiktir. Bu sebepten dolayı, gama ışınları yaşayan canlılar için çok tehlikelidir.

Işık, insan gözü tarafından algılanabilen ve görülebilen elektromanyetik yayılmanın özel bir tipidir. Görülebilir (renk) spektrum, görünür specturum Şekil 2.7'de genişletilmiş biçimde gösterilmektedir. Elektromanyetik spektrumun görülebilir bandı, yaklaşık olarak 0.43 μm (mor) ile 0.79 μm (kırmızı) aralığında uzanmaktadır.

Kolaylık için, renk spektrumu 6 geniş alana bölünmüştür: mor, mavi, yeşil, sarı, turuncu ve kırmızı.

Renkten yoksun ışık, renksiz (achromatic) veya tek renkli (monochromatic) ışık olarak adlandırılır. Bu tür ışığın tek özelliği yoğunluğu veya miktarıdır. Gri seviye terimi genelde tek renkli yoğunluğu tanımlamak için kullanılır, çünkü siyahtan griye ve son olarak beyaza değişir. Renkli ışığın elektromanyetik enerji spektrumu, yaklaşık olarak 0.43-0.79 μm arasında uzanır. renkli ışık kaynağının kalitesini tanımlamak için 3 temel büyüklük kullanılır:

Işıma veya Aydınlık (Radiance) : ışık kaynağından akan toplam enerji miktarıdır ve genelde watt (W) cinsinden ölçülür.

Parlama veya Parıldama (Luminance): lümen (lm) cinsinden ölçülür ve gözlemcinin ışık kaynağından algıladığı toplam enerji miktarının ölçümünü verir. Örneğin kızıl-ötesi bölgesinde ışınım yapan bir kaynaktan yayılan yüksek enerjili bir ışını, gözlemci zor algılar veya algılayamaz. Bu kaynağın Işıma' sı çok yüksek olsa bile gözlemci için Parıldama'sı nerdeyse 0' dır.

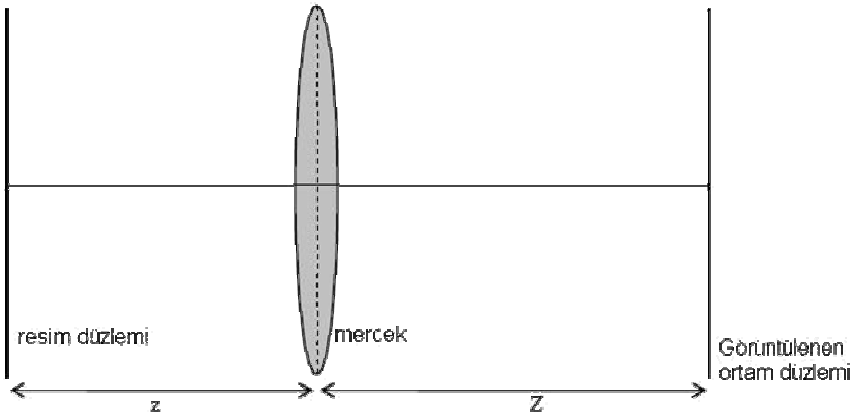
Parlaklık (Brightness) : ölçümünün pratik olarak imkansız olduğu ışık algılayışının öznel tanımlayıcısıdır. Bu ışık şiddetinin renksiz kavramını belirten ve renk hissini tanımlayan anahtar faktörlerden biridir. Parlaklık için gri seviye değeri, resim ışık şiddeti terimleride eşdeğeri olarak kullanılır.

Perspektif Projeksiyon Modelinde görüntülenen ortamdaki bir noktadan gelen ışığın, daha fazla küçülemez (pinhole) bir delikten geçerek resim düzlemine ulaştığı kabul edilmişti. Fakat bu durumda resim düzlemi üzerine çok az ışık miktarı düşmüş olur. Dolayısıyla resmi elde etmek için kullanılan ışık algılayıcısına yeterli miktarda ışık gelmemiş olur. Bu durum, perspektif izdüşümün temeliyle çelişse de daha büyük bir açıklık kullanma zorunluluğunu doğurmuştur.

Daha büyük bir açıklık ve daha fazla ışık almak için mercekler kullanılmaktadır. Bir mercek görüntülenen ortamdaki bir noktadan gelen ışığı, resim düzlemindeki bir noktaya odaklar. Bu resim noktası perspektif izdüşüm ile bulunabilir.

Genel Mercek denklemi Eş. 2.5' de verilmiştir. Şekil 2.9'da da görüldüğü gibi z resim düzlemi ile mercek merkezi arasındaki mesafe, Z ise görüntülenen ortam noktası ile mercek merkezi arasındaki uzaklıktır. f merceğin odak uzaklığıdır. Z ve z mutlak değerleri ile kullanılırlar [11].

$$\frac{1}{|z|} - \frac{1}{|Z|} = \frac{1}{f} \tag{2.5}$$

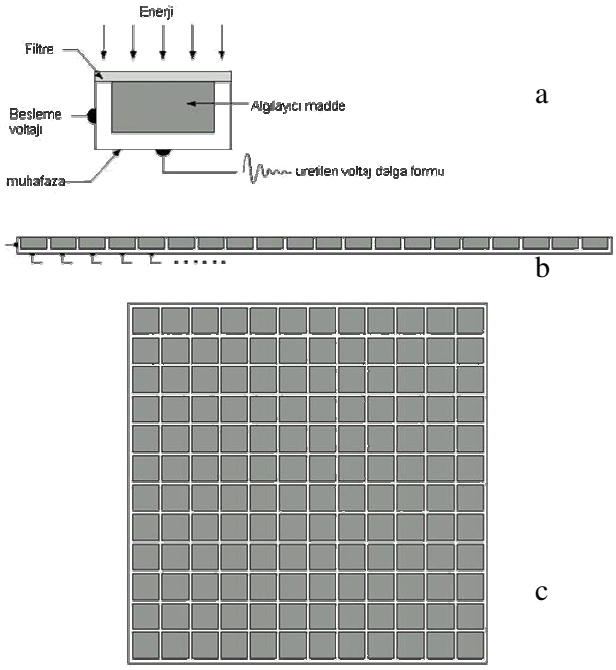


Şekil 2.9. Mercek

Eğer resim düzlemi mercek merkezinden z kadar uzakta ise ve sadece mercek merkezinden Z kadar uzaktaki noktalar tam olarak odaklanmış olur. Eğer görüntülenen ortam noktaları tam olarak odaklanırsa, o zaman resin düzlemi üzerinde ilgili ortam noktası için tek nokta oluşur. Yoksa resim üzerinde nokta yerine daireler oluşturur ki buda merceğin odaklanmamış olduğunu gösterir. Eğer oluşan dairenin çapı, görüntüleme donanımının çözünürlüğünün altında ise odaklanmama kabul edilebilir. Çözünürlük, görüntüleme donanımının, birbirine yakın noktaları ayırt edebilme kabiliyeti olarak düşünülebilir. Burada görüntüleme donanımının çözünürlüğü ile piksel çözünürlüğü ifade edilmektedir.

2.3 Görüntü Algılama ve Elde Etme

Genel uygulamalarda elde edilecek görüntüler, aydınlatma kaynağı ve bu kaynaktan yayılan enerjinin görüntülenen ortam elemanları tarafından yansıtılması veya emilmesinin birleşiminden oluşur. Kaynağın doğasına bağlı olarak, aydınlanma enerjisi nesneden yansır veya nesnenin içinden iletilir. İlk kategoriye örnek bir düzlemsel yüzeyden yansıyan ışıktır. İkinci kategoriye örnek, teşhise ait X-ışını filmi oluşturma amacıyla hastanın vücudundan geçen X-ışınlarıdır. Bazı uygulamalarda, yansıyan veya iletilen enerji, enerjiyi görülebilir ışığa çeviren foto çevirici üzerine odaklanır (örneğin fosfor ekran). Elektron mikroskop ve gama görüntülemenin bazı uygulamaları bu yaklaşımı kullanır.



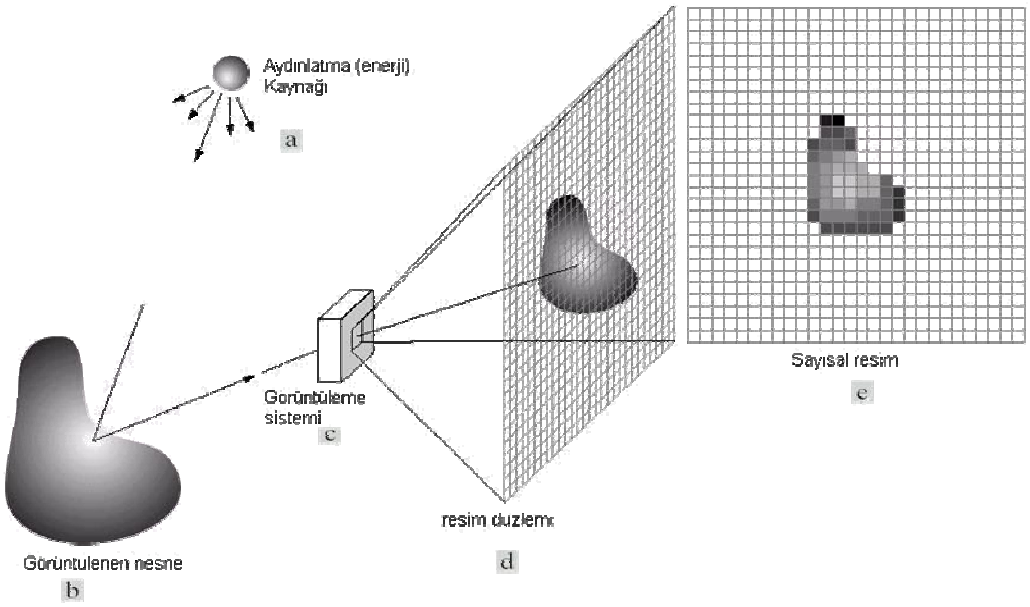
Şekil 2.10. Piksel sensoru ve sensor dizilimleri

Şekil 2.10, aydınlanma enerjisini sayısal görüntüye dönüştürmek için kullanılan üç ana algılayıcı yerleşimini göstermektedir [18]. Basitçe gelen enerji, giriş elektrik gücü ve sezilen özel tip enerjiye duyarlı algılayıcı malzemenin birleşimi voltaja

dönüştürülür. Çıkış voltajı dalga şekli algılayıcıların tepkisidir ve tepkisini sayısallaştırarak herbir algılayıcıdan elde edilen bir niceliktir.

2.3.1 Algılayıcı dizileri (sensor arrays) kullanarak görüntü elde etme

Şekil 2.10 (c)' de ayrı ayrı algılayıcıların 2-boyutlu dizi halindeki biçimi görülmektedir. Pek çok elektromanyetik ve bazı ultrasonik algılama donanımları sıklıkla bu şekilde biçimlendirilirler. Keza dijital kameraların büyük bir kısmıda bu dizilim biçimini kullanırlar. Bu kameralar için tipik algılayıcılar, geniş bir algılama aralığına sahip ve kabaca 4000 X 4000 veya daha fazla algılama elemanından oluşan CCD (Charge Coupled Device) dizilerdir. Herbir sensörün cevabı, sensör elemanın yüzeyine düşen ışık enerjisinin toplamının oranıdır.



Şekil 2.11. Algılama elemanı ile sayısal resmin elde edilmesi

Şekil 2.11'de bir nesnenin görüntüsünün CCD eleman ile elde edilmesi görülmektedir [18]. Burada ışık kaynağından gelen ışık enerjisi nesneden yansıdıktan sonra mercekten geçerek CCD eleman tarafından toplanmakta ve resim düzlemi üzerine odaklanmaktadır. Her sensor üzerine düşen ışık enerjisine göre oransal bir

çıktı üretilmektedir. Bu üretilen çıktı video sinyaline dönüştürülmekte sonrada örnekleme ve miktarlama ile dijital görüntüye dönüştürülmektedir. Bu yapının temel avantajı komple resmin kolayca elde edilebilmesidir.

2.3.2 Kameralar

Kameralar ortamdan gelen ve mercekten geçen ışık ışınlarını görüntü sensorları vasıtasıyla video sinyaline dönüştüren donanımlardır. Her kamerada ışığa duyarlı algılayıcılar dizisi mevcuttur. "Resim Düzlemi" terimi ile ifade edilen de aslında görüntü sensor düzlemidir. Piyasadaki en çok kullanılan sensörler ;

- Vidicon
- CID & MOS (Charge Injection Device and Metal-Oxide Semiconductor)
- CMOS (Complementary Metal-Oxide Semiconductor)
- CCD (Charge Coupled Device)

Bunların arasında, CCD sensör dizileri Görüntü Sistemleri uygulamalarında en çok kullanılanlarıdır. Bu uygulamada da CCD algılayıcılı bir kamera kullanılmıştır. CCD teknolojisi öncelikli gereklilik görüntü kalitesi olduğunda tercih edilir. Üç temel tip CCD mimarisi vardır [11]. Bunlar:

i. Interline transfer
ii. Full-Frame
iii. Frame transfer

<u>Interline transfer:</u>

CCD x, y matris düzeninde satır ve sütunlar içeren piksellerden veya görüntü elementlerinden oluşur. Her piksel bir fotodiyot ve bitişik bir şarj transfer bölgesinden meydana gelir. Bitişik şarj transfer bölgeleri birer sütün da yatay şarj transfer hatları biçimindedir. Fotodiyotlar ışık enerjisini (photon'ları algılayıp),

elektrik enerjisine çeviren elemanlardır. Oluşan elektrik sinyali ışık yoğunluğunun oranıdır. Genel olarak görüntülenen ortamdan gelen tüm ışık eşzamanlı olarak alınır ve bitişik sütünlardaki şarj transfer hücrelerine aktarılır.

Sonraki şarj mutlaka aktarılmış olmalıdır. Bunun için dikey şarj transfer hatları ayrılarak, sütundaki her bir pikselden elde edilen ve sinyal içeren bir satırlık veri transfer edilir. Böylece şarj paketleri seri olarak verilen satır için okunarak şarj-voltaj dönüşümü gerçekleştirilir.

Bu mimari, düşük gürültü seviyeli ve yüksek performanslı görüntüleme sağlar. Bu donanımların iyi tarafı, çok çabuk (mikro saniyeler veya daha küçük zaman dilimlerinde) kesme yapabilmeleridir. Bu durum, çok kısa zaman dilimlerinde oluşan iki olayın karşılaştırmalı ölçümünde onları çok güçlü algılayıcılar olarak ortaya çıkarır. Ayrıca CCD ler yazılımsal olarak da kontrol edilebilir. Olumsuz yönü ise ışık enerjisini düşürmekte ve potansiyel olarak negatif optik efektler göstermektedir.

Full Frame:

Full Frame CCD' de tüm CCD sensörleri ışığı algılama anı boyunca kullanıma hazırdırlar. Bu tip kameralar bir görüntünün oluşma anını, mekanik kesiciler veya strobe ışıkları ile kontrol eder. Kesicinin veya ışığın açık olması ile tüm pikseller şarj olur. Ancak mekanik kesiciler yüksek resim çerçeve oranı (20-30 fps) için uygun değillerdir.

Bu yapıda anlık görüntünün photon'larını algılamak için her bir pikselin tüm yüzeyi kullanılır. Ayrıca pikseller tipik olarak kare formundadır. Böylece piksel sensörlerinde dolayı gürültü olmaz.

Frame Transfer:

Frame Transfer CCD görüntüleyiciler, iki alana ayrılmış paralel, kayıt ortamına sahiptir. Üstteki resim piksellerinin oluşturduğu dizidir. Diğeri ise depolama

dizisidir. Büyüklüğü piksel dizisi ile tamamen aynıdır ve toplanan şarjı geçici olarak depolamak için saydam olmayan bir örtü ile kaplanmıştır. Görüntüden gelen ışıkla oluşan şarj olan piksel elementleri sinyali anında depolama hücresine iletirler ve bir sonraki resim için hazır halde olurlar. Bir frame transfer CCD görüntüleme elemanı herhangi bir kesiciye ihtiyaç duymaksızın yüksek görüntüleme oranlarında kullanılabilir.

CCD kameralarda iki temel tarama tekniği kullanılır. İlki aralıklı tarama (interlaced scanning) olarak bilinir. Bu teknik televizyon yayınlarında düşük görüntü çerçeve oranında daha iyi bir sonuç almak için kullanılır. Resim çerçevesinin üstünden itibaren ilk anda tüm tek satırları (1, 3, 5,...,2n-1 burada n toplam satır sayısıdır) okunur. Sonra anda yine baştan başlanarak çift satırlar (0, 2, 4,....,2n-2) okunur. Görüntü sistemlerinde interlaced taramada hareketli nesneler görüntülenirken bitişik satırlar farklı zamanlarda elde edildiği için nesne tek ve çift satırlarda farklı noktalarda gözükmekte bu da sorunlar oluşturmaktadır.

Diğer tarama tekniği ise aşamalı tarama (progresive scanning) olarak bilinir. Görüntüleme sistemlerinde aşamalı tarama yapan kameralar daha popülerdir. Bu tipte satırlar kendi doğal sıralarına göre (0, 1, 2,.....) yukarıdan aşağıya toplanırlar.

2.4 Basit Bir Görüntü Oluşturma Modeli

Sayısal görüntüler f(x,y) şeklinde iki boyutlu fonksiyonlar şeklinde ifade edilebilir. Uzaysal koordinatlardaki değeri veya genliğinin fiziksel anlamı, görüntü kaynağı tarafından belirlenen *f* pozitif sayısal bir niceliktir. Bu uygulamada değerleri gri ölçekte uzanan tek renkli görüntüler kullanılmıştır. Bir görüntü fiziksel işlemler sonucu oluşturulduğu zaman, değerleri fiziksel kaynaktan (örneğin, elektromanyetik dalgalar) yayılan enerji ile orantılıdır. Sonuç olarak, *f(x,y)* sıfır olmamalı ve sonlu olmalıdır;

$$0 < f(x,y) < \infty \qquad (2.6)$$

Bu fonksiyon *f(x,y)* iki bileşen tarafından tanımlanabilir:

1) görüntülenen manzara üzerine düşen kaynak aydınlatma miktarı ve
2) manzaradaki nesnelerden yansıyan aydınlanma miktarı.

bunlara *aydınlanma* ve *yansıma* bileşenleri denir ve sırasıyla *ay(x,y)* ve *yan(x,y)* olarak gösterilir. Bu iki fonksiyon birleşerek *f(x,y)* fonksiyonunu oluşturur [18]:

$$f(x,y) = ay(x,y)\, yan(x,y) \tag{2.7}$$

Burada aydınlatma ve yansıma bileşenleri Eş. 2.8 ve Eş. 2.9 dadır.

$$0 < ay(x,y) < \infty \tag{2.8}$$

$$0 < yan(x,y) < 1 \tag{2.9}$$

Eşitlik (2.9) yansımanın 0 (toplam emme) ve 1 (toplam yansıma) arasında sınırlı olduğunu göstermektedir. *ay(x,y)* aydınlatma kaynağı tarafından ve *yan(x,y)* görüntülenen nesnelerin karakteristiklerinden tespit edilir. Herhangi bir koordinattaki (x_0,y_0) tek renkli görüntü ışık şiddetine, görüntünün o noktadaki gri seviyesi denir. Burada *l* Eş.2.10 ile elde edilir.

$$l = f(x_0,y_0) \tag{2.10}$$

$$L\min \le l \le L\max \tag{2.11}$$

l'nin Eş 2.11 ile verilen aralıkta olduğu açıktır. Teoride, L_{min} için tek gereklilik pozitif olmasıdır ve L_{max} için ise sonlu olmasıdır. Pratikte, $L_{min}=ay_{min}\, yan_{min}$ ve $L_{max}=ay_{max}\, yan_{max}$ dir.

[L_{min}, L_{max}] aralığı gri ölçek olarak bilinir. Yaygın uygulama bu aralığı sayısal olarak [0, L-1] aralığına kaydırmaktır, gri ölçekte l = 0 siyah olarak ve l = L-1 beyaz olarak düşünülür. Tüm ara değerler siyahtan beyaza değişen grinin tonlarıdır. Şekil 2.12'de 8 bit gri düzey görülmektedir. Burada L_{min}=0, L_{max}=255 dir

.



Şekil 2.12. 256 düzey gri renk

2.5 Görüntü Örnekleme (Sampling) ve Nicelikleştirme (Quantization)

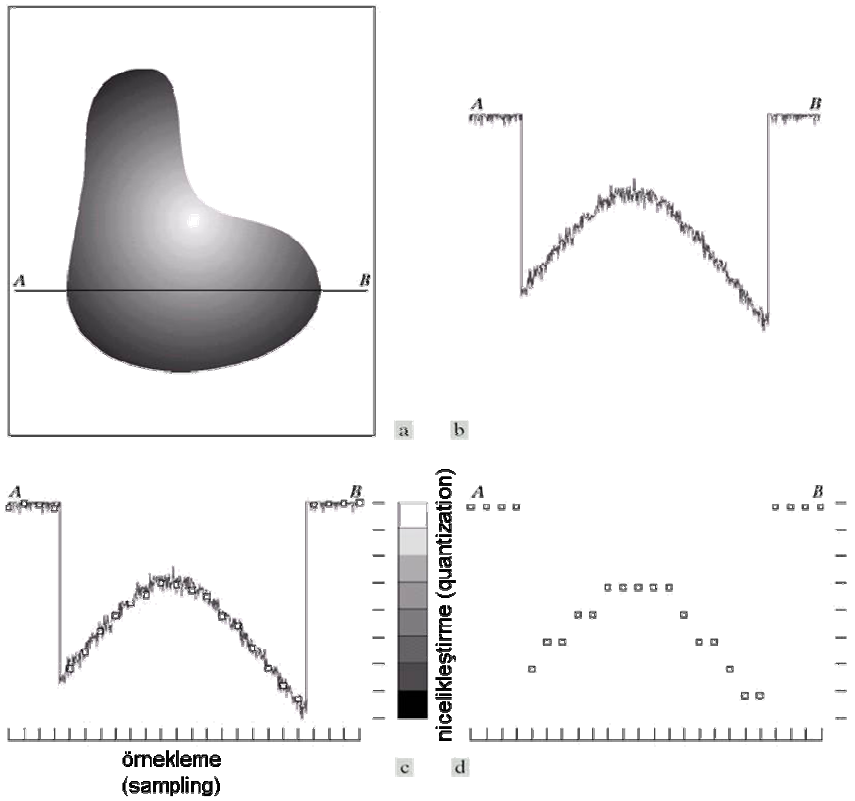
Örneklemede temel amaç algılanan veriden sayısal görüntü oluşturmaktır. Çoğu algılayıcının çıkışı, genliği ve uzaysal davranışı algılanan fiziksel olgu ile ilişkili olan, bir sürekli elektrik sinyali şeklidir. Sayısal bir görüntü oluşturmak için, algılanan sürekli verinin sayısal biçime dönüştürülmesi gerekir. Sayısallaştırma işlemi iki işlemi içermektedir: örnekleme ve nicelikleştirme.

2.5.1 Örnekleme ve nicelikleştirmede temel kavramlar

Örnekleme ve nicelikleştirmedeki temel fikir Şekil 2.13'de gösterilmiştir [18]. Şekil 2.13(a) sayısal biçime dönüştürmek istenilen sürekli görüntüyü *f(x,y)* göstermektedir. Bir görüntüyü, sayısal görüntüye dönüştürmek için, görüntü fonksiyonun hem koordinatlarda hem de genlikte örneklenmesi gerekir. Koordinatların sayısallaştırılmasına *örnekleme,* genliğin sayısallaştırılmasına da *nicelikleştirme* denir.

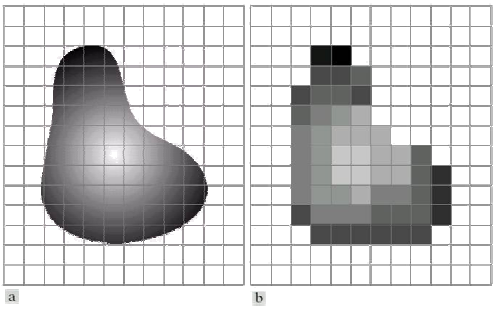
Şekil 2.13(a)'daki AB doğrusu parçası boyunca görüntünün genlik (gri seviye) değerine ait video sinyali, Şekil 2.13(b)'de gösterildiği gibi tek boyutlu sürekli bir fonksiyondur. Rastgele değişiklikler görüntü gürültüsüne bağlıdır. Bu fonksiyonu örneklemek için, Şekil 2.13(c)'deki gibi AB doğrusu boyunca eş aralıklı örnekler

alınır. Her bir örneğin yeri, şeklin alt kısmında dikey işaret ile verilmiştir. Örnekler fonksiyonun üzerine yerleştirilmiş küçük beyaz kareler olarak gösterilmiştir. Bu ayrık yerler kümesi örneklenmiş fonksiyonu vermektedir. Sayısal bir fonksiyon oluşturmak için, aynı zamanda gri-seviye değerleri de sayısal niceliklere dönüştürülmelidir (nicelikleştirme). Şekil 2.13(c)'de görüldüğü gibi, gri-seviye ölçeği siyahtan beyaza değişen sekiz ayrık seviyeye ile gösterilmiştir. Yatay işaretler her bir sekiz gri seviyeye atanan özel değeri belirtmektedir. Sürekli gri seviyeler, herbir örneğin sekiz ayrık gri seviyeden bir tanesine atanması ile nicelikleştirilir. Atama, örneğin dikey işarete olan dikey yakınlığına bağlı olarak yapılır. Örnekleme ve nicelikleştirme sonucunda elde edilen sayısal örnekler Şekil 2.13(d)'de gösterilmiştir. Görüntünün üstünden başlayarak ve bu prosedürü çizgi çizgi gerçekleştirerek iki boyutlu sayısal görüntü oluşturulur. Pratikte örnekleme yöntemi, görüntü oluşturmak için kullanılan algılayıcı ayarlaması ile tespit edilir.



Şekil 2.13. Örnekleme ve sayısallaştırma

Görüntü elde etmek için bir algılayıcı dizi (CCD) kullanıldığı zaman, dizideki algılayıcıların sayısı her iki yöndeki örnekleme sınırlarını saptar. Algılayıcı çıkışlarının nicelikleştirilmesi daha önceki gibidir. Şekil 2.14 bu kavramı göstermektedir. Şekil 2.14(a) dizi algılayıcının düzlemi üzerine yansıtılmış nesneye ait görüntüyü göstermektedir. Şekil 2.14(b) örnekleme ve nicelikleştirmeden sonraki görüntüyü göstermektedir [18].



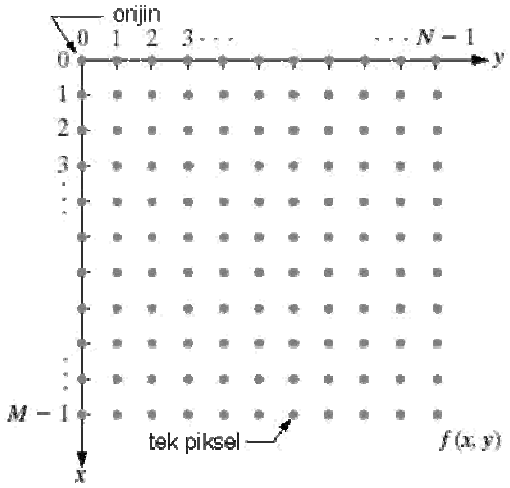
Şekil 2.14. Nesnenin resim düzlemi üzerindeki izdüşümü ve sayısallaştırılmış hali

Sayısal görüntünün kalitesi, örneklemenin sayısı ve nicelikleştirme için kullanılan ayrık gri seviyelerin sayısı ile belirlenir.

2.6 Sayısal Görüntünün İfade Edilmesi

Örnekleme işlemi sonucu sayısal görüntünün çözünürlüğü belirlenmiş olur (örneğin:1024x768, 720x576, 640x480), nicelikleştirme sonucu ise resimdeki renk derinliği belirlenmiş olur (renk derinliği; 8 bit gri seviye için 256, 16 bit gri seviye için 65536). Ayrıca örnekleme ve nicelikleştirmenin sonucunda gerçek sayılardan oluşan bir matris elde edilir. Bir sayısal $f(x,y)$ görüntüsünün M satır ve N sütunlu sayısal görüntü oluşturacak şekilde örneklendiğinde kordinatların değerleri (x, y), ayrık nicelikler halindedir. Notasyon açıklığı ve uygunluğu için, bu ayrık koordinatlar tamsayı değerlerle ifade edilir. Böylece, merkezdeki koordinatların değerleri $(x,y) = (0,0)$'dır. Görüntünün sonraki ilk satırı boyuncaki koordinat değerleri $(x,y) = (0,1)$ olarak gösterilir. (0,1) notasyonunun ilk satır boyuncaki ikinci örneği belirtmek için kullanıldığını akılda tutmak önemlidir. Bu, görüntü

örneklendiğinde, fiziksel koordinatların gerçek değerleri olduğu anlamına gelmemektedir. Şekil 2.15'de bu uygulama boyunca kullanılan koordinat tanımını göstermektedir [19].



Şekil 2.15. Sayısal resim koordinat yapısı

Bu notasyon, tüm *MxN* boyutlarındaki sayısal görüntüyü aşağıdaki matris formunda yazmayı sağlamaktadır.

$$f(x,y)=\begin{bmatrix} f(0,0) & f(0,1) & . & . & f(0,N-1) \\ f(1,0) & f(1,1) & . & . & f(1,N-1) \\ . & . & & & . \\ . & . & & & . \\ f(M-1,0) & f(M-1,1) & . & . & f(M-1,N-1) \end{bmatrix} \tag{2.12}$$

Eşitliğin sağ tarafı sayısal görüntünün tanımıdır. Matris dizisindeki her eleman görüntü elemanı, resim elemanı, piksel ve pel olarak adlandırılır. Görüntü ve piksel terimleri, bu tezin kalan bölümleri boyunca sayısal görüntüyü ve elemanlarını belirtmek için kullanılmıştır.

Bazı durumlarda, sayısal görüntüyü ve elemanlarını belirtmek için daha geleneksel matris notasyonu kullanmak avantajlı olabilir.

$$A = \begin{bmatrix} a_{0,0} & a_{0,1} & . & . & a_{0,N-1} \\ a_{1,0} & a_{1,1} & . & . & a_{1,N-1} \\ . & . & & & . \\ . & . & & & . \\ a_{M-1,0} & a_{M-1,1} & . & . & a_{M-1,N-1} \end{bmatrix} \tag{2.13}$$

Burada, $a_{ij} = f(x = i, y = j) = f(i,j)$, bundan dolayı *f(x, y)* ve *A(i, j)* özdeş matrislerdir. Örnekleme ve nicelikleştirmeyi daha formel matematiksel terimlerle ifade etmek sayısal görüntüyü daha anlaşılır kılar. Z ve R sırasıyla tamsayılar kümesini ve gerçek sayılar kümesini ifade etmektedir. Örnekleme işlemi x-y düzlemini gridler içinde kısımlara ayırarak görülebilir, burada her bir grid merkezinin koordinatları kartezyen çarpım Z^2'nin elemanı olan (z_i, z_j) düzenli eleman çiftlerinin kümesidir ve z_i ve z_j Z kümesinin (tamsayılar kümesi) elemanlarıdır. Bu nedenle, eğer (x, y) Z^2 kümesinin tamsayı elemanları ise *f(x,y)* sayısal görüntüdür ve *f*, bir gri seviye değerini gösteren ve gerçek sayılar kümesinin elemanı olan R'yi, her bir farklı (x,y) koordinat çiftine atayan bir fonksiyondur. Bu fonksiyonel atama açıkça daha önce tanımlanan nicelikleştirme işlemidir. Eğer gri seviyeler aynı zamanda tamsayı ise, Z, R' de yer alır ve sonra sayısal görüntü koordinatları ve genlik değerleri tamsayı olan 2-boyutlu fonksiyon haline gelir.

Bu sayısallaştırma işlemi *M,N* değerleri hakkında ve her bir piksel için izin verilmiş ayrık gri seviyelerin sayısı *L* için kararlar gerektirir. *M* ve *N* için pozitif tamsayı olmaları gerektiği dışında herhangi bir gereklilik yoktur. Örnekleme donanım faktörlerine bağlı olarak, gri seviyelerin sayısı tipik olarak 2'nin tamsayı kuvvetidir.

$$L=2^k \tag{2.14}$$

Ayrık seviyeler eş uzaklıkta yerleşmiştir ve [0, *L*-1] aralığında tamsayıdırlar. Bazen gri skalada uzanan değerler aralığına görüntünün dinamik aralığı denir. Fazla sayıdaki pikseller bu özelliği ortaya koyduğunda, görüntü yüksek kontrasta sahip olacaktır. Tersine düşük dinamik aralığına sahip görüntüler mat, yıkanmış gri bakışa sahip olma eğilimindedirler.

Sayısallaştırılmış görüntüyü depolamak için gerekli bit sayısı, b,

$$B_s = M \, x \, \mathrm{N} \, x \, k \tag{2.15}$$

şeklindedir. $M = N$ olduğunda yukarıdaki denklem 2.15 den denklem 2.16 elde edilir.

$$B_s = N^2 k \tag{2.16}$$

Bir görüntünün 2^k gri seviyesi olabileceği zaman, görüntüyü k-bit görüntü olarak belirtmek genel bir yaklaşımdır. Örneğin, 256 muhtemel gri-seviye değerleri olan bir görüntüye 8-bit görüntü denir ve k=2 dir. 1024x1024 veya daha yüksek boyuttaki 8-bit görüntülerin depolama için kullanılan miktar 8388608 bit olup bu 1048576 Byte'a eşittir.

2.7 Pikseller Arasındaki Temel İlişkiler

2.7.1 Bir pikselin komşuları

(x,y) koordinatlarındaki bir pikselin aşağıda belirtilen dört yatay ve dikey komşuları vardır; *(x+1,y), (x-1,y), (x,y+1), (x,y-1),* p'nin 4-komşusu olarak bilinen bu pikseller kümesi $N_4(p)$ olarak gösterilir. Her bir piksel (x,y)'den 1 birim uzaklıktadır. Eğer (x,y) görüntünün sınırında ise p'nin bazı komşuları sayısal görüntünün dışında kalır. p'nin dört diyagonal komşusunun koordinatları, *(x+1,y+1), (x+1,y-1), (x-1,y+1), (x-1,y-1)* şeklindedir ve $N_D(p)$ olarak gösterilir. Bu noktalara, 4-komşu noktaları ile birlikte p'nin 8-komşuları denir ve $N_8(p)$ olarak gösterilir. Eğer (x,y) görüntünün sınırında ise, $N_D(p)$ ve $N_8(p)$'nin bazı noktaları sayısal görüntünün dışında kalır.

2.7.2 Yakınlık, bağlanabilirlik, bölgeler ve sınırlar

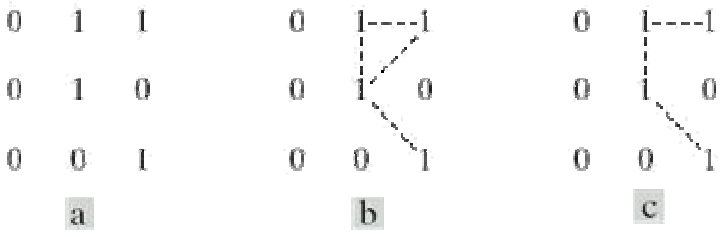
Pikseller arasındaki bağlanabilirlik, bölge ve sınırlar gibi birçok sayısal görüntü kavramlarının tanımlarını basitleştiren temel ifadelerdir. İki pikselin bağlanabilirliğinin saptanması için, komşu olduklarının veya gri seviyelerinin

benzerliğin belirlenmiş kriterini sağladıklarının (gri seviyelerinin eşit olduğunun) tespit edilmesi gerekmektedir. Örneğin, ikili görüntüde, iki piksel 4' lü komşu olabilir, fakat sadece aynı değere sahiplerse bağlanmış oldukları söylenebilir.

V, bitişikliği ifade etmek için kullanılan gri-seviye değerleri kümesi olsun. İkili Seviye (binary) bir görüntüde, eğer piksellerin bitişikliği 1 değeri ile gösteriyorsak, V={1}'dir. Gri-skala görüntüsünde de fikir aynıdır, fakat V daha fazla eleman içerir. Örneğin, 0-255 gri-seviyesi değerleri aralığındaki piksellerin bitişikliğinde, V kümesi 256 değerin herhangi bir alt kümesi olabilir. Üç tip yakınlık düşünülebilir[18]:

a) 4-yakınlık: eğer q, $N_4(p)$ kümesinin içinde ise, V'nin değerlerinden oluşan iki p ve q pikselleri 4-yakınlıktır.
b) 8-yakınlık: eğer q, $N_8(p)$ kümesinin içinde ise ,V'nin değerlerinden oluşan iki p ve q pikselleri 8-yakınlıktır.
c) m-yakınlık (karışık yakınlık): Eğer q, $N_4(p)$ içinde ise veya q, $N_D(p)$ içinde ve $N_4(p) \cap N_4(q)$'nun değerleri V'den olan hiç pikseli yok ise V'nin değerlerinden oluşan iki p ve q pikselleri m-yakınlıktır.

Karışık yakınlık, 8-yakınlığın bir değişikliğidir. Karışık yakınlık, 8-yakınlık kullanıldığı zaman sıkça ortaya çıkan belirsizliği ortadan kaldırmak için çıkarılmıştır. Örneğin, V={1} için Şekil 2.16(a)'da gösterilen piksel yerleşimini ele alınırsa Şekil 2.16(b)'nin üst kısmındaki 3 piksel, kesikli çizgiler ile çeşitli (belirsiz) 8-yakınlığı göstermektedir.



Şekil 2.16. a)rastgele bir piksel yerleşimi, b)8-yakınlık, c)m-yakınlık

Bu belirsizlik Şekil 2.16(c)'de görüldüğü gibi m-yakınlık kullanılarak ortadan kalkmıştır. Eğer S_1 altkümesindeki bazı pikseller S_2 altkümesindeki bazı piksellere bitişik ise, iki görüntü altkümesi S_1 ve S_2 yakındır.

(x,y) koordinatlı p pikselinden, (s,t) koordinatlı q pikseline doğru olan bir (sayısal) yol (veya eğri), koordinatları (x_0,y_0), (x_1,y_1), ..., (x_n,y_n) şeklindeki farklı pikseller dizisidir. Burada $(x_0,y_0) = (x,y)$, $(x_n,y_n) = (s,t)$ ve $1 \leq i \leq n$ için (x_i,y_i) ve (x_{i-1},y_{i-1}) pikselleri yakındır. Bu durumda n yolun uzunluğudur. Eğer $(x_0,y_0) = (x_n,y_n)$ ise yol kapalı yoldur. 4-, 8-, veya m- yolları, belirlenmiş yakınlık tipine bağlı olarak tanımlanabilir. Örneğin, Şekil 2.16(b)'de gösterilen kuzeydoğu ve güneydoğu noktaları arasındaki yollar 8-yoldur ve Şekil 2.16(c)'deki yol ise m-yoldur. m-yolunda belirsizlik olmadığı görülmektedir.

3. GÖRÜNTÜ HAZIRLAMA VE GELİŞTİRME

3.1 Giriş

Görüntü hazırlama ve iyileştirmenin amacı uygulamanın yapısına, şekline göre orjinal resimden daha kullanışlı bir görüntü elde etmektir. Örneğin X-ışınıyla çekilmiş bir görüntü üzerinde iyileştirme uygulamak daha iyi sonuç verebilir fakat bir uzay aracının gönderdiği görüntüler üzerinde iyileştirme işlemleri pek gerekli olmayabilir. Basit bir ifadeyle, uygulamanın yapısı ve şekli önemlidir.

Görüntü iyileştirme ve geliştirmede ele alınacak yaklaşım iki ana kategoriye ayrılabilir. Bunlar ;

Resim Tanım Bölgesi (Spatial Domain) de iyileştirme ve geliştirme, yani direk olarak resim düzlemi üzerinde piksellerle işlem yapmayı, diğeride,

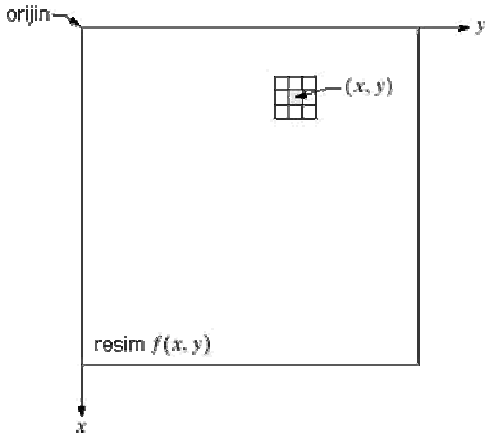
Frekans Tanım Bölgesi (Frequency Domain) de iyileştirme ve geliştirme, görüntünün Fourier dönüşümü yapılarak uygulanan iyileştirme tekniklerini içerir. Bu tezde ve hazırlanan uygulamada frekans tanım bölgesi kullanılmıştır. Yukarıda da belirtildiği gibi Spatial Domain, resim düzleminin kendisini ifade etmektedir. Hazırlanan yazılımda spatial domain kullanıldığı için direk olarak pikseller üzerinde işlemler gerçekleştirilmiştir. Dolayısıyla girdi ve çıktı elemanı piksellerdir.

Resim Tanım Bölgesinde yapılacak işlemler iki gruba ayrılabilir. Bunlar gri seviye dönüşümleri (intesity transformasyonu) ve filtreleme (spatial filtering) işlemleridir.

Görüntü Tanım Bölgesinde direk, resim pikselleri üzerinde yapılan işlemler, Eş.3.1 ile gösterilir [19].

$$g(x, y) = D[f(x, y)] \tag{3.1}$$

Burada *f(x,y)* üzerinde işlem yapılacak görüntü, *g(x,y)* işlenmiş görüntü, *D* ise giriş görüntüsündeki *(x,y)* noktasının özel piksel komşuluklarına göre tanımlanan işlem operatörüdür. Bir görüntü dosyasındaki *(x,y)* noktası Şekil 3.1'deki gibi ifade edilebilir.



Şekil 3.1. Görüntü içindeki bir (x,y) noktası ve 3x3 komşuluğu

D operatörü tüm *(x,y)* noktaları için uygulanır. Böylece *f(x,y)* görüntüsündeki her nokta için *g(x,y)* görüntüsü oluşturulur.

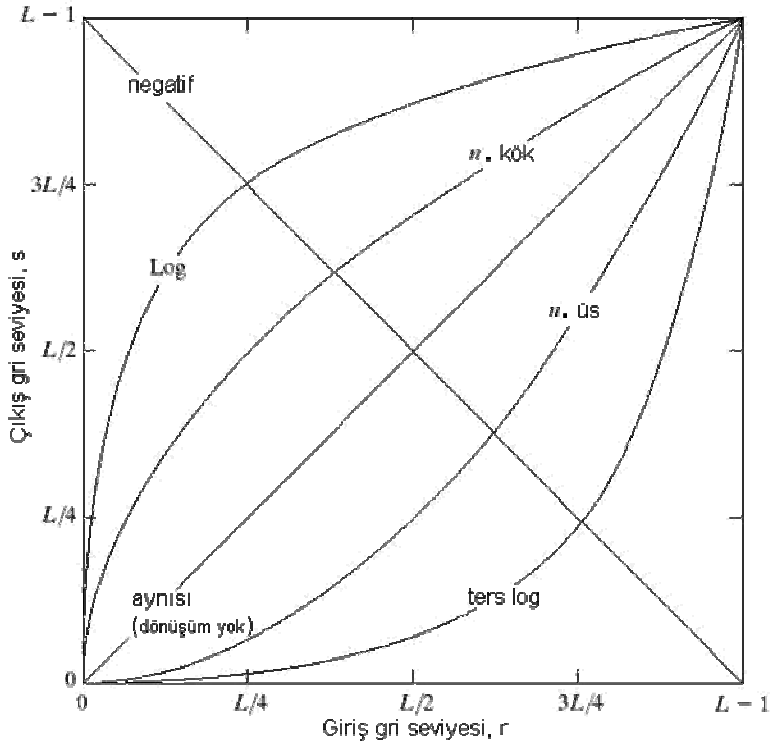
3.2 Gri Seviye Dönüşümü

Gri seviye dönüşümü en basit dönüşüm formudur. *g*' deki (x,y) noktasının değeri sadece *f*' deki (x,y)'nin ışık seviyesi şiddetine (intensity level) bağlıdır. Tekrenk (Monochrome) görüntülerde (ör: gri seviye görüntü) ışık şiddeti seviyesi, gri seviye ile aynı anlamda kullanılabilir. Renkli görüntüde ise intensity terimi ilgili renk bileşenlerini ifade eder. *f* ve *g* deki ilgili noktalara ait intensity dönüşümü Eş.3.2 dir.

$$s = D(r) \tag{3.2}$$

Burada *r* ve *s* sırasıyla *f* ve *g*' nin ilgili *(x,y)* noktaları için gri seviye değerleridir. 8 bit' lik renk seviyesinde *D*, 0ile 256 arasında değer alabilir. Bu da 256 farklı gri seviyesini ifade etmektedir.

Gri Seviye dönüşümünün üç temel kullanımı; lineer dönüşüm, logaritmik dönüşüm, üssel dönüşümle ilgili grafik Şekil 3.2' de görülmektedir [18].



Şekil 3.2. Gri seviye dönüşümleri, (negatif, üssel, logaritmik)

Kullanılan Görüntü İşleme Yazılımın da bu işlemi kolaylaştıran önceden tanımlı fonksiyonlar mevcuttur. Giriş ve çıkış görüntüsüne ait parametreler tanımlanarak görüntü üzerinde istenen düzeltme yapılabilir.

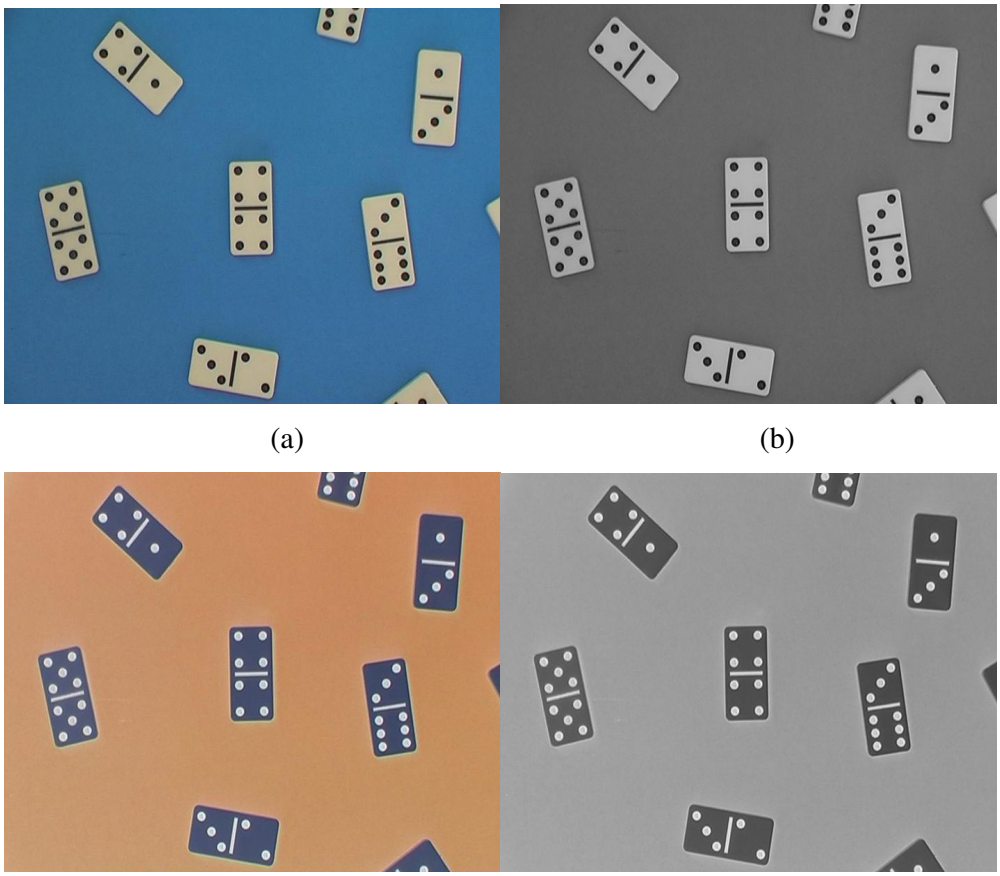
3.2.1 Negatif dönüşüm

Gri seviye bir görüntü için negatif dönüşüm aşağıdaki Eş.3.3 ile uygulanır.

$$s = L - 1 - r \tag{3.3}$$

Burada r giriş gri seviyesi, s çıkış gri seviyesi, L ise en yüksek gri seviyedir (8 bit'lik görüntü için 256, 16 bitlik görüntü için 65536). Bu işlem sonrası herhangi bir

görüntünün fotografik negatifi elde edilir. Resim 3.1 (a) ve (b) sırasıyla domino taşlarının RGB ve Gri seviye görüntüleri, (c) ve (d)' de ise taşların negatif dönüşümleri görülmektedir.



(a) (b)

(c) (d)

Resim 3.1. RBG ve gri görüntünün negatif dönüşümleri

3.2.2 Logaritmik ve kontrast germe dönüşümü

Logaritmik dönüşüm aşağıdaki Eş.3.4 ile elde edilir [19]. Bu dönüşümle girişte küçük bir aralıkta değişen gri seviye değeri çıkışta büyük aralığa dönüştürülür veya tersi yapılır.

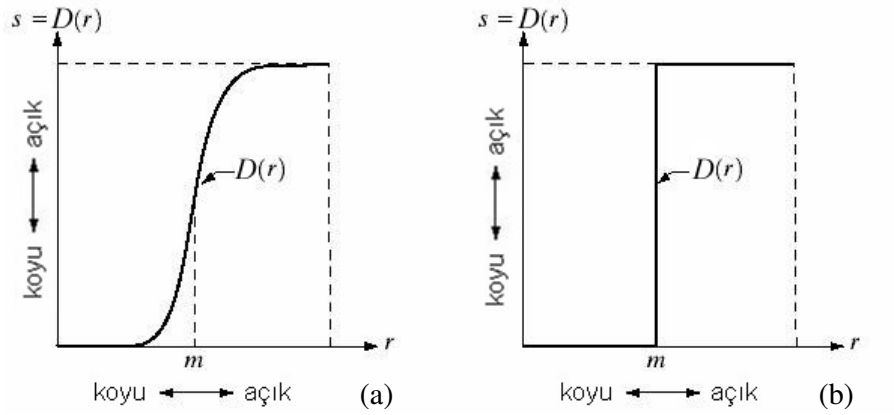
$$s = c_1 \, Log(1 + r) \tag{3.4}$$

Burada c_1' herhangi bir sabittir ve $r \geq 0$ dır. Üssel dönüşüme benzemektedir fakat gama değeri gibi değişken değil sabittir. Logaritmik dönüşümün ana kullanımlarından biri gri seviye değerlerini istenen şekilde bastırmaktır. Örneğin 8 bit bir gri görüntüde yüksek değerler (beyaz seviye) baskınsa bu durumda düşük değerli bölgeler belirsizleşebilir. Bu durumda log dönüşümü yapılarak çıkış görüntüsü daha kullanışlı hale getirilebilir.

Kontrast germede ise giriş görüntüsündeki belli bir değerin altındaki değerler çıkışta daha dar bir aralıkta gösterilir. Aynı şekilde bu değerin üstündeki değerlerde bir aralığa sıkıştırılmış olur. Eş.3.5'de kontrast germe fonksiyonu görülmektedir.

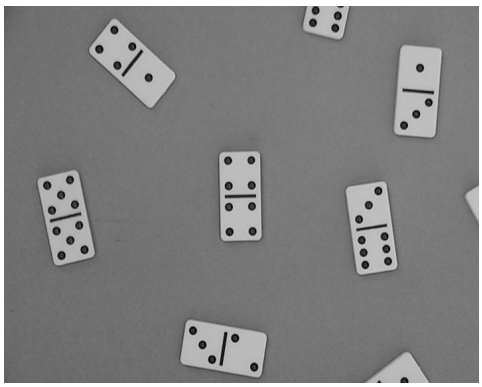
$$s = D(r) = \frac{1}{1 + (m/r)^E} \tag{3.5}$$

Burada m kontrast germesinin uygulanacağı noktayı belirler. r ve s sırasıyla giriş ve çıkış görüntüleridir. E ise fonksiyonun eğimini kontrol eder.

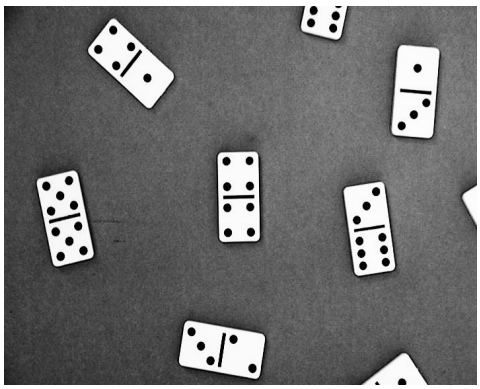


Şekil 3.3. Logaritmik ve kontrast germe dönüşümü

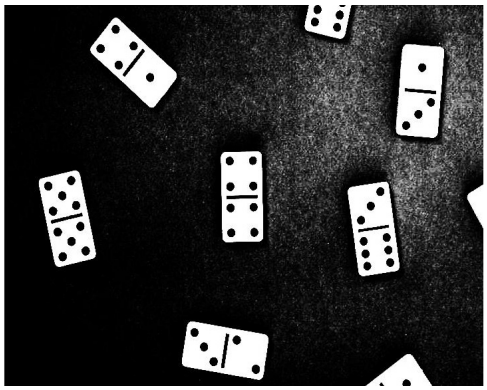
Şekil 3.3(a)'da kontrast germe fonksiyonunun grafiği görülmektedir ve burada E=20dir. Şekil 3.3.(b)'de m noktasında eğimin sonsuz olduğu görülmektedir [19]. Bu durumda Eş.3.5, "eşikleme" fonksiyonuna dönüşmüş olur. Resim 3.2'de domino taşlarına ait logaritmik dönüşümler görülmektedir.



(a)



(b)



(c)

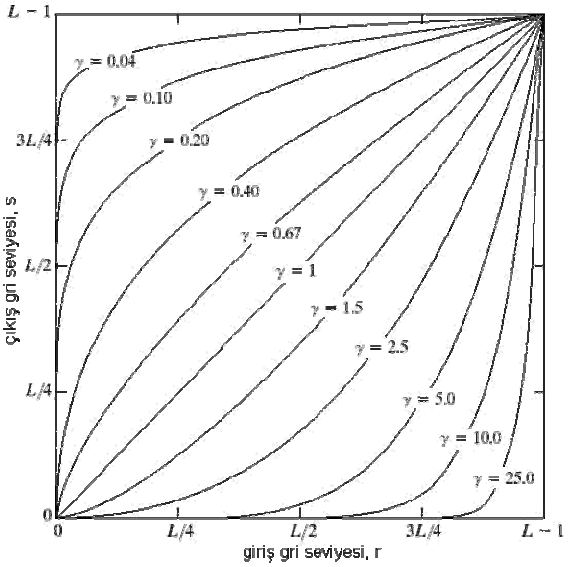
Resim 3.2. Sırasıyla E=2,8 ve 50 için log dönüşümü.(m için resim ortalama gri değeri alınmıştır.)

3.2.3 Üssel dönüşüm

Üssel dönüşüm "gamma correction" olarak da adlandırılır ve Eş.3.6 ile elde edilir.

$$s = c_2 r^{\gamma} \tag{3.6}$$

Burada c_2 ve γ pozitif sabitlerdir. γ'nın değişimine göre $r - s$ grafiği Şekil 3.4'deki gibidir. Gamma değeri çıkış görüntüsünün, gri seviyesi değerlerinin dönüşüm eğriliğini belirtir. Gama değeri 1 ise girişle aynı çıkış görüntüsü elde edilir. Eğer gama<1 ise çıkış görüntüsü daha beyazımsı ve parlak, gamma>1 ise koyu siyahımsı bir görüntü elde edilir. Üssel dönüşümde farklı γ değerleri için muhtemel dönüşüm eğrileri Şekil 3.4 r-s grafiğinde görülmektedir [18].



Şekil 3.4. farklı γ değerleri için muhtemel dönüşüm eğrileri

3.3 Histogram İşleme

Histogram [0, L-1] arasındaki renk derinliğine sahip dijital bir görüntünün her bir renkteki piksel sayısının belirtilmesidir.

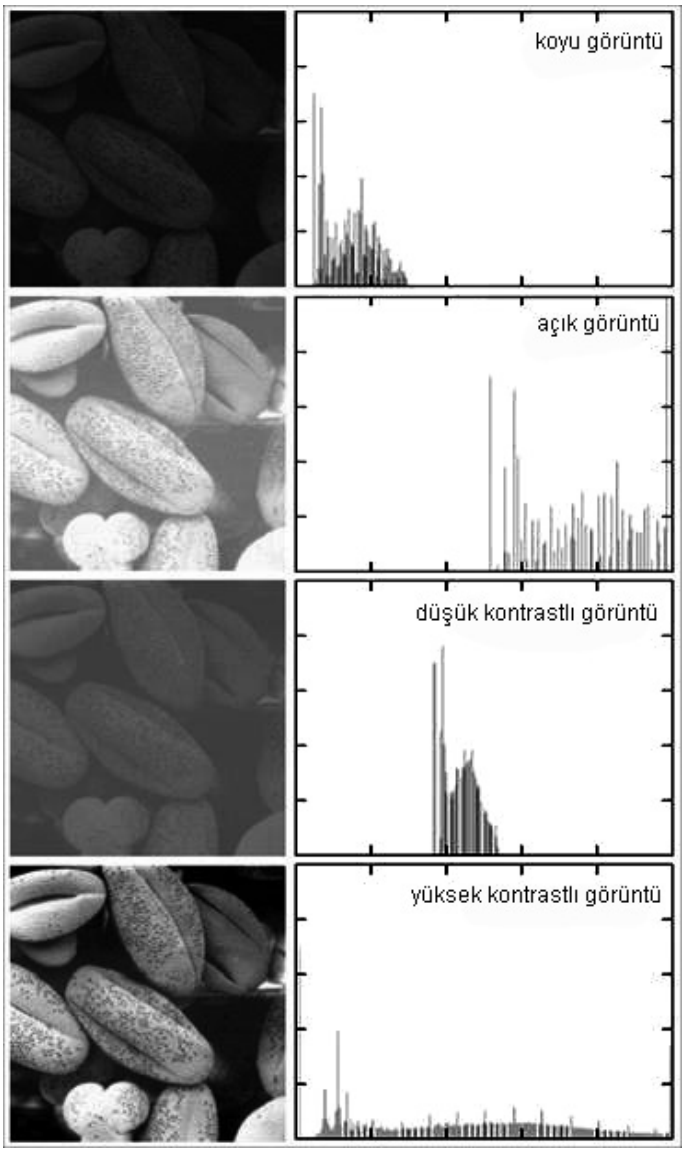
$$h(r_k) = n_k \tag{3.7}$$

Histogram fonksiyonu Eş.3.7 ile ifade edilir. Burada r_k; [0, L-1] arasında değişen k'ıncı gri seviye, n_k ; r_k' daki piksel sayısıdır. Görüntü içindeki toplam piksel sayısı "n" ise normalize edilmiş Histogram $p(r_k)$, Eş.3.8 ile bulunur.

$$p(r_k) = n_k/n \tag{3.8}$$

Histogram, görüntü tanım bölgesindeki pek çok işlem (görüntü geliştirme, görüntü düzeltme, görüntü sıkıştırma, bölümlendirme) için temel teşkil eder. Resim 3.3'de değişik görüntülerin histogram analizleri görülmektedir. Koyu görüntünün histogramı, sol tarafta, siyah bölgesine yakın şekilde dağılım göstermiş, benzer şekilde parlak görüntüde ise histogram beyaz bölgesinde toplanmıştır.

Düşük kontrasta sahip görüntüde histogram dar bir aralıktayken, yüksek kontrasta sahip görüntünün histogramı tüm gri seviye bandını kaplayacak şekilde dağılım göstermiştir. Bir görüntüye ait histogram, önceden tanımlanmış fonksiyon kullanılarak elde edilmiştir. Histogram 8 bit gri seviye için yani 256 renk için elde edilmiştir. Normalize edilmiş görüntüyü elde etmek için h toplam piksel sayısına bölünmelidir. Toplam piksel sayısı giriş görüntüsünün satır ve sütün değerleri çarpılarak elde edilir.



Resim 3.3. Bazı görüntülere ait histogram grafikleri [18]

3.3.1 Histogram eşitleme

Gri seviye değerlerinin [0, 1] normalize edilmiş aralığında ve sürekli miktarlarda olduğunu kabul edilsin. $p_r(r)$ ve $p_s(s)$ sırasıyla giriş ve çıkış görüntülerinin muhtemel yoğunluk fonksiyon (Probability Density Function - PDF) larını ifade etsin. Çıkış gri seviye değerlerini elde etmek için, giriş gri seviyelerine Eş.3.9'daki dönüşümü uygulanır.

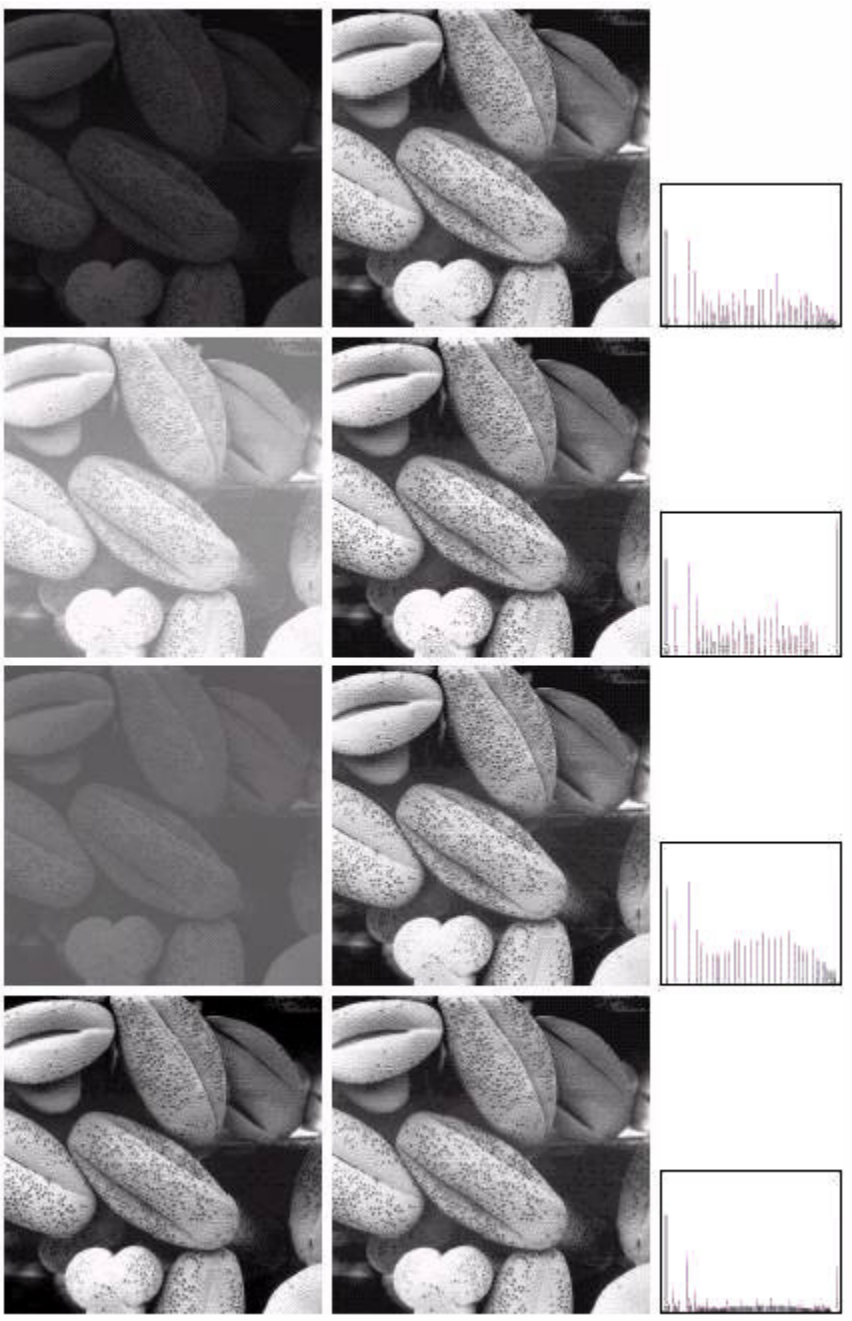
$$s = D(r) = \int_0^r p_r(\vartheta)\, d\vartheta \tag{3.9}$$

Burada ϑ sadece integral değişkenidir. Bu dönüşüm sonucu gri seviye değerleri giriş gri seviye değerleriyle tamamıyla benzer olan ve tüm [0, 1] aralığını kaplayacak şekilde dağılım gösteren bir görüntü elde edilir. Bu işlemin net sonucu yüksek kontrast seviyesine sahip bir görüntü elde edilmesidir. Yapı itibariyle bu dönüşüm aslında sadece Düzenli Dağılım Fonksiyonu (Cumulative Distribution Function – CDF) dur.

Histogram da olduğu gibi, ayrık miktarlarla işlem yapıldığında, bu teknik *Histogram Eşitleme* (Histogram Equalization) olarak adlandırılır. $p_r(r_j)$, j=1,2,.....,L histogramla ilişkili gri seviye değerlerini göstersin. Normalize edilmiş histogram, görüntü içindeki herbir griseviye değerinin oluşma olasılığını tahmin etmek için kullanılabilir. Bu durumda ayrık miktarlar için Eş.3.9'dan Eş.3.10 elde edilir [19]. Burada s_k k=1,2,.....,L için giriş görüntüsündeki ilgili r_k gri seviyesi için elde edilen çıkış görüntüsündeki gri seviye değeridir.

$$\begin{aligned} s = D(r) &= \int_0^r p_r(\vartheta)\, d\vartheta \\ &= \sum_{j=1}^{k} p_r(r_j) \\ &= \sum_{j=1}^{k} \frac{n_j}{n} \end{aligned} \tag{3.10}$$

Resim 3.4'de Resim 3.3'deki görüntürlerin Histogram eşitleme işlemi uygulanmış halleri görülmektedir. Resim 3.4'in ilk sütunu orjinal görüntüyü, ikinci sütunu histogram eşitleme uygulanmış görüntüyü, üçüncü sütun ise işlenmiş görüntülerin histogramını ifade etmektedir.



Resim 3.4. Histogram eşitleme yapılmış görüntüler [18]

Resim 3.4'de de bariz bir şekilde görülmektedir ki koyu, açık ve düşük kontrastlı görüntüde histogram eşitleme işlemi görüntüyü iyileştirmişken, yüksek kontrasta sahip görüntüde fazla bir değişim olmamıştır. Bunun nedeni, yüksek kontrastlı görüntünün histogramının zaten tüm gri seviye bandında dağılmış olmasıdır.

3.4 Görüntü Tanım Bölgesinde Filtreleme

Görüntü filtrelemenin amacı görüntüdeki gürültü ve bozuklukları bastırmaktır. Bu gürültü ve bozukluklar sensörlerden veya iletim ortamlarından kaynaklanmaktadır. Bazı ortak gürültü tipleri; salt & pepper (tuz ve biber), impulse gürültü, gaussian gürültü olarak sayılabilir. Tuz biber gürültüsü, görüntüde rastgele siyah-beyaz noktacıklar şeklinde görünür. Anlık değerli (impulse) gürültü sadece beyazlanma, renk açılması şeklinde görünür. Gaussian gürültü ise gri seviye görüntüde normal dağılım gösterir. Görüntüdeki bu gürültüleri silmek için lineer ve non-lineer filtreler kullanılır.

3.4.1 Filtrelemenin temeli

Bazı komşuluk operatörleri komşu piksellerin değerleriyle işlem yapar. Görüntü ile işleme giren altgörüntü , işlem gören pikselin komşularıyla aynı boyuta sahiptir. Görüntü ile işleme giren bu altbölüme ; filtre, maske, çekirdek, şablon, pencere gibi adlar verilir. Bu adlardan en yaygını filtre ve maske ifadeleridir. Filtre görüntüsündeki değerler aslında piksel değerlerinden çok, bir katsayı olarak görülmelidir.

Filtreleme mekanizmasının temelinde komşuluk ilişkileri yatmaktadır. Bu işlem basamakları;

i. Merkez noktayı (x, y) tanımlamak
ii. Merkez noktasının etrafındaki komşularla işlem uygulamak (8' li komşuluk, 4'lü komşuluk)
iii. İşlem sonucunu o nokta için almak

iv. İşlemi resimdeki tüm pikseller için uygulamak

şeklinde sıralanabilir. Görüntü tanım bölgesinde Filtreleme mekanizması aşağıdaki Şekil 3.5' de gösterilmiştir [19]. Görüldüğü gibi Filtreleme işlemi filtre maskesinin görüntü içinde noktadan noktaya taşınmasıyla uygulanır. Herbir (x, y) noktasında Filtreleme işleminin sonucu maske ve görüntünün aynı piksel hücrelerinin çarpılıp, hepsinin toplanmasıyla elde edilir. 3 X 3' lük bir maske için ;

$$R = w(-1,-1)\, f(x-1,\, y-1) + w(-1,\, 0)\, f(x-1,\, y) + \ldots\ldots\ldots\ldots + w(0,\, 0)\, f(x,\, y) + \ldots\ldots\ldots\ldots + w(1,\, 0)\, f(x+1,\, y) + w(1,\, 1)\, f(x+1,\, y+1)$$

dir. Burada R, $(x,\, y)$ noktasının filtreleme işlemi sonrası değerini belirtir. Filtrelemede maske seçilirken önemli bir nokta, maskenin boyutları $m \times n$ mutlaka tek olmalıdır. En küçük anlamlı maske boyutu 3 X 3 dür.

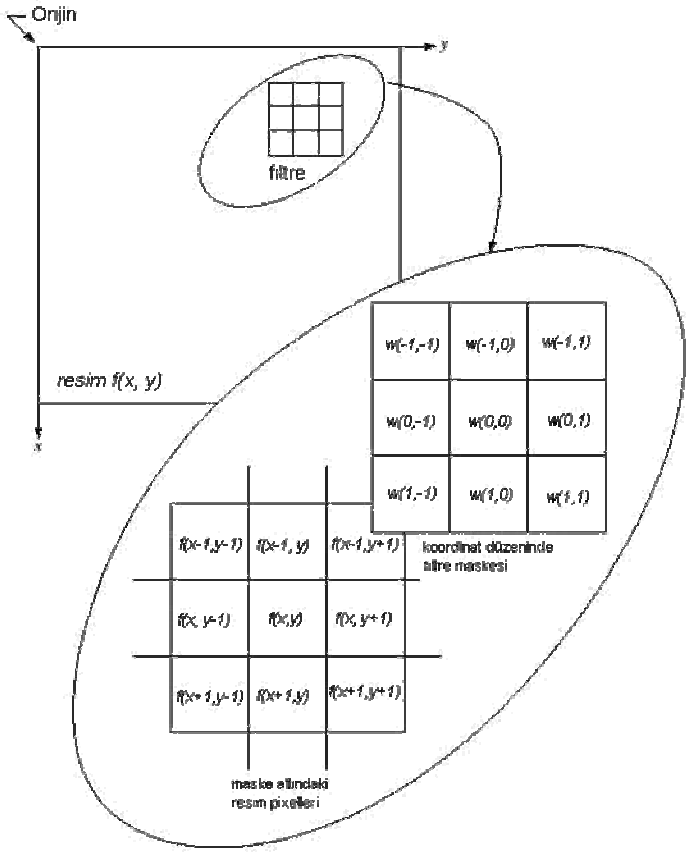
$M \times N$ boyutlarında bir f görüntüsüne $m \times n$ boyutlarında bir maske ile lineer filtreleme işlemi Eş. 3.11 ile uygulanır [18].

$$g(x,\, y) = \sum_{i=-a}^{a} \sum_{j=-b}^{b} w(i,\, j)\, f(x+i,\, y+j) \tag{3.11}$$

Burada a=(m-1)/2 , b=(n-1)/2 dir. Filtrelemede mekanizmadan ziyade önemli olan (x, y) noktasında uygulanan m x n maskenin sonuç değeridir. R için Eş.3.11 şu şekilde yazılabilir;

$$\begin{aligned} R &= w_1 z_1 + w_2 z_2 + \ldots\ldots\ldots + w_{mn} z_{mn} \\ &= \sum_{i=1}^{mn} w_i z_i \end{aligned} \tag{3.12}$$

Şekil 3.6'da 3X3 lük bir filtre maskesi görülmektedir. Bu maske için Eş 3.12 uygulanırsa R değeri için Eş 3.13 elde edilir.



Şekil 3.5. 3X3 lük bir maske ile, maskenin resim üzerinde uygulanma yapısı

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Şekil 3.6. 3X3 lük bir filtre maskesinin gösterimi

$$R = w_1 z_1 + w_2 z_2 + ...+ w_9 z_9 \qquad (3.13)$$

$$= \sum_{i=1}^{9} w_i z_i$$

Fitrelemede bir diğer önemli nokta, filtre maskesinin merkezi, görüntünün sınırlarına geldiğinde yapılacak işlemdir. Bunun icin bir çözüm yolu n x n bir maske için filtrelemeyi, resmin sınırlarında (n-1)/2 piksel önce bitirmektir. Bu işlem sonucu elde edilen filtrelenmiş görüntü orjinalinden küçük olur. Filtrelenmiş görüntünün orjinaliyle aynı büyüklükte olması için satır ve sütun kopyalama veya 0 yada başka bir gri seviye değeriyle resim etrafına takviye yapma yoluna gidilir. Ayrıca filtrelemede en iyi sonucu almak için filtre maskesini mümkün olduğu kadar küçük tutulur ve filtre merkezinin dışarı çıkmasını sınırdan (n-1)/2 piksel kadar önce sınırlandırılmalıdır.

3.4.2 Yumuşatma filtreleri

Ortalama filtre (averaging filter)

Yumuşatma filtrelerinde bir piksel için sonuç, komşulardaki piksellerin ortalaması alınarak elde edilir. Bu filtrelemeye ortalama filtre (average filters) adı da verilir. Bu filtreleme işleminin temelinde görüntü içindeki herbir piksele filtre maskesiyle tanımlanan, komşulardaki gri seviye değerinin ortalamasının atanmasıdır. Bu işlem sonucunda gri seviyedeki keskin geçişler azaltılmış olur.

Normalde rastgele gürültüler gri seviyedeki keskin geçişler şeklindedir ve bu filtreleme sonucu gürültü seviyesi düşer. Bununla beraber görüntü içerisinde kenarlar daha bulanık görünür. Şekil 3.7'de iki yumuşatma filtresi görülmektedir.

1/9

1	1	1
1	1	1
1	1	1

(a)

1/16

1	2	1
2	4	2
1	2	1

(b)

Şekil 3.7. iki farklı yumuşatma filtresi [20]

Şekil 3.7(a)'daki filtre normal bir ortalama işlemi yaparken, Şekil 3.7(b)'deki filtre ağırlıklı ortalama işlemi yapmaktadır.

Ağırlıklı ortalama filtreleme uygulamak için Eş.3.14 kullanılır [18].

$$g(x, y) = \frac{\sum_{i=-a}^{a} \sum_{j=-b}^{b} w(i, j) f(x+i, y+j)}{\sum_{i=-a}^{a} \sum_{j=-b}^{b} w(i, j)} \qquad (3.14)$$

Burada *M,N* büyüklüğünde *f* görüntüsü için x=0,1,2.....,M-1, y=0,1,2.....,N-1 ve a'da tekdir. n x n ortalama filtrede n büyüdükçe görüntü daha belirsiz ve bulanık bir hale gelir.

Sıra statistiği Filtreleri (Order-statistic Filters)

Bu filtrelemenin en bilinen örneği Median filtre'dir. Median filtre adından da anlaşıldığı gibi filtrelenmiş görüntüde, filtre merkezindeki piksele komşu piksellerdeki değerlerin ortancasını atamaktadır. Median filtre gürültü azaltma bakımından oldukça popüler bir filtredir. Median filtre tuz biber görüntüsü olarak bilinen gürültüye karşı etkilidir.

Bir noktaya median filtre uygulamak için öncelikle komşu piksellerdeki gri seviye değerleri sıraya sokulur ve 3x3'lük komşulukta beşinci değer, 5x5'lik komşulukta onüçüncü değer median değeri olarak elde edilir.

15	25	25
25	10	25
25	30	80

Şekil 3.8. 3X3 lük bir resim altbölümü

Şekil 3.8'deki 3x3 altresim bölümü sıraya sokulursa 10,15,25,25,25,25,25,30,80 olduğu görülür. Buradaki median 25'dir. Böylece merkez piksele, komşularıyla en benzeşen gri seviye değeri atanmış olur.

Median filtre, sıra istatistiği filtresi içinde en kullanışlı olanıdır . Ancak uygulamada eger seçim en büyük yüzdelik seviye değerine göre yapılırsa Max filtre, en düşük yüzdelik seviyeye göre yapılırsa Min filtre olarak kullanılabilir. Böylece resimdeki beyaz noktalar veya siyah noktalar baskın olacak şekilde işlem uygulanabilir.

3.4.3 Keskinleştirme filtreleri

Keskinleştirmenin temel amacı görüntüdeki detayları açığa çıkarmak veya bulanıklıkları düzeltmektir. Yumuşatma filtreleri yapı itibariyle integral işlemine benzetilebilir. Keskinleştirme ise görüntü düzleminin diferansiyelini alma gibi düşünülebilir. Temelde görüntü düzlemi üzerinde diferansiyal işleminin cevap etkinliği, büyük oranda görüntü üzerinde işlem gören noktaların süreksizliğine (kenar noktası vb.) bağlıdır. Böylece görüntü üzerinde türev işlemi, kenar gibi gürültü gibi süreksiz noktalarda vurguyu artırırken, düzenli değişen gri seviye bölgelerini belirsizleştirir.

Keskinleştirmenin Temeli

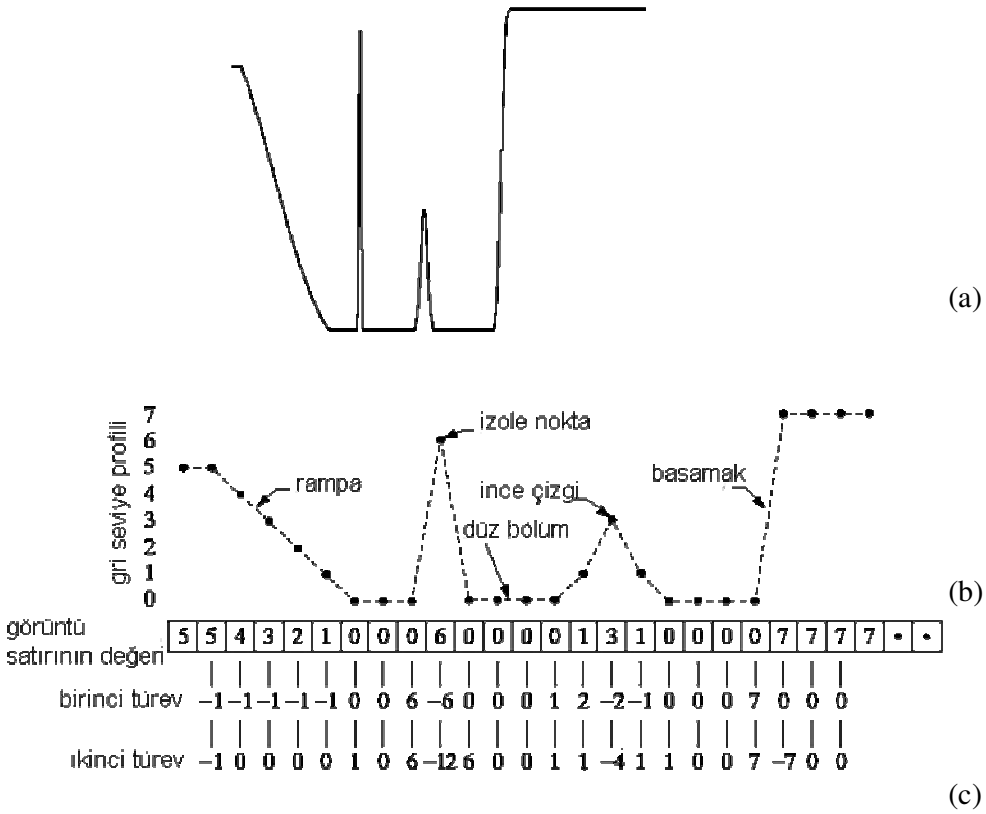
Tek boyutlu bir *f(x)* fonksiyonu için türev;

$$\frac{\delta f}{\delta x} = f(x+1) - f(x) \tag{3.15}$$

dir.İkinci türev ise Eş. 3.16 ile elde edilir.

$$\frac{\delta^2 f}{\delta x^2} = f(x+1) + f(x-1) - 2f(x) \tag{3.16}$$

Bu türetmenin görüntüde düz gri seviye bölgesinde, basamak ve rampa bölgelerde nasıl uygulandığı Şekil 3.9'da gösterilmiştir [18]. Şekil 3.9(a) bir görüntüye ait herhangi bir satırının profilidir. Eş.3.15 ve Eş.3.16 bu görüntü satırı için uygulanırsa, işlem sonuçları aşağıda görüldüğü gibi olur.



Şekil 3.9. Bir görüntü satırı için birinci ve ikinci türev

Şekil 3.9(b)'de rampa, izole gürültü noktası, basamak ve düz bölgeler bariz olarak belirlidir. Bu satırın sayısal değeri şekildeki gibidir.

Resim satırına birinci ve ikinci türev soldan sağa doğru uygulandığında şekil 3.9(c)'de görüldüğü gibi birinci türev rampa boyunca sıfırdan farklıyken, ikinci türev sadece rampanın başında ve sonunda sıfırdan farlıdır [18]. Genelde gri seviye görüntü içinde kenarlar bu tür bir yapı (rampa) gösterdikleri için birinci türev kenarları kalın olarak elde ederken, ikinci türev daha iyi sonuç verir. İzole gürültü noktası için ikinci türevin cevabı daha belirgindir. Aynı durum ince çizgi içinde

benzerlik gösterir. Bu sonuçlardan da görülmektedir ki, birinci türev kalın kenarlar üretirken, ikinci türev izole nokta (gürültü), ince çizgi gibi detaylar için daha belirgin sonuçlar üretir. Basamakta ise birinci türev daha güçlü bir cevaba sahipken ikinci türev basamakta çift cevap vermektedir. Uygulamada ikinci türev ince detaylarda daha iyi sonuçlar veresi nedeniyle daha çok tercih edilir. Görüntü geliştirmede daha iyi sonuçlar almak için bu metotlar beraber de kullanılmaktadır.

Laplacian Filtre

f(x,y) için laplace operatörünün yazılımı Eş. 3.17'de görülmektedir

$$\nabla^2 f = \frac{\delta^2 f}{\delta x^2} + \frac{\delta^2 f}{\delta y^2} \tag{3.17}$$

Dijital görüntü işlemede kullanmak için yukarıdaki denklemi ayrık formda ifade edersek ve iki değişken için ikinci türevi ayrı ayrı yazarsak, x - doğrultusu için ikinci türev Eş. 3.18 dir [19].

$$\frac{\delta^2 f}{\delta x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y) \tag{3.18}$$

Aynı şekilde y – doğrultusu için Eş. 3.19 dur [19].

$$\frac{\delta^2 f}{\delta y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y) \tag{3.19}$$

Eş.3.18 ve 3.19, Eş.3.17'de yerine konulursa, Eş. 3.20 elde edilir.

$$\nabla^2 \mathrm{f} = \left[\mathrm{f}(\mathrm{x}+1,\mathrm{y}) + \mathrm{f}(\mathrm{x}-1,\mathrm{y}) + \mathrm{f}(\mathrm{x},\mathrm{y}+1) + \mathrm{f}(\mathrm{x},\mathrm{y}-1) - 4\,\mathrm{f}(\mathrm{x},\mathrm{y})\right] \tag{3.20}$$

0	1	0
1	-4	1
0	1	0

(a)

0	-1	0
-1	4	-1
0	-1	0

(b)

$\pm 90^0$

1	1	1
1	-8	1
1	1	1

(c)

-1	-1	-1
-1	8	-1
-1	-1	-1

(d)

$\pm 45^0$

Şekil 3.10. Farklı laplacian maskeler

Şekil 3.10'da görülen 4 laplacin maske $\pm 90^0$ ve $\pm 45^0$'lik sınırları belirginleştirmek için kullanılır. Laplacian işlemi basitçe, orjinal resimle işlenmiş resmin toplamasıyla elde edilir. Eğer laplacian filtrenin merkez katsayısı negatif ise orjinal görüntüde filtrelenmiş görüntü çıkarılır. Merkez katsayı pozitifse tersi uygulanır. Bu durum Eş.3.21 ile ifade edilir [19].

$$g(x,y) = \begin{cases} f(x,y) - \nabla^2 f(x,y) & \text{Eger merkez katsayı negatifse} \\ f(x,y) + \nabla^2 f(x,y) & \text{Eger merkez katsayı pozitifse} \end{cases} \tag{3.21}$$

Bu işlemin ana nedeni laplacian operatörü sonunda görüntü içerisindeki sınırlar harici bölgelerin iyice belirsizleşmesidir. Eğer orjinal görüntüyle filtrelenmiş görüntüyü toplarsak böylelikle hem sınırlar belirginleşmiş olur, hem de diğer bölgelerde görünür kalır.

Yukarıdaki maskelerden Şekil 3.10 (a)'yı Şekil 3.11'deki görüntü alt bölümü için Eş 3.20 ile uygulandığında elde edilen sonuç Eş 3.22'dedir.

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

Şekil 3.11 Resim altbölümü

$$\nabla^2 f = 4z_5 - (z_2 + z_4 + z_6 + z_8) \tag{3.22}$$

Şekil 3.10 (c)'yi kullanarak diagonal komşuluklarda dahil edilirse Eş.23 elde edilir.

$$\nabla^2 f = 8z_5 - (z_1 + z_2 + z_3 + z_4 + z_6 + z_7 + z_8 + z_9) \tag{3.23}$$

Fakat Laplacian operatörü, gürültülere hassas olması, genliğinin çitf kenarlar üretmesi ve kenar doğrultusunu belirleyememesi yüzünden yumuşatma ile beraber kullanılır.

$$\Gamma(\mathrm{r}) = -\mathrm{e}^{-\frac{\mathrm{r}^2}{2\sigma^2}} \tag{3.24}$$

Bunun için gaussian fonksiyonundan yararlanılır. Eş.3.24'deki gaussian fonksiyonunda $r^2=x^2+y^2$ ve σ' standart sapmadır. Bu fonksiyon görüntünün bulanıklaşmasına neden olur. Bulanıklık seviyesini σ' nın değeri belirler. Γ fonksiyonuna Laplacian operatörü uygulanırsa (r' ye göre) Eş.3.25 elde edilir [19].

$$\nabla^2\Gamma(\mathrm{r}) = -\left[\frac{\mathrm{r}^2 - \sigma^2}{\sigma^4}\right] \mathrm{e}^{-\frac{\mathrm{r}^2}{2\sigma^2}} \tag{3.25}$$

Eş.3.25 Gauss'un Laplace' i (LoG) olarak bilinir. Şekil 3.12'de LoG ile üretilmiş 5 X 5 lik bir maske görülmektedir.

0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

Şekil 3.12. LoG filtre örneği

Gradient Filtre

Görüntü işlemede birinci türev gradient'in genliği alınarak uygulanır. Bir f(x,y) fonksiyonunun (x,y) kordinatlarındaki gradyantı Eş.3.26 da görüldüğü gibi iki boyutlu sütun vector olarak tanımlanır ve genliği Eş.3.27 ile elde edilir.

$$\nabla \mathrm{f} = \begin{bmatrix} \mathrm{Gx} \\ \mathrm{Gy} \end{bmatrix} = \begin{bmatrix} \frac{\delta \mathrm{f}}{\delta \mathrm{x}} \\ \frac{\delta \mathrm{f}}{\delta \mathrm{y}} \end{bmatrix} \tag{3.26}$$

$$\begin{aligned} \nabla f &= mag(\nabla f) \\ &= \left[Gx^2 + Gy^2\right]^{1/2} \\ &= \left[\left(\frac{\delta f}{\delta x}\right)^2 + \left(\frac{\delta f}{\delta y}\right)^2\right]^{1/2} \end{aligned} \tag{3.27}$$

Eş.3.27'nin uygulama zorluğu nedeniyle görüntü işlemede gradient vektörün genliği için karesini alıp sonra kökünü bulmak yerine, Eş.3.28 ile direk mutlak değerlerini kullanmak daha uygundur. [19].

$$\nabla f \cong |Gx| + |Gy| \tag{3.28}$$

Şekil 3.10'da verilen bir görüntünün 3x3'lük bölgesi için z_5, merkez noktası f(x,y)' i, z_1 f(x-1, y-1)'i ve diğerleri de benzer şekilde diğer noktaları ifade etsin. Keskinleştirme işleminde gösterildiği gibi Eş.3.15 (birinci türev) uygulanırsa sonuç yaklaşık olarak Eş.3.29 ile elde edilir [19].

$$Gx = (z_8 - z_5) \text{ ve } Gy = (z_1 - z_5) \tag{3.29}$$

$$Gx = (z_9 - z_5) \text{ ve } Gy = (z_8 - z_6) \tag{3.30}$$

Eş.3.30 ise Roberts cross – gradient operatorü olarak bilinir. Bu durumda gradient Eş.3.31 ile eldedilir [18].

$$\nabla f = \left[(z_9 - z_5)^2 + (z_8 - z_6)^2\right]^{1/2} \tag{3.31}$$

Genlik için karesini alıp karekök almak yerine mutlak değerlerini kullanılısa Eş.3.32 elde edilir.

$$\nabla f \approx |z_9 - z_5| + |z_8 - z_6| \tag{3.32}$$

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

(a)

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

(b)

Şekil 3.13. Gradient filtre örneği

Şekil 3.13'de 3x3'lük 2 filtre maske görülmektedir. Bu maskeler Sobel operatörü olarak bilinir ve sırasıyla dikey ve yatay çizgileri belirginleştirmek için kullanılır. 3X3 lük maskeler için gradient işlemi Eş.3.33 ile uygulanır.

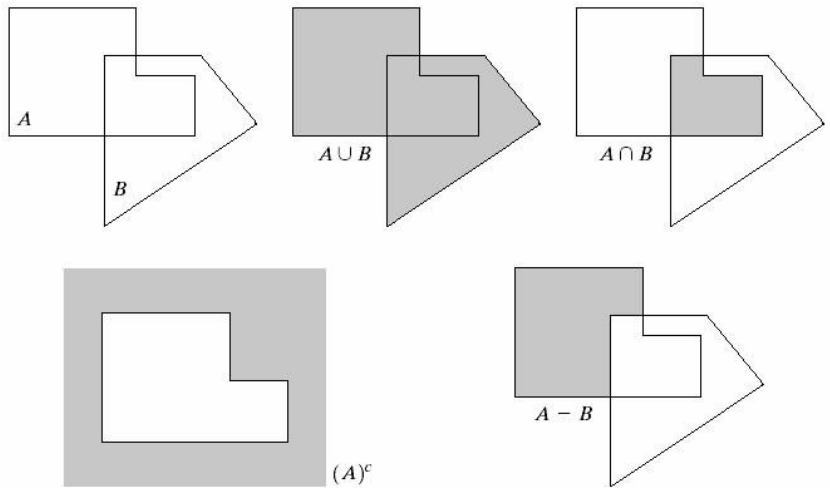
$$\nabla f \approx |(z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)| + |(z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)| \tag{3.33}$$

4. GÖRÜNTÜ YAPISAL İŞLEMLERİ

4.1 İkili Düzey (Binary) Görüntü Yapısal İşlemleri

İkili resim yapısında, resim renk değerleri 1 bit ile ifade edilir. Dolayısıyla 2^1=2 renk değeri (0 siyah 1 beyazı ifade eder) sunar. Genellikle '0' arka plan piksellerini,1 de nesne piksellerini oluşturur. İkili resim yapısı resimdeki bazı özelikleri çıkarmak için kullanılabilir. 8 bit gri düzey resim ikili resim yapısına dönüştürüldükten sonra değişik işlemlere tabi tutularak istenen alan, kütle merkezi vb. nesneye ait özelikler resimden çıkartılabilir. Şekil 4.1'de ikili resim üzerinde uygulanan mantıksal işlemler aşağıdaki gibidir ve resim üzerinde gösterimi Şekil 4.1 da görülmektedir [19].

A or B = A U B
A and B =A ∩ B
Not A =A^C
A and (not B) =A-B



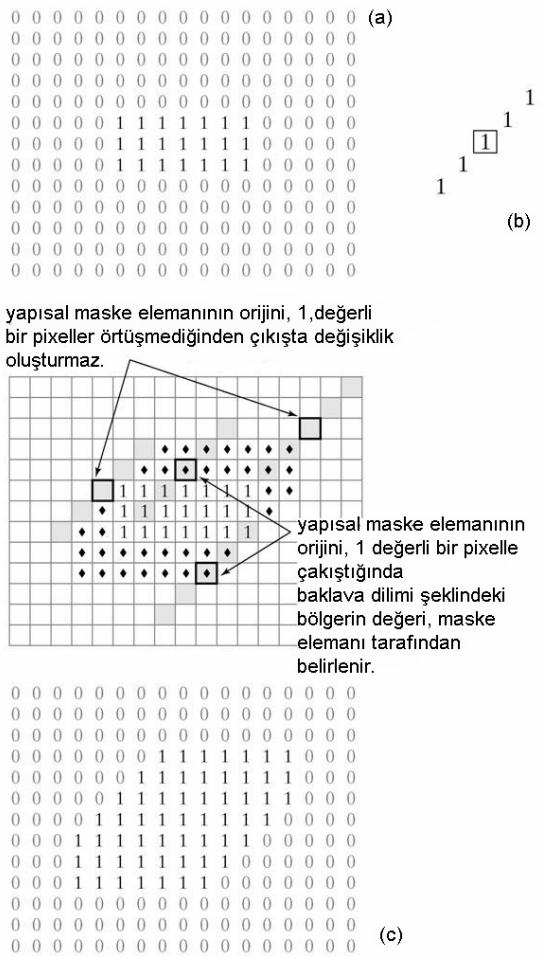
Şekil 4.1. İkili resim üzerinde uygulanan mantıksal işlemlerin gösterimi

İkili düzey resimde her iki piksel ve belirtilen komşuluk ilişkisine göre komşuları, yapısal eleman ile yukarıda belirtilen mantıksal işlemlere tabii tutularak resim

içindeki nesnelerle ilgili bilgilerin edinilmesinde daha doğru sonuçlar elde edilmesini sağlar.

4.1.1 Genişleme (dilation)

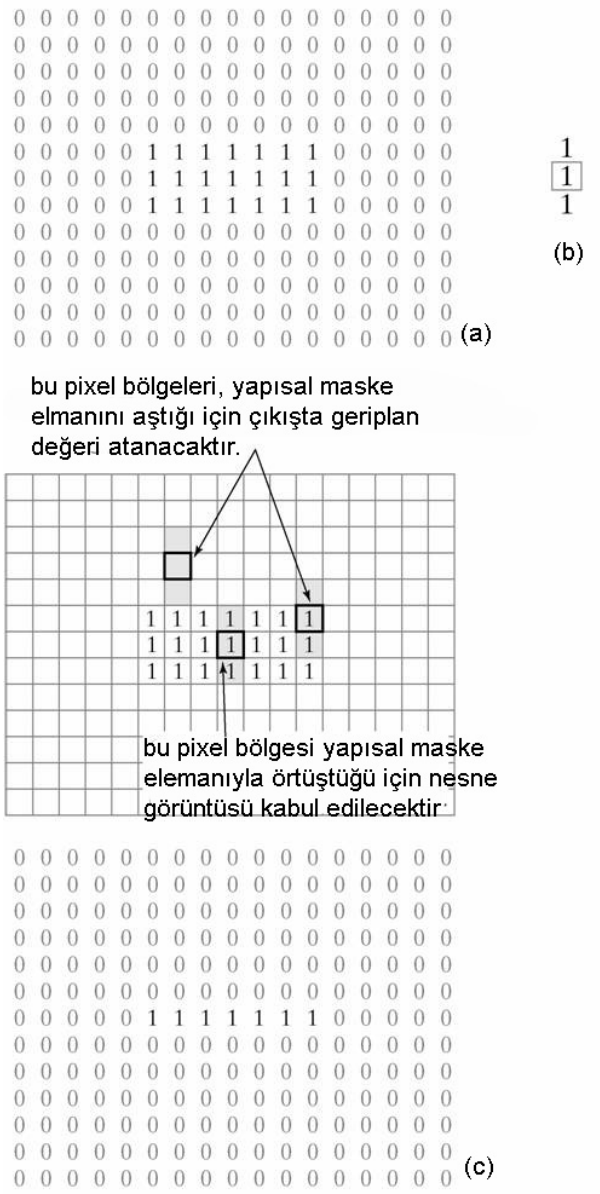
Genişleme kalınlaştırma, büyütme işlemidir. Bu büyüme işleminin şeklini yapısal eleman tayin eder. Şekil 4.2 (a)'da verilen resim bölümü için, (b)'deki yapısal elemanın nasıl uygulandığı ve (c)'de işlem sonucu görülmektedir [19]. Yapısal elemanın şekli elde edilecek sonucu belirler. Yapısal elemanı, baklava dilimi, disk, çizgi, kare, dikdörtgen gibi şekillerde tanımlanabilir veya bizim istediğimiz özel bir şekle sahip olabilir. Böylece nesne kenarları istenilen forma sokulabilir.



Şekil 4.2. Genişleme işlemi

4.1.2 Aşınma (erosion)

Aşınma işlemi ise genişlemenin tersidir. Aşınmada da genişlemede olduğu gibi aşınmanın şeklini yapısal eleman belirler. Şekil 4.3 (a)'de verilen resim bölümü için, (b)'deki yapısal elemanın orijinal görüntüye nasıl uygulandığı ve (c)'de sonucu görülmektedir [19].

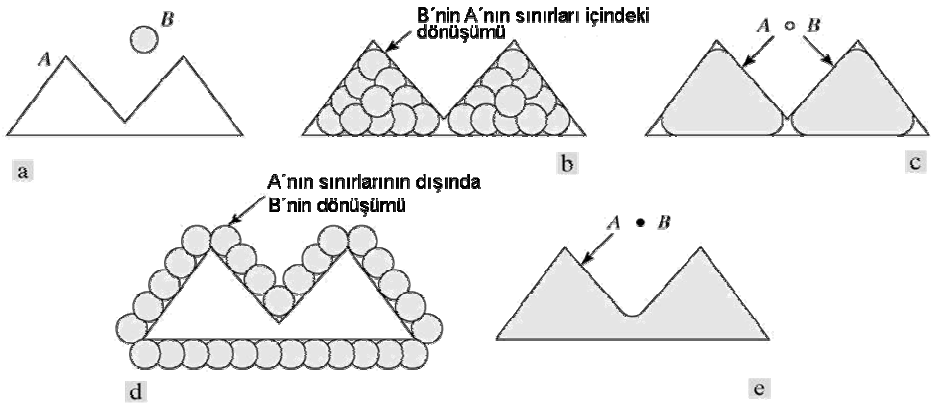


Şekil 4.3. Aşınma işlemi

4.1.3 Açma ve kapama (opening and closing)

Pratikte genişleme ve aşınma işlemleri çoğunlukta beraber, değişik kombinasyonlarla kullanılırlar. Yani görüntü farklı veya aynı yapısal maske elemanlarıyla bir dizi genişleme ve/veya aşınma tabii tutulabilir.

Açma işlemi kendisini genişlemenin takip ettiği basit bir aşınma işlemidir. Şekil 4.4'da opening ve closing işlemleri görülmektedir [19]. Şekil 4.4 (a)'da A nesneyi, B de yapısal elemanı göstermektedir. Şekil 4.4 (b)'de opening işleminin uygulanışı, Şekil 4.4 (c)'de ise sonucu görülmektedir. Açma uygulandıktan sonra beyaz kalan kısımlar artık nesnenin dışındadır. Genişleme ve Aşınma işleminde olduğu gibi, sonucu maske elemanı belirler. Maskenin örtüştüğü nesne bölümleri düzelirken, maskenin yapısını karşılamayan yerler silinir. Şekil 4.4 (c)'de üçgenler içindeki boyalı olmayan kısımlara disk şeklindeki yapısal eleman örtüşmediğinden bu kısımlar silinmiştir. Böylece nesne hatları yumuşar, ince bağlantılar kopar ve ince çıkıntılar temizlenir.



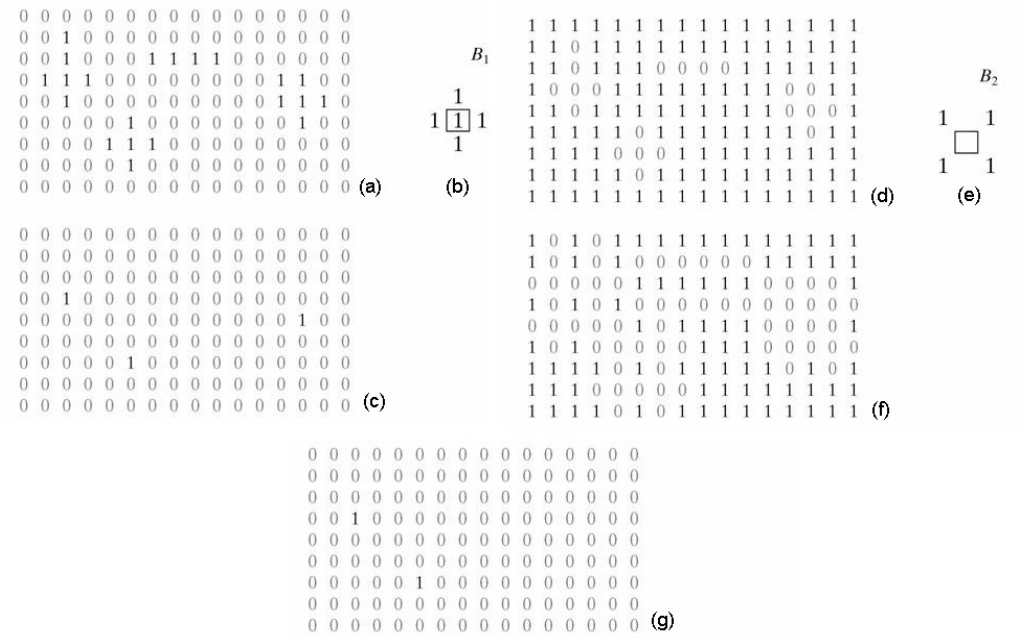
Şekil 4.4. Açma ve Kapama işlemi

Kapama işlemi ise açmanın tersine kendisini aşınmaın takip ettiği bir genişleme işlemidir. Şekil 4.4 (d)'de görüldüğü gibi uygulanır ve sonucu Şekil 4.4 (e)'de dir. Kapama işlemi de açmaya benzer olarak nesne sınırlarını yumuşatır. Açmadan farkı ise yapısal elemandan küçük olan girinti kırık ve delikler kapatılır.

4.1.4 Hit Or Miss dönüşümü

Hit Or Miss dönüşümü özel bir piksel konfigurasyonuna (yapısal elemana) göre orjinal resmin piksellerinin, izole bir nesne pikseli mi veya bir çizgi parçasının bitiş pikseli mi olup olmadığını anlamada oldukça kullanışlı bir dönüşümdür.

Şekil 4.5 (b)'de B_1 yapısal elemanı için, Şekil 4.5 (a)'da ki resim bölümünün hit or miss uygulaması görülmektedir. Önce B_1 e göre resim işleme tabii tutulur ve yapısal elemanla örtüşen noktalar belirlenir. Sonuç Şekil 4.5 (c)'de dir. Sonra resmin ve yapısal elemanın (d) ve (e) de görülen tümleyenleri birbirleriyle işleme girer. Her iki işlemde elde edilen sonuçlar (c) ve (f)'nin kesişimi bize hit or miss dönüşümünün sonucunu verir. Sonuç Şekil 4.5 (g)'dedir [19]. Böylece orijinal görüntüde özel bir nesne yapısı veya özel noktaları bulabilir.



Şekil 4.5. Hit or miss işlemi

4.1.5 Yeniden yapılandırma

Yeniden yapılandırma işlemi diğer işlemlerdeki tek görüntünün yerine iki görüntü ve yapısal eleman gerekmektedir. Bu görüntülerden biri "Marker" olarak adlandırılır ve

içerisinde yeniden yapılandırma işleminin başlangıç noktasını bulundurur. Bu nokta görüntü içerisinde belirlemek istediğimiz nesneye ait bir nokta olabilir. Diğer görüntü ise "Maske" olarak adlandırılır ve yeniden yapılandırma işleminde kısıtlamaları belirler. Yapısal eleman ise komşuluk ilişkisini tanımlamak için kullanılır. Yeniden yapılandırma işlemi g' maske görüntüsü, f marker görüntüsü ve B yapısal elemanı için aşağıdaki iterasyonun yapılması ile uygulanır [19].

1. h_1' i marker görüntüsü olarak alınır.
2. Yapısal elemanı oluşturulur
3. $h_{k+1}=h_k$ olana kadar, $h_{k+1}=(h_k \oplus B) \cap g$ işlemini tekrar edilir.

Burada '$\oplus$' operatörü genişletme işlemini ifade etmektedir. Yeniden yapılandırmada dikkat edilmesi gereken en önemli nokta f' marker görüntüsünün, g maske görüntüsünden bir alt set olması zorunluluğudur.

4.2 Gridüzey Görüntü Yapılandırma İşlemleri

Bölüm 4.1 de anlatılan ikili düzey yapısal işlemlerinden hit or miss dönüşümü hariç, açma, kapama, aşınma ve genişleme işlemleri gri seviye görüntü üzerinde de uygulanabilir. Bu işlemler görüntüyü yeniden yapılandırma sağlayarak resim içindeki istenen bilgiyi elde etmeyi kolaylaştırır. Yapısal işlemler gri seviye görüntüye, ikili düzey yapısal işlemlerde olduğu gibi ikili düzey biçimdeki yapısal eleman ile uygulanırlar. Elde edilecek sonuçu bu yapısal elemanın şekli, büyüklüğü belirler.

4.2.1 Genişleme ve aşınma (dilation - erosion)

Gri seviye görüntünün yapılandırma elemanı ile genişleme işlemi, içerik olarak yapısal elemanı resim içindeki tüm noktalar için resimdeki gri seviye değerine göre dönüştürme işlemi gibi düşünülebilir. Genişleme işlemi resim içindeki her yerde yapısal eleman değeri ile piksel gri seviye değeri toplanarak sonucun maksimumunun alınması şeklinde uygulanır.

Bu işlem görüntü üzerendeki filtreleme işlemine benzetilebilir. En önemli fark, yapısal eleman içindeki rastgele bir (x_0,y_0) koordinatı için genişleme sonucunun maksimum alma işlemi dahil olabilmesi için yapısal elemanın (x_0,y_0) koordinatlarında 1 değerine sahip olması gerekliliğidir. Genişleme işlemi sonrasında yapısal elemanı şekline bağlı olarak resim içindeki karanlık detayların azaldığı veya yok olduğu ve çıktı görüntüsünün daha parlak olduğu bir sonuç elde edilir.

Gri seviye görüntüde aşınma işlemi ise, yapısal eleman değeri işlem gören pikseldeki gri seviye değerinden çıkarıldıktan sonra sonucun minimumunun alınmasıyla bulunur. Aşınma sonucu görüntü daha koyulaşırken, açık bölümlerin yok olur veya küçülür.

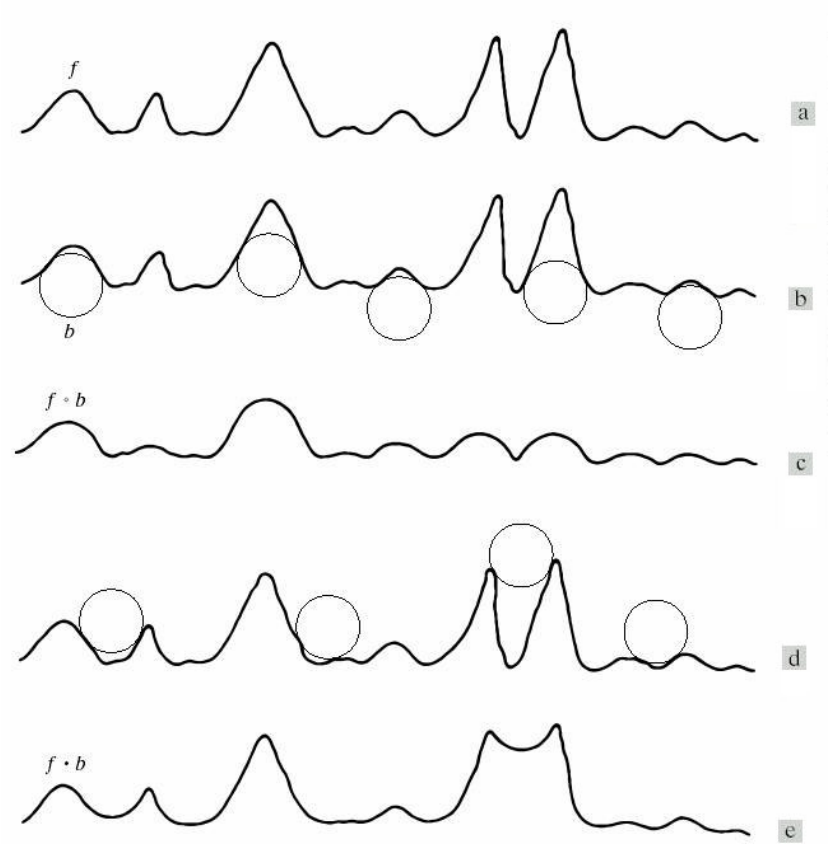
4.2.2 Açma ve kapama (opening – closing)

Açma ve Kapama işlemi ikili düzey işlemlerinde olduğu gibi genişletme ve aşındırma işlemlerinin uygulanmasıdır. Gri seviye görüntü için Açma işlemi, görüntünün yapısal eleman ile önce aşınmasını, ardından sonucun genişlemesini alma işlemlerini ifade etmektedir. Benzer şekilde Kapama işlemi ise görüntüye yapısal eleman ile genişleme sonrasında sonucun yine aynı yapısal eleman ile aşınma uygulanmasını ifade eder. Her iki işlemi geometrik olarak ifade etmek için herhangi bir görüntü koordinat ve gri seviye değerleri kullanılarak oluşturulan üç boyutlu bir yüzey şeklinde düşünülebilir. Bu görüntünün yapısal eleman tarafından açılma işlemi, yapısal elemanın görüntünün yüzeyinin altından kaydırılarak, yapısal elemanın tüm görüntü için çevrilmesidir. Açma işlemi, yapısal elemanın görüntü yüzeyinin altında kayarken ulaşabileceği en yüksek değerin alınması ile elde edilir.

Şekil 4.6 c'de bir resim satırının gri seviye profilinin, daire biçimindeki bir yapısal eleman ile açma işlem sonucu daire biçimli yapısal elemanın ulaşamadığı diğer bir ifade ile sığmadığı yerler (sivri çıkıntı) açma işlemi sonrasında silinmişlerdir. Açma işlemi genelde tüm gri seviyeleri ve büyük parlak bölgeleri bozmadan küçük parlak detayların silinmesi için kullanılır.

Kapamada ise Şekil 4.6 d' de görüldüğü gibi daire biçimli yapısal elemanı görüntü satırının gri seviye profilinin oluşturduğu sınır üzerinde kaydırılarak o satırdaki tüm noktalar için yapısal elemanın dönüşüm sonucu elde edilmiştir. Kapama işlemi, yapısal elemanın görüntü yüzeyinin üst sınırında kaydırılırken ulaşabileceği en düşük noktanın değerinin alınması ile elde edilir. Kapama işlemi sonucu Şekil 4.6 e'de de görüldüğü gibi yapısal elemanın erişebileceğinden daha küçük ayrıntılar bastırılmış olur.

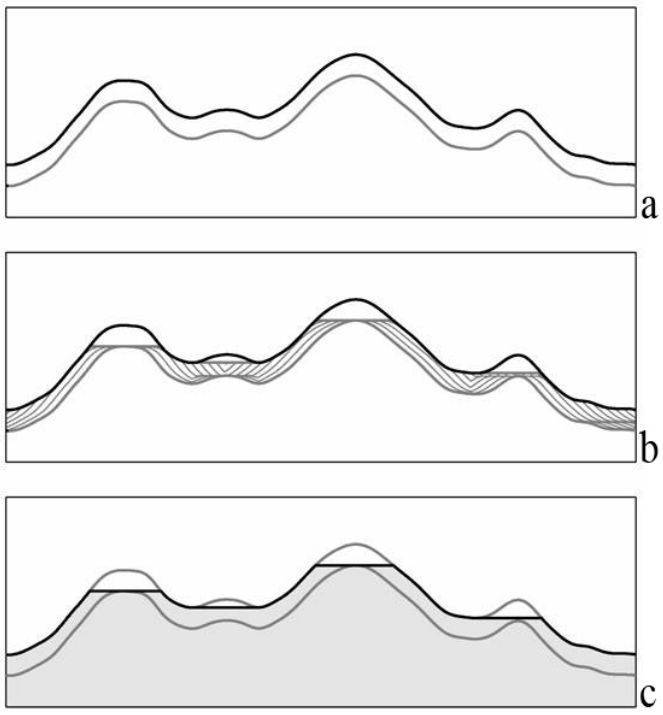
Uygulamada açma işleminin yapısal elemanın ulaşabileceğinden daha küçük parlak ayrıntıları, kapama işleminin ise yapısal elemanın ulaşabileceğinden daha küçük siyah ayrıntıları bastırma özelliği görüntü yumuşatma ve gürültüleri silme amacıyla beraberce kullanılırlar.



Şekil 4.6. a) resim satırı gri seviye profili, b) açma işlemi, c) açma işlemi sonucu, d) kapama işlemi, e) kapama işlemi sonucu [18]

4.2.3 Gri seviye yeniden yapılandırma

Gri seviye yeniden yapılandırma içinde ikili seviye yeniden yapılandırmadaki iterasyonun aynı uygulanır. Şekil 4.7 de bir görüntü satırı için gri seviye profilinin yeniden yapılandırması görülmektedir. Burada üst sınır maske satırını, alt sınır ise marker satırını ifade etmektedir. Şekil 4.7 a'da de görüldüğü gibi marker görüntüsü, maske görüntüsünden belirli bir sabitin çıkarılması ile elde edilmiştir. Ancak istenirse maskeyi aşmayacak şekilde herhangi bir marker görüntüsü de kullanılabilir.



Şekil 4.7. a) yeniden yapılandırma için marker ve maske görüntüleri, b) yeniden yapılandırma işlemi, c) yeniden yapılandırma işlem sonucu [19]

Her bir iterasyonda marker görüntüsündeki büyük çıkıntılar maskeye kadar açılarak yayılır. Yeniden yapılandırma sonucu Şekil 4.7 c'de görülmektedir. Buruda yüksek çıkıntılar bastırılırken aradaki küçük çıkıntılar yükseltilmiştir. Genelde maske görüntüsü olarak orijinal görüntü kullanılır.

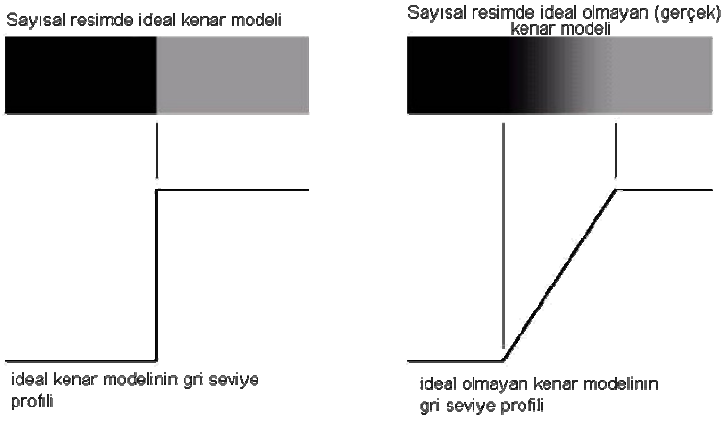
5. BÖLÜMLENDİRME (SEGMENTASYON)

Görüntü işlemedeki en önemli problemlerden biri görüntü içerisinden, nesnelerin bulunduğu alt bölümlerin belirlenmesi işlemidir. İnsanoğlu için çok kolay olan bu işlem görüntü işleme sistemleri için oldukça zordur. Bölümlendirme bir görüntüyü, bölge olarak adlandıran ve bu bölgede belirlenmek istenen nesne adayının olduğu parçalara ayırmak olarak tanımlanabilir. Bir başka ifadeyle bir görüntüyü kendisini oluşturan bölge ve nesnelere ayırma işlemidir. Bölümlendirme bir görüntüyü anlamlandırmak için önemli bir aşamadır.

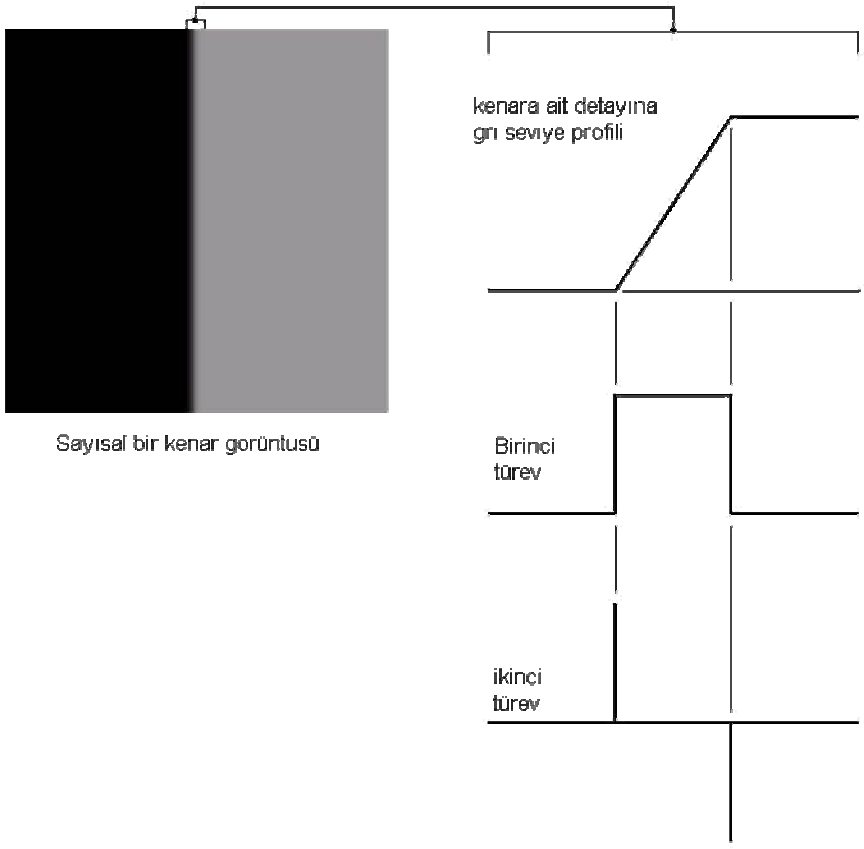
Bölümlendirme algoritması, görüntüdeki gri seviye değerinin 2 ana özelliğini temel alır: Benzerlik ve süreksizlik. Süreksizlik görüntü içerisinde kendini gri seviye değerinde normal olmayan (Ör.: neske kenarı gibi) değişiklik şeklinde gösterir. Benzerlikte ise görüntü içerisinde gri seviye değerleri birbirine benzeyen kısımları bölgelere ayırma mantığı yatar. Thresholding, region growing, region splitting, watershed bu kategorideki örnek metotlar olarak verilebilir. Bu tezde bu metotlardan eşikleme, kenar belirleme ve watershed bölümlendirme uygulanmıştır.

5.1 Kenar Belirleme

Kenar belirleme işlemi için dijital bir görüntüde kenarın nasıl göründüğüne yakından bakılırsa bir sayısal resimde ideal bir kenarla, normal bir kenarın detayı Şekil 5.1'deki gibidir [18]. İdeal olmayan kenardaki rampanın eğimi görüntünün bulanıklık (blur) seviyesi ile orantılıdır. Pratikte elde edilen kenar görüntüsünün bulanık ve dağınık olması, örnekleme, yetersiz aydınlatma, optik ve kullanılan diğer donanımların ideal olmaması nedenlerinden ve hatalarından kaynaklanmaktadır. Bunun sonucunda nesne kenarlarındaki gri seviye değerinde rampa şeklinde bir değişim görülür. Şekil 5.2'de ideal olmayan nesne kenarının birinci ve ikinci türevleri görülmektedir [18].



Şekil 5.1. Kenar profil modeli



Şekil 5.2. Sayısal kenarın birinci ve ikinci türevi

Bu işlemden şu sonuç çıkmaktadır. Bir görüntünün birinci türevinin genliği (gradienti), görüntü içerisindeki o noktada herhangi bir kenarın varolup olmadığını

kontrol etmek için kullanılabilir. İkinci türev ise kenar pikselinin siyah bölgede mi yoksa beyazda mı olduğunu anlatmak için kullanılabilir. Ayrıca ikinci türev sonucu iki ek özellik daha oluşmaktadır. Bunlar; kenar bölgesinde iki değer üretilmektedir ki bu istenmeyen bir durumdur. Diğeri ise; kenarın tam orta noktasında ikinci türev sıfır olmaktadır ki bu duruma "zero-crossing" özelliği denir ve sınırı tek ince bir çizgi olarak ifade etmek istenildiğinde kolaylık sağlar.

5.2 Kenar Birleştirme ve Sınır Belirleme

Kenar belirlemede kullanılan biraz önce de bahsedilen piksel değerlerindeki süresizlikler, pratikte ideal olmayan donanımlar, yetersiz aydınlatma, ortam gürültü v.b nedenlerden dolayı kenar belirlemede nadiren tatminkar sonuçlar verir. Dolayısıyla kenar belirleme işlemlerini kenar birleştirme işlemleri takip eder.

5.2.1 Bölgesel işleme

Kenar piksellerinin tanımlanması için görüntüsündeki her (x,y) noktasında küçük komşuluk ilişkileri (3 x 3 veya 5 x 5) analiz edilmişti. Benzer olan kenar piksellerinin birleştirilmesinde kullanılan iki temel özellik;

1. Gradient operatörünün genliği
2. Gradient vektörünün yönü

dür. (x_0, y_0) (x, y) nin komşuluğundaki kenar pikselidir ve genliği (x,y) ile benzer ise, Eş. 5.1 sağlanır [18].

$$\left| \nabla f(x, y) - \nabla f(x_0, y_0) \right| \leq E_T \qquad (5.1)$$

Burada E_T negatif olmayan eşik değeridir. Gradient vektörün açısı x' eksenine göre (x_0, y_0) daki açısı, Eş 5.2 ile elde edilir. [18].

$$\alpha(x, y)\tan^{-1}\left(\frac{G_y}{G_x}\right) \quad (5.2)$$

$$\left| \alpha(x, y) - \alpha(x_0, y_0) \right| < A_T \quad (5.3)$$

Eş.5.3'de A_T negatif olmayan açı eşik değeridir. (x, y) noktasındaki kenar, o noktadaki gradient vektöre dik doğrultudadır [18].

Eğer (x,y) nin belirtilen yöndeki komşusu Eş 5.2 ve Eş 5.3'deki kriterlerini sağlıyorsa bu iki nokta bir hat oluşturur. Bu işlem ile görüntü içindeki tüm pikseller için kenar pikselleri birleştirilir.

5.2.2 Hough dönüşümü ile genel işleme

$y_i = ax_i + b$ noktasını kesen bir doğru denklemi ve bu denklemi sağlayan bir (x_i,y_i) nokta olsun. (x_i, y_i) noktasından pek çok doğru geçer ve her doğruda farklı a, b değerleri için denklemi sağlar. Bu denklem $b = -x_i\, a + y_i$ şeklinde yazılırsa, bu denklem a b düzleminde (parametre uzayı) sadece bir (x_i,y_i) değeri için tek bir doğru verir. Ayrıca ikinci bir (x_j,y_j) noktası için a –b düzleminde ikinci bir doğru elde edilir ve bu iki doğru a',b' noktasında kesişirler. a',b' değerleri (x_i,y_i) ve (x_j,y_j) noktalarını üzerinde barındıran doğrunun sırasıyla eğim ve kesişim değerlerini ifade eder.

Hough dönüşümü eğim ve kesişim noktalarını, akümülator hücresi adı verilen parametre uzayında eğim ve kesişim değerleri (a_{max} ve a_{min}) ve (b_{max} b_{min}) aralığında beklenen alt parçalara böler. (a_i , b_j) parametre uzayı ise ilgili akümülatör hücre koordinatı ve değeri sırasıyla (i , j) , A (i , j) dir. Görüntü içindeki her nokta için (max-min) aralıklarında a, b değerleri elde edilir. Bu a_p ve b_q değerleriyle A(p, q) =A(p, q) +1 elde edilir. Bunun sonucunda A (i , j) deki Q değerine sahip (i, j) noktaları için x - y düzleminde $y = a_i\, x + b_i$ doğrusu elde edilir. Bu ifadeyle verilen

denklemde dikey doğrultuda eğimin sonsuza yaklaşması nedeniyle doğruyu Eş.5.4 ile polar koordinatlarla ifade etmek kolaylık sağlar.

$$x\cos\theta + y\sin\theta = \rho \tag{5.4}$$

Hough dönüşümünün uygulama adımları;

i. Görüntünün gradientini hesapla, ve eşikleme işlemiyle ikili görüntüyü elde edilir.
ii. ρ - θ düzleminde alt bölmeleri tanımlanır.
iii. Yüksel piksel konsantrasyonlu akümülator hücreleri için sayi adedini incelenir.
iv. Seçilen akümülator hücresindeki pikseller arasındaki bağlantı olup olmadığını (yani süreklilik) incelenir.

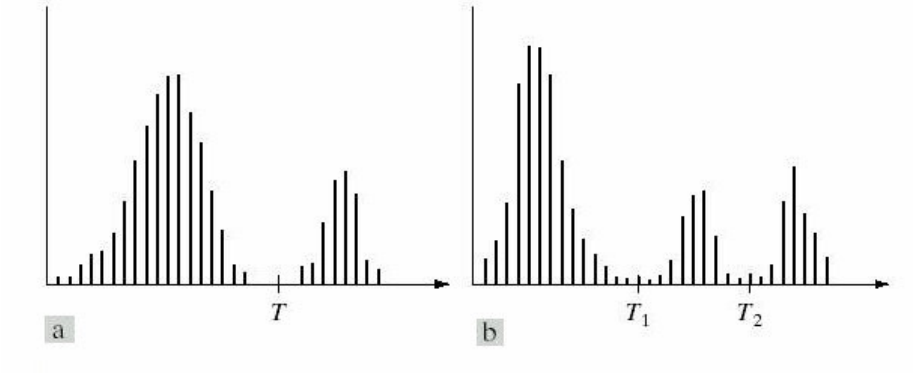
5.3 Eşikleme (Thresholding)

Özellikleri ve uygulamadaki kolaylığından ötürü eşikleme bölümlendirme uygulamaları içinde merkezi bir konuma sahiptir.

5.3.1 Eşiklemenin temeli

Koyu renk bir gri plan üzerinde açık renk bir f (x, y) görüntüsünün histogramı Şekil 5.3(a)'da da görüldüğü gibi iki baskın moddadır. Nesneyi geri plandan ayırmanın en bariz yolu bu iki modu birbirinden ayıracak bir T threshold değeri seçmektir. Bu görüntü içinde $f(x,y) > T$ yi sağlayan noktaları nesneyi, $f(x,y) \leq T$ ise geri planı ifade eder.

Şekil 5.3(b) ise daha genel durumu ifade eder. Burada histogram 3 baskın modadır (2 farklı tip nesne vb.). Bu durumda ise çoklu seviye eşikleme kullanılarak $T_1 \leq (x,y) \leq T_2$ bir nesneyi $f(x,y) > T2$ diğer nesnesi ve $f(x, y) < T1$ ise geri planı sınıflandırmak için kullanılabilir.



Şekil 5.3. Histogram görüntüsü

Genelde çoklu eşiklendirme gerektiren uygulamaların çoğu bölge büyültme yöntemleri kullanılarak çözülür.

$$T = T\,[x,y,\, ng(x,\, y)\,,\, f(x,y)\,] \tag{5.5}$$

Eş 5.5'de *f(x,y), (x,y)* noktasının gri değerini, *ng(x,y)* de bu noktanın sahip olduğu bölgesel özelliği (komşularının ortalama gri seviyesi v.b) göstersin. Bu durumda eşikleme işlemi Eş 5.5'in resimdeki tüm noktalar için test edilerek uygulanması şeklinde görülebilir.

Eşiklenmiş bir resim *g(x, y)* Eş 5.6 ile ifade edilir.

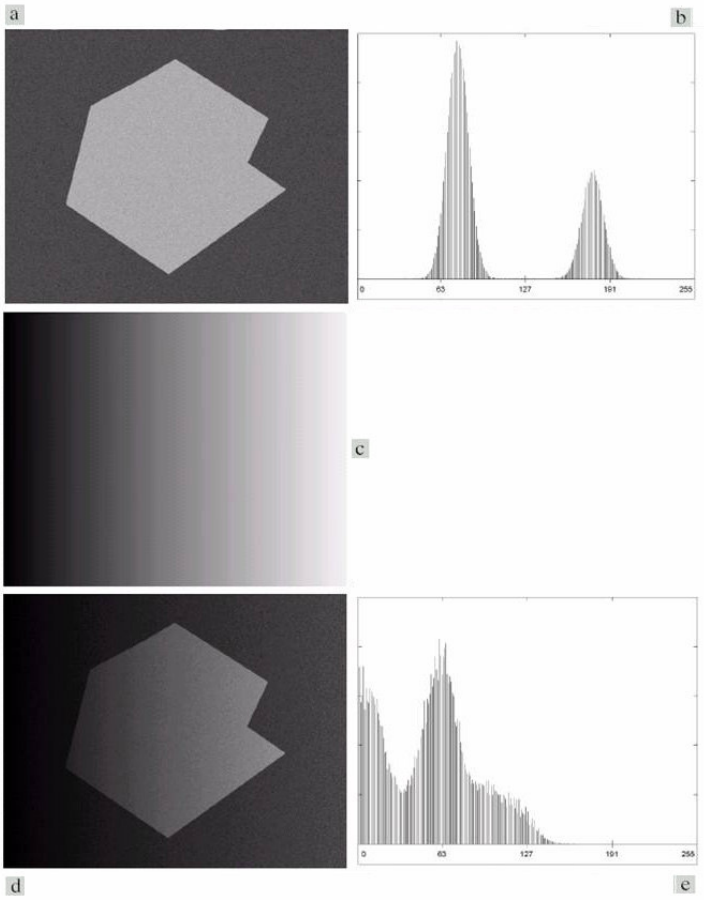
$$g(x,y)=\begin{cases}1 & \text{Eger } f(x,\, y) > T \text{ ise} \\ 0 & \text{Eger } f(x,\, y) \le T \text{ ise}\end{cases} \tag{5.6}$$

Böylece nesneler "1" (veya herhangi bir gri seviye) ile, geriplan "0" (veya nesnelere atanmamış herhangi bir gri seviye) ile etiketlenmiş olur.

T yalnızca *f(x, y)*' ye bağlı olursa (yalnızca gri seviye değerlerine) eşikleme "global eşikleme" olarak adlandırılır. T hem *f(x,y)*, hem de *ng(x,y)* bağlı olursa eşikleme yerel olarak adlandırılır. Ayrıca eğer T görüntü tanım bölgesi koordinatları olan x ve y' ye bağlı olursa eşikleme "dinamik" veya "uyarlanabilir" olarak adlandırılır [18].

5.3.2 Eşiklemede aydınlatmanın rolü

Şekil 5.4 (a)'da *yan(x,y)* yansıma ve *ay(x, y)* aydınlatma bileşenini içeren f (x,y) görüntüsü görülmektedir. Bu görüntünün histogramı Şekil.5.4 (b)'deki gibidir. Bu görüntüde iki histogram modu arasında bir T eşik değeri seçilerek rahatlıkla bölümlendirme yapılabilir. Aynı görüntü, Şekil 4.4 (c)'deki aydınlanma fonksiyonu ile çarpılarak Şekil 5.4 (d)'deki görüntü elde edilir ve bunun histogramı Şekil 5.4 (e)' de görülmektedir. Bu durumda artık bölümlendirme tek bir T' eşik değeriyle yapılamamaktadır.



Şekil 5.4. Aydınlatma fonksiyonun etkisi [18]

f(x,y) görüntüsünü Eş.5.7' daki gibi *ay(x, y)* aydınlatma ve *yan(x, y)* yansıma bileşenlerine ayrılırsa; Eş 5.7'nin logaritması Eş 5.8'i toplamı verir.

$$f(x,y) = ay(x,y)\, yan(x,y) \tag{5.7}$$

$$\begin{aligned} z(x, y) &= \ln f(x,y) \\ &= \ln ay(x,y) + \ln yan(x,y) \\ &= ay'(x,y) + yan'(x,y) \end{aligned} \tag{5.8}$$

Olasılık teorisine göre, eğer yan' ve ay' bağımsız rastgele değişkenler ise $z(x, y)$ nin histogramı $ay'(x, y)$ ve $yan'(x, y)$ nın histogramlarının üst üste katlanmasıyla verilebilir [18].

Aydınlatma kaynağına erişim varolduğu zaman, uniforum olmayan aydınlatma sorununu düzeltmek için çok kullanılan bir çözüm; aydınlatma kaynağını sabit beyaz bir yansıtıcı yüzeye tutmaktır. Bunun sonucunda $g(x, y) = c_3\, ay(x, y)$ görüntüsü elde edilir ve c_3 sabiti yansıtma yüzeyine ve $ay(x, y)$ 'ye bağlıdır. Sonrasında herhangi bir görüntü için yukarıdaki denklem denk ile aynı aydınlatma fonksiyonuyla elde edilebilir. Basitçe $f(x, y)$' nin $g(x, y)$ ye bölünmesi normalize edilmiş $\eta(x, y)$ görüntüsünü elde etmeyi sağlar.

$$\eta(x, y) = \frac{f(x, y)}{g(x, y)} = \frac{yan(x, y)}{k} \tag{5.9}$$

Böylece eğer $yan(x, y)$ tekbir T eşik değeri ile bölümlendirilebilirse, Eş 5.9'da tekbir T eşik değeri kullanılarak bölümlendirilebilir [18].

5.3.3 Genel eşikleme

Görüntü içerisindeki nesne bölgelerini bölümlendirmenin kolay yolu tek bir eşik değeri T ile bölümlere ayırmaktır. Bu işlemin başarısı histogramın nekadar iyi bölümlenmiş olduğuna bağlıdır.

T eşik değerini otomatik olarak elde etmek için aşağıdaki algoritma kullanılabilir:

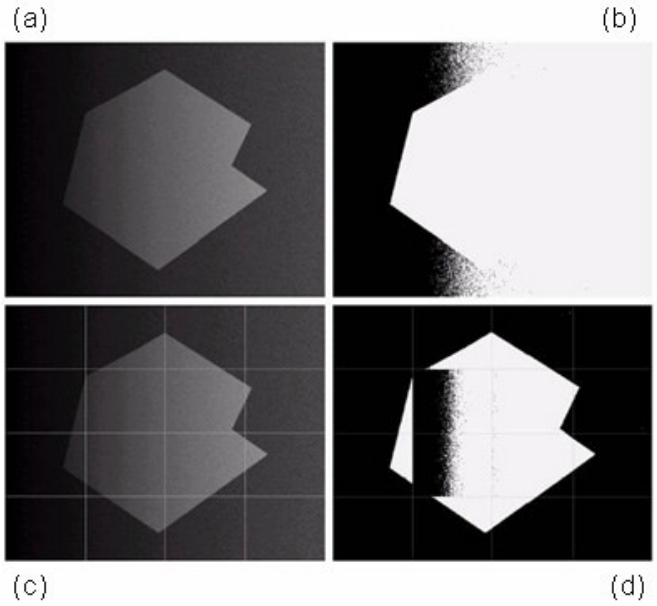
1) T için bir tahmini başlangıç değeri seçilir,
2) T eşik değeri ile resim bölümlendirilir. (Bunun sonucunda iki grup piksel ortaya çıkar. Bunlar, G_1; T < gri seviyesi değerine sahip tüm pikseller ve G_2; T ≥ gri seviye değerine sahip pikseller),
3) G_1 ve G_2 bölgelerindeki pikseller için sırasıyla μ_1 ve μ_2 ortalama gri seviye değerleri hesaplanır,
4) Yeni eşik değeri $T=1/2(\mu_{1+} \mu_2)$ ile elde edilir.
5) İterasyon sonucu elde edilen T değeri değişmeyene kadar iterasyon devam ettirilir.

Bu prosedürde başlangıç T değeri için resmin ortalama gri seviye değeri kullanılabilir. Eğer nesneler arkaplana nazaran daha az yer kaplıyorsa bu durumda T eşik değeri için maximum ve minimum değerleri arasında bir orta değeri seçmek daha doğru olur [18].

5.3.4 Uyarlanabilir eşikleme

Düzgün olmayan aydınlatmalı görüntüde tekbir T genel eşik değeriyle düzgün olarak bölünebilen bir histogramı etkili bir şekilde bölünemeyen bir hale dönüştürüyor. Bunun gibi durumlarda kullanılabilecek yaklaşım, orijinal resmi alt resimlere bölmek ve sonra her alt resmi bölümlendirmek için farklı bir eşik kullanmaktır. Bu yaklaşımdaki temel sorunlar resmin nasıl alt resimlere bölüneceği ve ortaya çıkan her bir alt resmin eşiğinin nasıl hesaplanacağıdır. Alt resimlere her piksel için pikselin konumuna göre eşik değeri uygulanması "uyarlanabilir eşikleme" olarak adlandırılır. Şekil.5.4'deki resim için tek eşik değeri verilirse, bu durumda sonuç Şekil 5.5 (b)'deki gibi görünür.

Ancak resmi Şekil 5.5 (c)'deki gibi alt bölümlere bölüp (16 alt parça) hepsi için eşikleme yapıldığında Şekil 5.5 (d)'deki sonuc elde edilir. Ortasındaki bölümüde tekrar alt resimlere bölerek nesne sınırlarını net olarak elde edilebilir.



Şekil 5.5. Uniform aydınlatılmamış resmin bölümlendirilmesi [18]

5.3.5 Optimum genel ve uyarlanabilir eşikleme

Bir resim iki temel gri seviye bölgesi içersin ve z gri seviye değeri ifade etsin. Bu değerler rastgele miktarlar olarak gösterilebilir ve resmin histogramından p(z) muhtemel yoğunluk fonksiyonu (PDF) kestirilebilir. Tüm yoğunluk fonksiyonu görüntüdeki siyah ve beyaz bölgelerin toplamı veya karışımıdır. Eğer yoğunlukların biçimi bilinir veya kestirilebilirse görüntüyü iki bölüme ayırmak için optimal bir eşik değeri belirlenebilir.

$p_1(z)$ ve $p_2(z)$ sırasıyla nesne ve geriplan için muhtemel yoğunluk fonksiyonları (PDF) olsunlar. Görüntüdeki toplam gri seviye değişimini gösteren toplam PDF Eş 5.10 dur.

$$p(z) = P_1 p_1(z) + P_2 p_2(z) \tag{5.10}$$

Burada P1+P2=1 olup sırasıyla P1 herhangi bir z' gri seviye değerli pikselin, nesneya ait olma olasılığını, P2 de geri plana ait olma olasılığını ifade eder. Resim T

eşiğinden daha büyük değerler arkaplan olacak şekilde bölümlenebilir. Diğer bütün pikseller nesne pikseli olarak atanır. Buradaki temel amac verilen bir pikselin, nesneye mi yoksa arkaplana mı ait olduğuna karar verirken yapılan ortalama hatayı en aza indirecek T değerini seçmektir.

Bir geriplan noktasının yanlışlıkla nesne noktası olarak sınıflandırılma olasılığı:

$$E_1(T) = \int_{-\infty}^{T} p_2(z)dz \tag{5.11}$$

dir. Aynı şekilde nesne noktasının yanlışlıkla geriplan noktası olarak sınıflandırılma olasılığı;

$$E_2(T) = \int_{-\infty}^{T} p_1(z)dz \tag{5.12}$$

dir. Bu durumda toplam hata olasılığı Eş 5.11 ve Eş 5.12 den :

$$E(T) = P_2 E_1(T) + P_1 E_2(T) \tag{4.13}$$

olur. Bu hatayı minimuma indirmek için Eş 5.13'ün T ye göre türevi alınıp 0'a eşitlenirse (leibniz kuralı) ;

$$P_1 p_1(T) = P_2 p_2(T) \tag{5.14}$$

elde edilir. Buradan T çözülebilir. T değerinin analitik olarak elde edilebilmesi için her iki PDF'in bilinmesi gereklidir. Fakat pratikte bu yoğunlukları hesaplamak mümkün değildir. Bu konuda kullanılan yoğunluklardan biri Gaussian yoğunluğudur ve iki parametre ile tanımlanır. Bunlar; ortalama (μ) ve değişim (σ)'dir. Bu durumda Gaussian muhtemel yoğunluk fonksiyonu; Eş. 5.15 dedir.

$$p(z) = \frac{P_1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(z-\mu_1)^2}{2\sigma_1^2}} + \frac{P_2}{\sqrt{2\pi}\sigma_2} e^{-\frac{(z-\mu_2)^2}{2\sigma_2^2}} \tag{5.15}$$

Eş 5.14 ve Eş 5.15 den;

$$\begin{aligned} &AT^2 + BT + C = 0 \\ &A = \sigma_1^2 - \sigma_2^2 \\ &B = 2(\mu_1\sigma_2^2 - \mu_2\sigma_1^2) \\ &C = \sigma_1^2\mu_1^2 - \sigma_2^2\mu_1^2 + 2\sigma_1^2\sigma_2^2 \ln(\sigma_2 P_1 / \sigma_1 P_2) \end{aligned} \tag{5.16}$$

elde edilir. Bu ikinci dereceden denklemin tekbir çözümü olması için $\sigma^2=\sigma_1^2=\sigma_2^2$ seçilmesi yeterlidir. Bu durumda T eşik değeri :

$$T = \frac{\mu_1 + \mu_2}{2} + \frac{\sigma^2}{\mu_1 - \mu_2} \ln(\frac{P_2}{P_1}) \tag{5.17}$$

dir. Eş 5.17'de $P_1=P_2$ olduğunda veya $\sigma=0$ olduğunda T eşik değeri, ortalama değer olarak elde edilir [18]. Optimal eşik değeri Rayleigh ve log-normal yoğunluk fonksiyonlarıyla da elde edilebilir.

5.4 Bölge Tabanlı Bölümlendirme

Bölümlendirme işleminde kullanılan diğer bir yaklaşım, Bölge Tabanlı bölümlendirmedir. Bu metotta gri seviye süreksizliklerinde yararlanılarak yapılan veya eşikleme ile yapılan bölümlendirmeden farklı olarak resim üzerinde direk olarak bölgelerin bulunması mantığı yatar. Bu bölümlendirme tipine örnek olarak Bölge Büyütme, Bölge bölme ve birleştirme, watershed metotları verilebilir. Oluşturulan yazılımda bölge tabanlı bölümlendirme metotları kullanılmamıştır.

5.4.1 Bölge tabanlı bölümlendirmenin temeli

Im bir resmin tümünü ifade etsin ve Im_i, i=1,2,3,.....,n resim içindeki altbölgeleri ifade etsin. Bu alt bölgeler şu şartları sağlamalıdır [18].

a. $\bigcup_{i=1}^{n} \mathrm{Im}_i = I$ olmalıdır. Yani her piksel mutlaka bir bölge içinde olmalıdır.

b. Ri içindeki pikseller komşuluk şartlarını sağlamalıdır.

c. $\mathrm{Im}_i \bigcap \mathrm{Im}_j = \varphi$ tüm i ve j i ≠ j için iki Im_i ve Im_j nin kesişimi olmamalıdır.

d. Prop(Im_i) = true i=1,2,3,.....,n için Im_i bölgesindeki tüm pikseller aynı özelliği göstermelidir (örneğin aynı gri seviye değeri gibi).

e. Prop(Im_i U Im_j) = false Im_i ve Im_j bitişik bölgeleri belirtilen bir özelliğe göre farklı olmalıdır.

Burada Prop herhangi bir Im_k seti içindeki piksellerle ilgili özelliği göstermektedir.

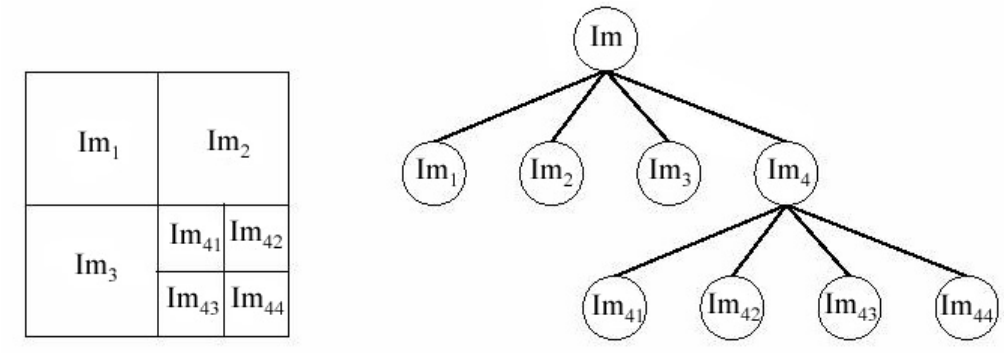
5.4.2 Bölge büyütme metodu

Adından da anlaşılacağı gibi piksellerin belirli kriterler temel alınarak, altbölgeleri daha büyük bölgeler haline getirme işlemidir.r. Bu işlem önce "tohum " adı verilen bir nokta ile başlar ve bu noktanın komşuları belirli bir kritere göre (örneğin griseviyesi veya rengi) bir bölgeye dahil edilir. Bu metotta temel problem başlangıç noktası seçmektir. Bunun için öncelikle her pikselin özelliklerine bakılarak piksellerin benzer dizilere atanması yapılır. Bu dizilerin değerleri kümeleri oluşturur ve bu kümelerin merkezlerindeki pikseller başlangıç "tohum" pikseli olarak kullanılabilir.

Bölge büyütme işleminde, komşuluk ve bitişiklik bilgileri kullanılmadan belirleyici özellikler tek başına kullanıldığında hatalı sonuçlar verir. Bölge büyütme işlemi pikseller artık gri seviye değeri, kaplama özelliği, rengi gibi özellikleri istenen kriterleri sağlamadığı zaman durdurulur. Bölge büyütme metodu, histogramında ikiden fazla mod olan görüntülerin bölümlendirmesinde daha iyi sonuçlar verir.

5.4.3 Bölge bölme ve birleştirme metodu

Bu metotta görüntü içindeki bir çekirdek nokta ile başlanır. Bölge büyütme metodundan farkı resmi başlangıçta, rasgele bağlantısız bölgelere bölerek ve sonra girişte verilen kriterlere göre birleştirmek ve veya ayırmaktır. Bu işlem iterasyon sonucu değişmeyene kadar uygulanır. Bu tekniği ifade etmek için 'quadtree' ifadesi kullanılır.



Şekil 5.6. Parçalanmış resim ve "quadtree" yapısı

Şekil 5.6'da görüldüğü gibi resim önce dallara sonra alt dallara bölünür ve alt dallar belirtilen kriterlere göre birleştirilir. Bu işlem iterayonla daha fazla bölme veya birleştirme gerçekleştirilemeyene kadar devam ettirilir.

5.5 Watershed Bölümlendirme

Watershed bölümlendirme içerik olarak kenar belirleme, eşikleme gibi bölümlendirme metotlarını kapsamaktadır. Fakat onlardan farklı olarak daha kararlı ve sürekli sınırlardan oluşan bir bölümlendirme sağlar.

Watershed bölümlendirmede temel mantık görüntünün üç boyutlu halde kullanılmasına dayanır. Bu gösterim için resmin iki boyutlu koordinatları ve gri seviye değerleri kullanılır. Görüntünün üç boyutlu topografik gösterimi ele alınırken üç tip nokta önemlidir. Bunlar;

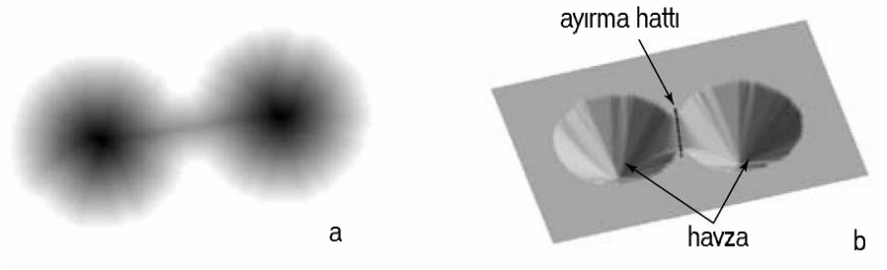
a. Yerel minimuma ait noktalar,
b. Su damlası düştüğünde suyu kesinlikle tutabilecek tek minimum noktası,
c. Su damlası düştüğünde tek bir minimumdan fazla yere eşit olarak dağılmasına yol açacak noktalar.

Bu koşullar altında b' şıkkında tanımlanan yapıya uyan noktalara havza (watershed veya yakalama kabı) , c' şıkkında tanımlanan yapıya uyan noktalara ise ayırma hattı veya sırt hattı (watershed lines, divide lines) adı verilir.

Watershed bölümlendirmede hedef, resim içerisinde ayırma hattı (watershed lines) adı verilen yerlerin bulunmasıdır. Örnekleme için bir görüntüyü üç boyutlu gösterimdeki her bir minimumunda bir delik olduğu düşünülürse ve bu deliklerden suların yükseldiği kabul edilirse ayrı havzalardaki suların birleşmesini önlemek için her rezervuar arasında bir duvar olması gerektiği açıktır ve bu duvaların üstü sular tamamen yükseldiğinde sürekli bir sınır oluştururlar. Bu sınır çizgileri kullanılarak kopukluk ve kararsızlık olmadan bölümlendirme yapılır.

Şekil 5.7'de bu durum daha açık bir şekilde görülmektedir. Şekil 5.7 a, iki boyutlu gri seviye görüntü, b ise bu görüntünün üç boyutlu gösterimidir. Burada iki havza arasına bir damla düştüğünde damlanın iki havzadan birinde toplanması aynı derecede muhtemeldir. Her iki havza arasındaki çizgi watershed hattıdır ve bölümlendirmede sürekli bir sınır elde etmeyi sağlar. Buradaki iki havza birbirine çok yakın olan ve görüntü içinde belirlemek istenilen nesneler olarak düşünülebilir. Watershed bölümlendirme genelde "distance dönüşümü" sonrası uygulanır.Bu dönüşümde ikili düzey resim kullanılır ve her bir pikselin kendine en yakın sıfırdan farklı piksele olan mesafesinin ölçülmesiyle elde edilir.

Watershed bölümlendirmede en önemli sorun görüntü direk ele alındığında aşırı bölümlendirme (oversegmentation) sonucu vermesidir. Bu problemi aşmak için görüntü içinde kabul edilebilir bölgeler, bölümlendirme prosedürüne, ön işlemlerle ek bilgi katarak sınırlandırmalar getirilir. Aşırı bölümlendirmeyi kontrol etmede temel fikir markerları kullanmaktır. Marker bir görüntüye ait bağlı bileşenlerdir.



Şekil 5.7. a) gri seviye görüntü, b) görüntünün üç boyutlu gösterimi [19]

Görüntü içerisinde her bölgede belirlenmek istenen nesnenin olduğu yerler dahili marker olarak ve o bölgedeki geriplan da harici marker olarak tanımlandıktan sonra watershed bölümlendirmesi uygulamak aşırı bölümlendirme problemini çözmede kullanılan bir metottur.

5.6 Özellik Çıkarma

5.6.1 Alan

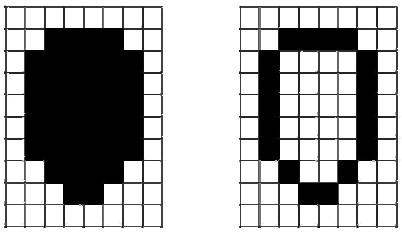
Bir nesnenin alanı o nesneyi oluşturan piksel sayısının toplamıyla bulunabilir. B(x,y) n satır m sütuna sahip etiketlenmiş bir resmin x'inci sütununu ve y'inci satırdaki değerini ifade etsin. k'ıncı nesnenin alanı Eş 5.18 ile bulunabilir [11].

$$A_k = \sum_{x=1}^{n} \sum_{y=1}^{m} B(x,y) \tag{5.18}$$

Burada $B(x,y) = \begin{cases} 1 & \text{Eger } B(x,y) \in k \text{ ise} \\ 0 & \text{Eger } B(x,y) \notin k \text{ ise} \end{cases}$ dir.

5.6.2 Çevre

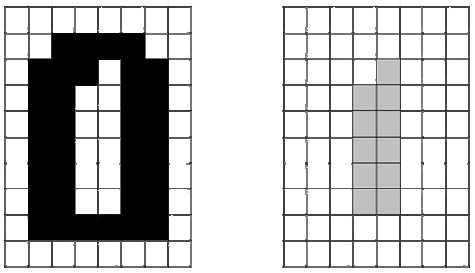
Çevre nesnenin kenar piksellerinin sayısıdır. Şekil 5.8'de bir nesnenin ikili düzey resmi ve kenar pikselleri görülmektedir.



Şekil 5.8. Nesne çevresi

5.6.3 Delik sayısı

Nesnenin içerdiği delik sayısını ifade eder. Nesne içerisindeki geriplan piksel değerine sahip olup, geriplanla bağlantısı olmayan bölegeler delik olarak kabul edilir. Şekil 5.9'da bir nesnenin ikili düzey resmi ve nesnedeki delik görülmektedir.



Şekil 5.9. Nesnedeki delik

5.6.4 Delik alanı

Nesne içerisindeki deliğin piksel miktarıdır.Yukarıdaki şekil için piksel alanı 11 pikseldir.

5.6.5 Yuvarlaklık

Bir nesnenin daireselliğini ifade eder. Eş 5.19 ile elde edilir [11].

$$C = \frac{P_{ç}^{2}}{A_{k}} \tag{5.19}$$

Burada C yuvarlaklık $P_ç$ çevre A_k ise alandır.

5.6.6 Yarıçap

Yarıçap nesnenin merkezinden sınırına kadar olan piksel miktarıdır. Yarıçap için merkezden tüm sınır piksellerine olan miktar bulunarak ortalaması alınır.

5.6.7 Kütle merkezi (centroid)

Centroid nesnenin kütle merkezinin yerini ifade eder. Yani kütle merkezi pikselinin koordinatları kütle merkezinin yeridir. Kütle merkezinin koordinatları Eş 5.20 ile elde edilir [11].

$$\overline{X} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} iB(i,j)}{A_k} \qquad \overline{Y} = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} jB(i,j)}{A_k} \tag{5.20}$$

5.6.8 Doğrultu

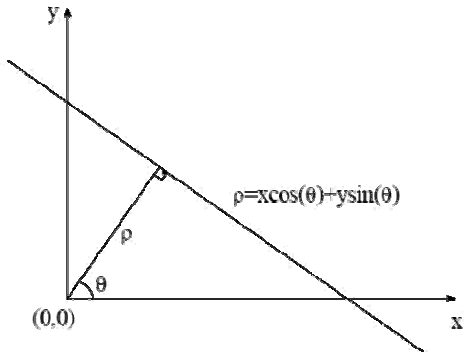
Bir nesnenin doğrultusunu ifade etmek için eksen uzantılarının doğrultuları kullanılabilir. Genelde ikinci momenti en düşük olan eksen, eksen uzantısı olarak kullanılır. Bir nesnenin ikinci moment ekseni eksen çizgisiyle nesne noktaları arasındaki mesafenin karelerinin toplamı minimum olan çizgidir. Bu çizgi Eş 5.21'in minimize edilmesiyle elde edilebilir [11].

$$X^2 = \sum_{i=1}^{n}\sum_{j=1}^{m} r_{ij}^2 B(i,j) \tag{5.21}$$

Bu fonksiyon nesne noktaları ile eksen çizgisi arasındaki dikey mesafenin kareleri toplamını verir. Burada r_{ij} (i,j) koordinatlarındaki nesne noktasıyla eksen çizgisi arasındaki dikey mesafedir.

Doğrultu ekseni çizgisi polar koordinatlarla ifade edilirse Şekil 5.10'da görüldüğü gibi ,Eş.5.22 şeklinde yazılabilir.

$$\rho = x.Cos(\theta) + y.Sin(\theta) \tag{5.22}$$



Şekil 5.10. Bir doğrunun polar koordinatlarla ifadesi

r mesafesi (x,y) noktası için Eş 5.23'den elde edilir.

$$r^2 = (x.\mathrm{Cos}(\theta) + y.\mathrm{Sin}(\theta) - \rho)^2 \tag{5.23}$$

Eş 5.23'ü Eş 5.21'de yerine koyarsak

$$X^2 = \sum_{i=1}^{n} \sum_{j=1}^{m} (x_{ij}\mathrm{Cos}(\theta) + y_{ij}\mathrm{Sin}(\theta) - \rho)^2 B(i, j) \tag{5.24}$$

elde edilir. Bu ifadenin ρ'ya göre türevi alınıp "0" eşitlenirse

$$\rho = \bar{x}.\mathrm{Cos}(\theta) + \bar{y}.\mathrm{Sin}(\theta) \tag{5.25}$$

elde edilir. ρ'yu Eş 5.25'de yerine koyarsak ve

$$x' = x - \bar{x} \tag{5.26}$$

$$y' = y - \bar{y} \tag{5.27}$$

şeklinde yazarsak, minimizasyon problemi;

$$X^2 = a.\mathrm{Cos}^2(\theta) + b.\mathrm{Sin}(\theta).\mathrm{Cos}(\theta) + c.\mathrm{Sin}^2(\theta) \tag{5.28}$$

olur. Burada

$$a = \sum_{i=1}^{n}\sum_{j=1}^{m}(x'_{ij})B(i,j) \tag{5.29}$$

$$b = 2\sum_{i=1}^{n}\sum_{j=1}^{m}(x'_{ij}\,y'_{ij})B(i,j) \tag{5.30}$$

$$c = \sum_{i=1}^{n}\sum_{j=1}^{m}(y'_{ij})^2 B(i,j) \tag{5.31}$$

dir. Eş 5.28 'den

$$X^2 = \frac{1}{2}(a+c) + \frac{1}{2}(a-c)\mathrm{Cos}(2\theta) + \frac{1}{2}b.\mathrm{Sin}(2\theta) \tag{5.32}$$

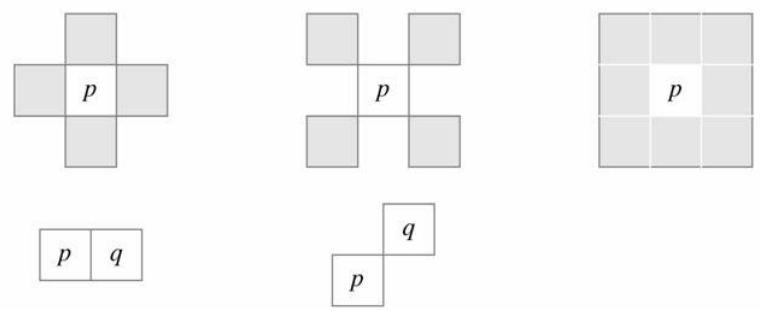
şeklinde yazılabilir. Eş 5.32 θ'ya göre türevi alınıp "0" a eşitlenirse Eş 4.33 elde edilir [11].

$$\mathrm{Tan}(2\theta) = \frac{b}{a-c} \tag{5.33}$$

5.6.9 Birbirleriyle ilişkili piksel gruplarının etiketlenmesi (labeling)

Etiketleme, iki resim içerisindeki her bir pikselin belli bir komşuluk ilişkisi göz önüne alınarak gruplanması işlenilir. Etiketleme işlemi sonucu her bir ayrı grup ayrı bir nesneyi temsil eder. Birden fazla resim içeren nesnelerde nesne özelliklerini ayırt edebilmek için etiketleme gerekli bir aşamadır. Piksel gruplama işlemi pikselin komşularının kontrolüyle yapılır.

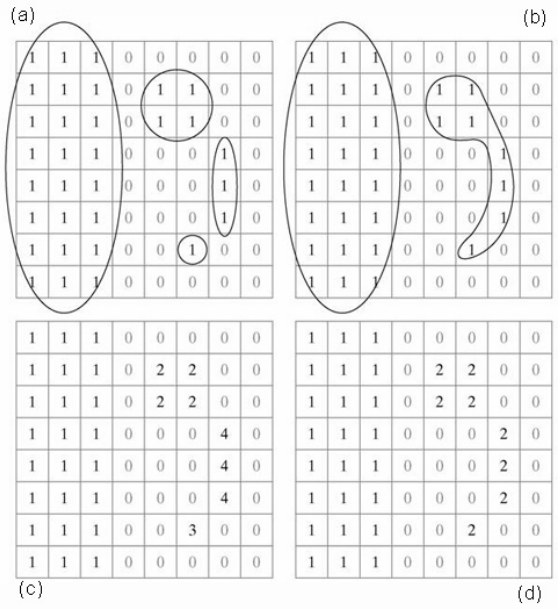
Şekil 5.11'deki gibi bir P(x,y) noktası için dörtlü komşuluk seti (x+1,y) (x-1,y) (x,y+1) (x,y-1) şeklindedir. $N_4(P)$ ile gösterilir. Diyagonal komşular ise (x+1,y+1) (x+1,y-1) (x-1,y+1) (x-1,y-1) dir. Tüm bu pikseller 8li komşuluk ilişkisini verir.



Şekil 5.11. Piksel komşuluk setleri

İki p ve q noktaları için verilen komşuluk ilişkisine göre bir yol varsa bu durumda p ve q noktaları 'connected component' bağlı piksellerdir ve aynı piksel grubunun içindedirler. Yani parçaların bağlı olup olmadığını seçilen komşuluk ilişkisi belirler.

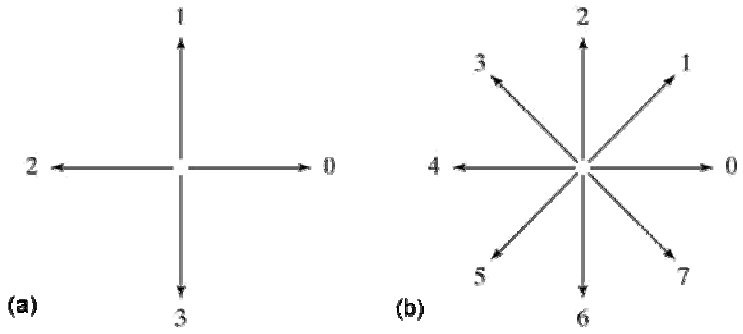
Şekil 5.12 (a)'da bir resim bölümünün 4'lü ve (b)'de 8'li komşuluk ilişkisine göre bağlı bölgelerinin belirlenmesi görülmektedir. (c)'de 4'lü komşuluğa göre belirlenen bölgenin ve (d)'de ise 8'li komşuluğa göre bu bölümlerin etiketlenmesi görülmektedir. Etiketlemede geri plan pikselleri sıfır ile ifade edilir. Sonra her bir piksel grubu 1'den başlayarak etiketlenir.



Şekil 5.12. Etiketleme (Labelling)

5.6.10 Zincir kodlama

Bilgisayar ortamında nesne ne kadar iyi kodlanırsa üzerinde yapılacak işlemler o kadar iyi olur. İkili düzey görüntüden nesnenin sınır analizi en iyi zincir kodlamayla yapılabilir. Zincir kodlama sınırlarının, belirlenmiş uzunluk ve doğrultuda düz çizgi parçalarıyla ifade edilmesinde kullanılır. Bu işlemde yine 4 veya 8'li komşuluk ilişkisi esas alınır.



Şekil 5.13. Freeman zincir kodları

Her bir çizik parçasının yönünün belirlemek için Şekil 5.13'de görüldüğü gibi (a)'da 4'lü, (b)'de 8'li komşuluk tipine göre düzenlenmiş doğrultu numaraları kullanılır. Bu ifade planı 'freeman chain codes' olarak bilinir.

Bu düzenlemeyle elde edilen ikili görüntü üzerinde nesnenin alanı çevresi genişliği gibi özelikler elde edilebilir.

5.7 Nesne Tanıma (Sınıflandırma)

Nesne tanıma görüntü işlemenin en son aşaması olarak ifade edilebilir. Daha önceki işlemlerde nesne olma ihtimali olan ayrı piksel grupları elde edilmiş ve bunlar etiketlenmişti. Bu aday bölgelere nesne veya kalıp denir.

Kalıp tanıma metotları iki ana gruba ayrılabilir. Bunlar; Karar-Teorik tanıma ve Yapısal tanımadır. Karar-Teorik tanıma alan, uzunluk, eksen gibi niceliksel tanımlayıcılarla tanımlanmış kalıpları ele alır. Yapısal tanıma ise yapısal ilişkiyi

veren (Ör; bir yerleşim yeri fotoğrafında özel binalar gibi) niteliksel tanımlayıcılarla tanımlanan kalıplarla ilgilenir.

5.7.1 Kalıplar ve kalıp sınıfları

Bir kalıp alan, çevre, ağırlık merkezi, doğrultu vb. özelliklerden oluşan tanımlayıcıların düzenini ifade eder. Kalıp sınıfı ortak özellikleri paylaşan bir kalıp ailesidir.

Uygulamada yaygın olarak kullanılan 2 kalıp düzenlemesi; vektörler (niceliksel tanımlamalar için) diziler ve ağaçlardır (yapısal tanımlamalar için).

Kalıp vektörleri $x = [x_1, x_2, \ldots\ldots x_n]^T$ şeklindedir [18]. Kalıp vektörleri (n x 1 matrislerdir). Kalıp vektör yönteminin sonuç performansını etkileyen en önemli faktör, vektör bileşenlerinin üzerine kurulu olduğu tanımlayıcıların seçimidir. Kalıp vektörü seçiminde alan, çevre gibi tanımlayıcıların yanı sıra bazı uygulamalarda yapısal ilişkilerde kullanılır. Örnek olarak parmak izi tanımada "minutiae" denilen çizgi düğüm, bitim noktaları verilebilir.

Birçok uygulama için daha güçlü bir yöntem ise ağaç tanımlamalarıdır. Bu yöntemde ise resim kök olarak kabul edilip içerisindeki bölgeler ve bölgeler içindeki alt bölgeler dallanarak birbirleriyle ilişkilendirilirler.

Kalıp vektörlerindeki tanımlayıcı bileşenler (alan, çevre, oryantasyon vb.) nümerik olabileceği gibi sembolikte olabilir. Eğer her tanımlayıcı bir sayı ise tanımlama uzayı R^n şeklindedir. Bir nesneyi tanımlama işlemi, tanımlama uzayının parçalara bölünmesini yapar. Bu parçalar arasındaki sınıra ise "karar sınırı" adı verilir. Tanıma işlemi tanımlama uzayı parçalara bölünüp karar sınırları belirlendikten sonra kolaylaşır. Kalıp vektörü hangi bölgeye düşüyorsa, kalıp vektörü ilgili tanımlanmış sınıfa dahil edilir.

5.7.2 Karar teorik tabanlı sınıflandırma metotları

Karar teorik sınıflandırma yaklaşımları, karar (veya Discriminant) fonksiyonunu temel alır. n boyutlu kalıp vektörü Eş 5.34 olsun.

$$x = [x_1, x_2, \ldots\ldots x_n]^T \tag{5.34}$$

$\omega_1, \omega_2, \ldots\ldots, \omega_w$ şeklindeki W tane kalıp sınıfları için karar teorik kalıp tanımadaki temel problem, $d_1(x), d_2(x), \ldots\ldots d_w(x)$ karar fonksiyonlarını, eğer bir *x* kalıbı, ω_i sınıfına ait ise Eş 5.35'i sağlayacak şekilde bulmaktır.

$$d_i(x) > d_j(x) \quad j=1,2,\ldots W;\ j \neq i \tag{5.35}$$

Diğer bir deyişle eğer $d_i(x)$, *x* bütün karar fonksiyonlarında yerine konduktan sonra, en büyük sayısal değerini verirse, kalıp *x*'in ω_i'e ait olduğu söylenir.

ω_i ve ω_j'yi ayıran karar sınırı *x*'in $d_i(x) - d_j(x) = 0$ değerleri ile verilir. İki sınıf arasındaki karar sınırını belirlemek için yaygın olarak tek bir fonksiyon

$$d_{ij}(x) = d_i(x) - d_j(x) = 0 \tag{5.36}$$

kullanılır [18]. Böylece ω_i sınıfının kalıpları için $d_{ij}(x)>0$ ve ω_j sınıfının kalıpları için $d_{ij}(x)<0$ olur.

Karşılaştırmalı sınıflandırma

Karşılaştırmaya dayalı sınıflandırma teknikleri her sınıfı bir prototip kalıp vektör ile gösterir. Bilinmeyen bir kalıp önceden tanımlanmış bir ölçüye göre en yakın olan sınıfa atanır. En kolay yaklaşım bilinmeyen kalıp vektörüyle her bir prototip vektörü karşılaştırıp arasındaki mesafeyi hesaplamaktır. Bir karar vermek için en küçük mesafe seçilir.

Minimum mesafe sınıflandırıcılar

Her kalıp sınıfının prototipi, o sınıfın kalıplarının ortalama vektörü olsun.

$$m_j = \frac{1}{N_j} \sum_{x \in \varpi_j} x_j \qquad j=1,2,\ldots\ldots,W \tag{5.37}$$

Burada N_j, ω_j sınıfındaki kalıp vektörlerinin sayısıdır. Bilinmeyen bir kalıp vektörü x'in sınıf üyeliğini belirlerken izlenecek yol; prototip vektörlerle farkına bakıp en

yakın prototip vektörünün sınıfına atamaktır. Yakınlığı belirlemek için Euclidean mesafesini kullanarak, mesafeyi hesaplamadaki problemi azaltır. Euclidean mesafe

$$D_j(x) = \left\| x - m_j \right\| \qquad j = 1,2,.....,W \tag{5.38}$$

dir. Eş 5.38'e Euclidean kuralını ($D_j(x) = \left(\left[x - m_j \right]^T . \left[x - m_j \right] \right)^{1/2}$) uygulanırsa,

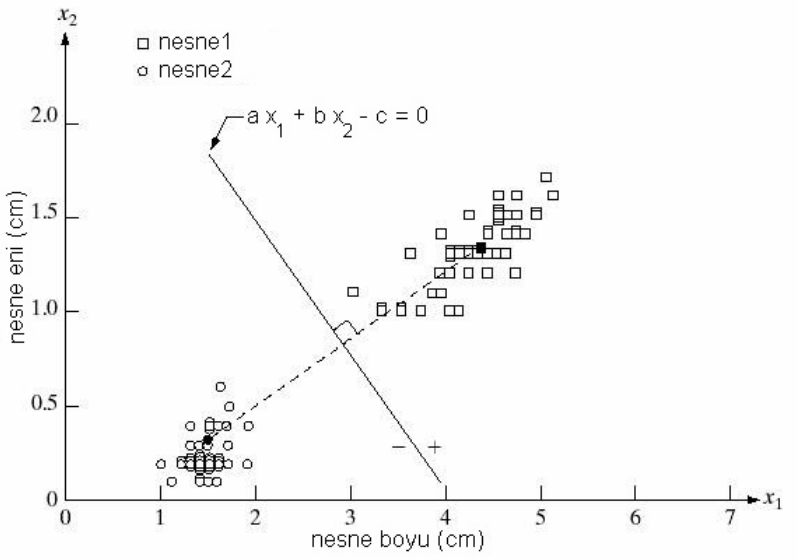
$$d_j = x^T m_j - \frac{1}{2} m_j^T m_j \tag{5.39}$$

elde edilir. Bu durumda ω_i ve ω_j arasındaki karar sınır fonksiyonu

$$\begin{aligned} d_{ij}(x) &= d_i(x) - d_j(x) \\ &= x^T (m_i - m_j) - \frac{1}{2} (m_i - m_j)^T (m_i + m_j) = 0 \end{aligned} \tag{5.40}$$

şeklinde elde edilir.

Şekil 5.14'de minimum mesafe sınıflandırmaya ait bir örnek görülmektedir.



Şekil 5.14. Minimum mesafe sınıflandırma [18]

Burada ω_1 ve ω_2 kalıp sınıfıyla iki farklı nesne ifade edilmiştir. Kalıp vektörleri ise çiçeklerin genişlik ve uzunluk tanımlayıcılarını içermektedir. m_1 ve m_2 ise prototip kalıp vektörleri olup sırasıyla ω_1 ve ω_2 sınıflarındaki kalıp vektörlerinin ortalamalarıdır. Karar fonksiyonlarını Eş 5.39 ile bulunduktan sonra d_{12}, Eş 5.40 ile elde edilir [18].

Herhangi bir x kalıp vektörü için $d_{12}(x) > 0 \Rightarrow x \ \omega_1$ sınıfına, $d_{12}(x) < 0 \Rightarrow x \ \omega_2$ sınıfına ait olur.

Optimum istatistiksel sınıflandırıcılar

Optimum istatistiksel sınıflandırmada, tanıma için istatistiksel yaklaşımlar kullanılır.

Optimum istatistiksel sınıflandırıcıların temeli:

Belirli bir kalıp vektörü x'in ω_i kalıp sınıfından olma olasılığı $p(\omega_i/x)$ ile gösterilir. Eğer kalıp sınıflandırıcısı x'in ω_j'den geldiğine karar verirse, gerçekte ω_i'den geldiği halde L_{ij} ile gösterilen bir kayıba uğrar. Değerlendirme yapılırken x kalıbı, W tane kalıp sınıfından herhangi birine ait olabileceği için x'i ω_j sınıfına atarken uğranan ortalama kayıp;

$$r_j(x) = \sum_{k=1}^{W} L_{kj}\, p(\varpi_k / x) \tag{5.41}$$

dir. Bu denklem karar teorisi terminolojisinde genellikle "koşullu ortalama risk" veya "kayıp" olarak adlandırılır. Temel olasılık teorisinden Eş 5.41'i açarsak;

$$r_j(x) = \frac{1}{p(x)} \sum_{k=1}^{W} L_{kj}\, p(x / \varpi_k).P(\varpi_k) \tag{5.42}$$

olur [18]. Burada $p(x/\omega_k)$, ω_k sınıfından olan kalıpların olasılık yoğunluk fonksiyonu (Probability Density Function) dur. $P(\omega_k)$ ise ω_k sınıfının oluşma olasılığıdır. $1/p(x)$, bütün $r_j(x)$, j=1,2,.....,W'lere pozitif ve ortak olduğu için ve en küçük değerden en büyük değere kadar bu fonksiyonların bağlantılı düzenini etkilemediğinden Eş 5.42'den çıkarılırsa;

$$r_j(x) = \sum_{k=1}^{W} L_{kj}\, p(x/\varpi_k).P(\varpi_k) \tag{5.43}$$

olur. Sınıflandırıcı, verilen herhangi bir bilinmeyen kalıp için içinden sınıf seçebileceği W olası sınıfa sahiptir. Eğer sınıflandırıcı, her kalıp x için $r_1(x)$, $r_2(x)$,.....$r_w(x)$'i hesaplarsa ve kalıbı en az kayıplı sınıfa atarsa, bütün kararlarla ilgili toplam ortalama kayıp minimum olacaktır. Toplam ortalama kaybı minimize eden sınıflandırıcı, "Bayes Sınıflandırıcısı" olarak adlandırılır.

$$\sum_{k=1}^{W} L_{kj}\, p(x/\varpi_k).P(w_k) < \sum_{q=1}^{W} L_{qj}\, p(x/\varpi_q)P(\varpi_q) \tag{5.44}$$

Bayes sınıflandırıcı her $j; j \neq i$ için eğer Eş 5.44 sağlanıyorsa x_1'i ω_i sınıfına atar. Doğru bir karar için kayıba genellikle sıfır değeri atanır ve her yanlış karar için kayıba genellikle aynı sıfır olmayan değer (örneğin 1) atanır. Bu durumda kayıp fonksiyonu Eş 5.45'deki gibi olur.

$$L_{ij} = 1 - \delta_{ij} \tag{5.45}$$

Eğer $i = j$ ise $\delta_{ij} = 1$ dir, $i \neq j$ ise $\delta = 0$ 'dır. Eş 5.45, Eş 5.34'de yerine konulursa;

$$\begin{aligned} r_j(x) &= \sum_{k=1}^{W} (1 - \delta_{kj})\, p(x/\varpi_k)P(\varpi_k) \\ &= p(x) - p(x/\varpi_j)P(\varpi_j) \end{aligned} \tag{5.44}$$

olur. Bayes sınıflandırıcı sonra her $j \neq i$ için

$$\begin{aligned} &p(x) - p(x/\varpi_i)P(\varpi_i) < p(x) - p(x/\varpi_j).P(\varpi_j) \\ &j = 1,2,.....,W; \quad j \neq i \end{aligned} \tag{5.45}$$

ise o kalıp x'i, ω_i sınıfına atar. Optimumluğu sağlamak için yanlış sınıflandırmadaki ortalama kayıp minimize edilmelidir. Bununla birlikte optimaliteyi tutturmak için her sınıfın oluşma olasılığının yanı sıra her sınıftaki kalıpların olasılık yoğunluk fonksiyonlarının da biliniyor olması gerekir. Kalıp sınıflarının oluşma olasılığı

genellikle problemden çıkarılabilir veya kolayca elde edilebilir. Fakat olasılık yoğunluk fonksiyonunun tahmini bir sorundur. Eğer kalıp *x*, n boyutlu ise $p(x/\omega_i)$, n değişkenlerin bir fonksiyonudur. Bu fonksiyonun şekli bilinmiyorsa tahmin edilebilmesi için en yaygın varsayılan form Gaussian PDF'dir [18].

Gaussian kalıp sınıfları için bayes sınıflandırıcı

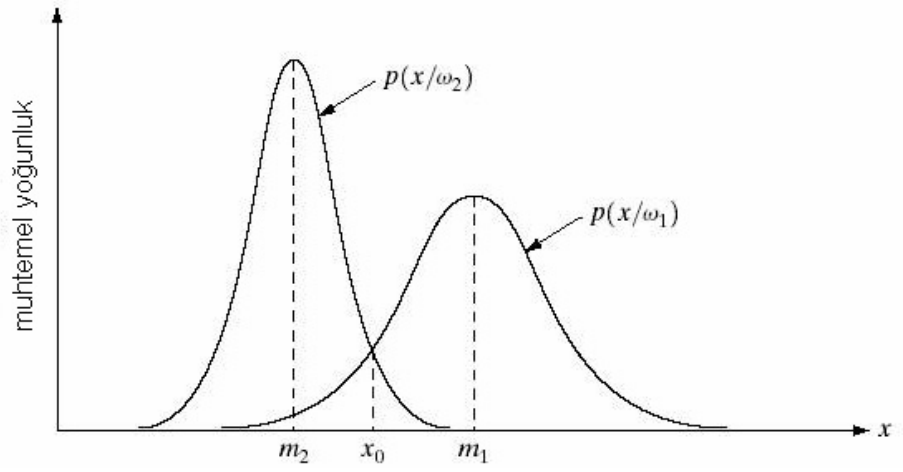
Gaussian yoğunluğuna sahip ortalamaları m_1, m_2 standart sapmalar σ_1, σ_2 olan iki kalıp sınıfı ω_1, ω_2 olsun (W=2). Bir boyutlu Bayes karar fonksiyonunu gaussian yoğunluğuna sahip kalıp sınıfları;

$$d_j(x) = p(x/\varpi_j)P(\varpi_j)$$

$$= \frac{1}{\sqrt{2\pi}\sigma_j} e^{-\frac{(x-m_j)^2}{2\sigma_j^2}} P(\varpi_j) \qquad (5.46)$$

$$j = 1,2$$

olur.



Şekil 5.15. İki kalıp sınıfı için muhtemel yoğunluk grafiği

Şekil 5.15'da bu iki sınıf için PDF'in grafiği görülmektedir. İki sınıf arasındaki sınır ek bir noktadır (x_0). Eğer iki sınıfın oluşma olasılığı eşitse $P(\varpi_1) = P(\varpi_2) = \frac{1}{2}$ dir ve karar sınırı $p(x_0/\omega_1)= p(x_0/\omega_2)$ için x_0 değeridir. Şekilde de görüldüğü gibi x_0'ın sağındaki her kalıp ω_1 sınıfına, solundakilerde ω_2 sınıfına atanır.

n boyutlu durumda, j'nci kalıp sınıfındaki vektörlerin Gaussian yoğunluğu;

$$p(x/\varpi_j) = \frac{1}{(2\pi)^{n/2}\left|C_j\right|^{1/2}} e^{-\frac{1}{2}(x-m_j)^T C_j^{-1}(x-m_j)} \tag{5.47}$$

şeklindedir. Burada her yoğunluk kendi ortalama vektörü m_j ve kovaryans matrisi C_j ile belirtilir; $\left|C_j\right|$ kovaryans matrisinin determinantı, n kalıp vektörlerinin boyutudur. Ortalama vektör ve kovaryans matris m_j ve C_j;

$$m_j = \frac{1}{N_j} \sum_{x \in \varpi_j} x \tag{5.48}$$

$$C_j = \frac{1}{N_j} xx^T - m_j m_j^T \tag{5.49}$$

şeklinde tahmin edilebilirler. Burada N_j, ω_j kalıp sınıfındaki vektörlerin sayısıdır ve toplam bu vektörler üzerinden alınmıştır. Konvaryans matrisi simetrik ve pozitif semidefinittir.

Bayes karar fonksiyonunu, gaussian yoğunluğunun üssel formundan dolayı doğal logaritmasıyla çalışmak daha kullanışlıdır. Tekrar düzenlenirse:

$$d_j(x) = \ln\left[\, p(x/\varpi_j)P(\varpi_j)\,\right] \\ = \ln\, p(x/\varpi_j) + \ln P(\varpi_j) \tag{5.50}$$

olur. Eş 5.50, Eş 5.47'de yerine koyulursa;

$$d_j(x) = \ln P(\varpi_j) - \frac{n}{2}\ln 2\pi - \frac{1}{2}\ln|C_j| - \frac{1}{2}\left[(x-m_j)^T\, C_j^{-1}\,(x-m_j)\right] \tag{5.51}$$

elde edilir. $\frac{n}{2}\ln 2\pi$' i ihmal edersek (Çünkü tüm sınıflar için aynı) Eş 5.51;

$$d_j(x) = \ln P(\varpi_j) - \frac{1}{2}\ln|C_j| - \frac{1}{2}\left[(x-m_j)^T\, C_j^{-1}\,(x-m_j)\right] \tag{5.52}$$

olur[18]. Bu ifade 0-1 kayıp fonksiyonu koşulu altında Gaussian kalıp sınıfları için Bayes Karar Fonksiyonudur.

Eğer C_j=C (j=1,2, ..., W) ve kalıp sınıflarının oluşma olasılığı $p(\varpi_j) = \frac{1}{W}$ ise ve C=I (birim matris) ise ayrıca kalıp sınıfları Gaussian ise; Gaussian kalıp sınıfları için Bayes Karar Fonksiyonu;

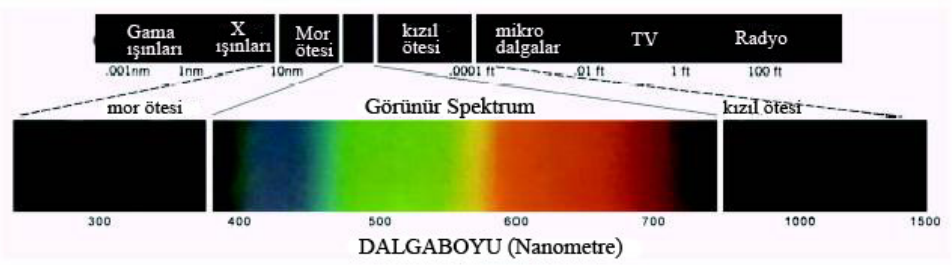
$$d_j(x) = x^T\, m_j - \frac{1}{2}\, m_j^T\, m_j \qquad j=1,2,\ldots..,W \tag{5.53}$$

şekline dönüşür. Eş 5.53, Eş 5.39'a eşittir [18]. Bu da minimum mesafe sınıflandırıcısının, Bayes mantığında optimum olduğunu gösterir.

6. RENKLİ RESİM İŞLEME

6.1 Renk Temelleri

Bölüm 2 de bir ışık huzmesinin bir cam prizmadan geçince çıkan ışığın, mordan kırmızıya kadar uzanan renklerden oluşan sürekli bir spektrum içerdiği belirtilmiti. Renk spektrum'u altı farklı bölgeye bölünebilir: mor, mavi, yeşil, sarı, turuncu ve kırmızı. Şekil 6.1'de de görüldüğü gibi hiçbir rengin ani bir bitişi yoktur, her rengin bir sonrakine yumuşak bir karışım şeklinde geçişi vardır.



Şekil 6.1. Görünür ışığı da içeren Elektromanyetik spektrum

Temel olarak insanların ve diğer bazı hayvanların bir nesneden algıladığı renkler, ışığın nesneden yansıması özelliğidir. Görülebilen ışık, elektromanyetik spektrumda nispeten dar frekans bantlarından oluşur. Her görüşebilir dalga boyunda dengelenmiş ışığı yansıtan bir kütle, gözlemciye beyaz görünür. Bununla birlikte görülebilir spektrumun sınırlı bir alanını yansıtan bir kütle, rengin bazı tonlarını gösterir. Örneğin yeşil nesneler öncelikle 500 ile 750 nm arası dalgaboyundaki ışığı yansıtırken, diğer dalgaboylarındaki enerjinin çoğunu emer.

Işığın karakterize edilmesi renk biliminin merkezindedir. Eğer ışık renkten yoksun (akromatik) ise, tek özelliği "şiddeti" veya miktarıdır. Akromatik ışık izleyicilerin siyah beyaz televizyonda gördükleri şeydir. Bölüm 2 de belirtildiği gibi ve şu ana kadar kullanıldığı üzere "gri seviye" terimi ışık şiddetin siyah, gri ve son olarak beyaza kadar değişen bir ölçüsüdür.

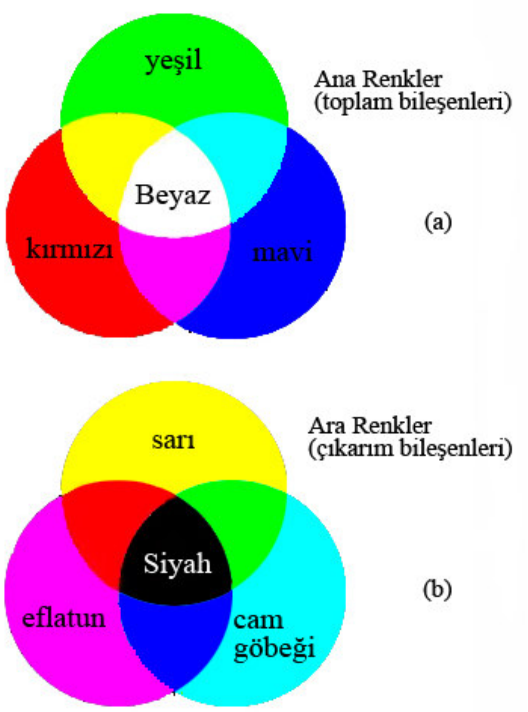
Renkten yoksun bir (kromatik) ışık kaynağını tanımlamak için Bölüm 2.2.2 de de anlatıldığı gibi üç temel nicelik kullanılır; Işıma-Aydınlık (Radiance), Parıldama-Parlama (Luminance), parlaklık (Brightness).

İnsan gözü üç temel duyarlılık katagorisine ayrılabilir; kabaca kırmızı, yeşil ve maviye karşılık gelir. Gözdeki bütün konilerin yaklaşık %65'i kırmızıya duyarlıdır, %33'ü yeşil ışığa duyarlıdır ve yalnızca yaklaşık %2'si maviye duyarlıdır (Fakat mavi koniler en duyarlı olanlardır). İnsan gözünün bu emiş özelliklerinden dolayı, renkler, ana renkler denilen renklerin (kırmızı R, yeşil G, mavi B) çeşitli kombinasyonları olarak görülmektedir. Standardizasyon amacıyla CIE (Commission Internationale de l'Eclairage – International Commission of Illumination – uluslar arası aydınlatma komisyonu) üç ana renk için şu dalgaboyu değerlerini belirlemiştir. Mavi = 435,8 nm, yeşil = 546,1 nm ve kırmızı = 700 nm. Bu nedenle CIE standartları deneysel eğrilerle yalnızca yaklaşık olarak örtüşmektedir. Bu yüzden hiçbir tek renk kırmızı, yeşil veya mavi olarak adlandırılamaz. Ayrıca unutulmaması gereken bir konuda şudur ki; standardizasyon amacıyla üç ana renk dalgaboylarına sahip olmak tek başına bu üç sabit RGB bileşenleri ile bütün spektrumun renklerini oluşturabilmek anlamına gelmez. Ana kelimesi genellikle yanlış anlaşılmaktadır, bu renklerin çeşitli şiddet oranlarında karşılaştırılınca "bütün" görülen renklerin oluşturabileceği sanılmaktadır.

Ana renkler, ışığın ara renklerini üretmek için karıştırılabilirler. Kırmızı+Mavi (Magenta veya eflatun), yeşil+mavi (cyan veya camgöbeği) ve kırmızı+yeşil (sarı). Üç ana renk, doğru yoğunluklarda karıştırılırsa beyaz elde edilir. Şekil 6.2(a)'da bu sonuç gösterilmektedir, ayrıca üç ana renk ve ara renkleri oluşturmak için kombinasyonları gösterilmektedir.

Işık ana renkleri arasında ve pigmentlerin veya renklendiricilerin ana renkleri arasında ayrım yapmak önemlidir. Sonrasında bir ana renk, ışığın bir ana rengini çıkaran veya emen ve diğer ikisini yansıtan veya ileten renk olarak tanımlanır. Bu yüzden pigmentlerin ana renkleri macenta, camgöbeği ve sarıdır. Ve ara renkleri de kırmızı, yeşil ve mavidir. Bu renkler Şekil 6.2(b)'de gösterilmiştir. Pigmentlerin üç

ana rengi tam olarak karıştırılırsa veya bir ara rengi zıt ana rengiyle karıştırılırsa siyah elde edilir.



Şekil 6.2. Toplama renk bileşenleri, çıkarma renk bileşenleri

Renkli televizyon, ışık renklerinin karışımı özelliğine bir örnektir. Çoğu renkli televizyonun tüpünün iç kısmı elektrona duyarlı fosforun üç köşeli nokta deseninde oluşturulmuş geniş bir dizisinden oluşmaktadır. Uyarıldığında, üçlü grup içindeki her bir nokta ışığın ana renklerinden birini üretme kabiliyetine sahiptir. Kırmızı yayan fosfor noktaların şiddeti tüpün içindeki bir elektron tabancası ile ayarlanır. Bu tabanca TV kamerası tarafından görülen "kırmızı enerjiye" karşılık gelen sinyaller üretir. Her üçlü gruptaki mavi ve yeşil fosfor noktaları da aynı şekilde ayarlanır.

Bir rengi diğerinden ayırmak için genelde kullanılan karakteristikler parlaklık, renk özü ve satürasyondur. Parlaklık, ışık şiddetinin akromatik ölçüsünü oluşturur. Renk özü, bir ışık dalgaları karışımdaki baskın bir dalgaboyu ile ilişkili olan bir niteliktir. Renk özü bir izleyici tarafından algılanan baskın rengi ifade eder. Bu nedenle, bir nesneye kırmızı, sarı yada turuncu dendiğinde, onun renk özü belirtilmiş olur.

Doygunluk, bir renk özü ile karıştırılmış olan beyaz ışığın nispi saflığı veya miktarını belirtir. Saf spektrum renkleri tam satüre edilmiştir. Pembe (kırmızı ve beyaz) ve lavanta (mor ve beyaz) gibi renkler, eklenen beyaz ışığın miktarına ters orantılı olan bir doygunluk derecesi ile az satüre edilmiştir.

Renk özü ve satürasyon birlikte alındığında "kromatiklik" adını alır ve böylece renk parlaklık ve kromatikliği ile karakterize edilebilir. Herhangi belirli bir rengi oluşturmak için ihtiyaç duyulan kırmızı, yeşil ve mavi miktarlarına tristimulus değerleri denir ve sırası ile *X, Y* ve Z olarak gösterilir. Böylece bir renk, üç renkli katsayıları ile belirtilir. Bu katsayılar Eş. 6.1- Eş. 6.3 ile ifade edilir [18]..

$$x = \frac{X}{X + Y + Z} \tag{6.1}$$

$$y = \frac{Y}{X + Y + Z} \tag{6.2}$$

$$z = \frac{Z}{X + Y + Z} \tag{6.3}$$

Eş 6.1, Eş 6.2 ve Eş 6.3'den, Eş. 6.4 elde edilir.

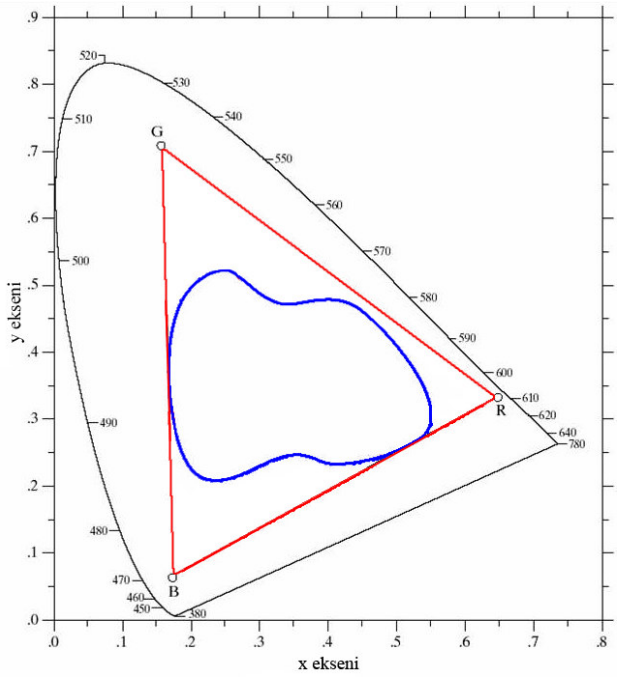
$$x + y + z = 1 \tag{6.4}$$

Renkleri belirtmek için CIE kromatiklik diyagramını kullanmaktır. Bu diyagram renk bileşimini *x* (kırmızı) ve *y* (yeşilin) bir fonksiyonu olarak gösterir. *x* ve *y*'nin herhangi bir değeri için karşılık gelen *z* (mavi) değeri $z= 1- (x + y)$'ye dikkat edilerek Eş. 6.4'ten elde edilir.

Çeşitli spektrum renklerinin pozisyonları dil şeklindeki kromatiklik diyagramının sınırları etrafında gösterilmiştir. Sınırın tam üzerinde olmayan fakat diyagram içinde olan herhangi bir nokta, spektrum renklerinin bir karışımını gösterir. Şekil 6.3'te gösterilen eşit enerji noktası üç ana rengin eşit parçalarına karşılık gelmektedir ve beyaz ışık için CIE standartları gösterir. Kromatiklik diyagramı renk karıştırmak için

kullanışlıdır çünkü diyagramdaki herhangi iki noktayı birleştiren düz çizgi, bu iki rengi birleştirerek elde edilebilen bütün farklı renk varyasyonlarını tanımlar. Örneğin şekil 6.3'teki kırmızı noktadan yeşil noktaya çizilmiş olan bir çizgi düşünün eğer yeşil ışıktan daha fazla kırmızı ışık olursa yeni rengi gösteren nokta çizgi üzerinde olacaktır fakat kırmızı noktaya yeşil noktadan daha yakın olacaktır. Benzer şekilde, eşit enerji noktasından baskın sınırı üzerindeki herhangi bir noktaya çizilen bir çizgi, o belirli spektrum renginin bütün tonlarını tanımlayacaktır.

Bu prosedürün üç renge genişletilmesi aynı şekildedir. Kromatiklik diyagramında verilen herhangi üç renkten elde edilebilen renklerin aralığını tanımlamak için basitçe her üç noktayı birleştiren çizgiler çizilir. Sonuç bir üçgendir ve bu üçgen içindeki her renk, bu üç rengin değişik kombinasyonları ile üretilebilir. Herhangi üç "sabit" renkteki diklerden oluşan bir üçgen, görünür banttaki bütün renk bölgesini kapsayamaz. Bu durum Şekil 6.3'de grafik olarak görülmektedir ki; üç, sabit ana renklerle bütün renkler elde edilemez.



Şekil 6.3. C.I.E. kromatiklik grafiği, kırmızı üçgen renkli monitörlere ait, mavi şekil ise renkli yazıcılara ait renk aralığını göstermektedir [18].

Şekil 6.3'teki üçgen, RGB monitörler tarafından üretilen tipik bir renk aralığını gösterir. Üçgen içindeki düzensiz bölge bugünün yüksek kalite yazıcılarının renk aralığını temsil etmektedir. Renkli yazma gamının sınırı düzensizdir çünkü renkli yazma işlemi eklemeli ve çıkarmalı bir renk karıştırma işlemidir; bunu kontrol etmek renkleri monitör üzerinde göstermekten daha zordur, çünkü monitör çok iyi kontrol edilebilen ışık ana renklerini karışımı üzerine kuruludur.

6.2 Renk Modelleri

Bir renk modelinin (renk uzayı veya renk sistemi olarak da adlandırılır) amacı, renklerin bir standartta, genellikle kabul edilen şekilde, belirlenmesini kolaylaştırmaktır. Esasında bir renk modeli, bir koordinat sisteminin ve sistem içindeki her rengin tek bir nokta ile gösterildiği bir alt başlığın belirtilmesidir.

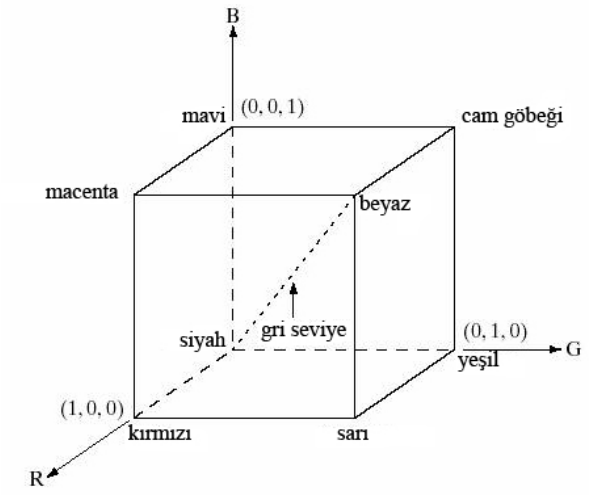
Bugün kullanılan çoğu renk modeli donanıma yöneliktir (monitörler ve yazıcılar gibi) veya renk karışımının amaçlandığı uygulamalara yöneliktir (animasyon için renkli grafiklerin oluşturulması gibi). Dijital resim işleme yoluyla pratikte en yaygın olarak kullanılmakta olan donanıma yönelik modelleri RGB (Red, Green, Blue – Kırmızı, Yeşil, Mavi) modeli; renkli monitörler ve geniş bir sınıf video kameralar için CMY (Cyan, Magenta, Yellow – Camgöbeği, Macenta, Sarı) modeli ve CMYK (Cyan, Magenta, Yellow, Black – Camgöbeği, Macenta, Sarı, Siyah) modelleri; renkli yazım için HSI (Hueg Saturation Intensity – Renk özü, Satürasyon, Şiddet) modeli ki bu model insanın rengi tanımlama ve yorumlama yoluna yakındır.

Ayrıca HSI modelinin bir avantajı da şudur; bir resimdeki renk bilgisini ve gri-seviye bilgisini ayırır. Bu durumda bu modeli çoğu gri-seviye teknikleri için uygun hale getirir.

6.2.1 RGB renk modeli

RGB modelinde her renk, kendi kırmızı, yeşil ve mavisinin ana spektrumsal bileşeni olarak görünür. Bu model bir kartezyan koordinat sistemi üzerine kuruludur.

Bahsedilen renk alt boşluğu Şekil 6.4'de gösterilen küptür. Burada RGB değeri üç köşededir; camgöbeği, macenta ve sarı diğer üç köşede; siyah orijinde ve beyaz orjinden en uzak köşededir. Bu modelde gri skala (eşit RGB değerleri noktaları) siyahtan beyaza kadar, bu iki noktayı birleştiren çizgi boyunca uzanır. Bu modeldeki farklı renkler küpün üstünde veya içindeki noktalardır ve orijinden çıkan vektörlerle tanımlanır.



Şekil 6.4. RGB birim renk küpü

RGB renk modelinde gösterilen resimler üç bileşen resminden oluşur; her ana renk için bir tane. Bir RGB monitörde bu üç resim birleşik bir renkli resim oluşturmak için fosfor ekran üzerinde birleşir. RGB uzayındaki her pikseli temsil etmek için kullanılan bit sayısına "piksel derinliği" denir. Her kırmızı, yeşil ve mavi resmin bir 8 bit resim olduğu bir RGB resmi düşünün. Bu durumda her RGB renk pikseli (yeni bir RGB değerleri üçlüsü) 24 bit'lik bir derinliğe sahiptir (her yüzey için üç resim yüzeyi). "Tam renk" terimi genellikle bir 24 bit RGB renkli resmi göstermek için kullanılır. Bir 24 bit RGB resminde ki toplam renk sayısı $(2^8)^3 = 16.777.216$'dır.

Bugün kullanılan sistemlerin çeşitliliği ile izleyici donanımların kapasitesinde olan, bir alt küme renklerinin yeniden üretilmesi düşünülebilir. Bu renklerin alt kümesine "güvenilir" RGB renkleri denir veya "bütün sistemlerde güvenilir" renk kümesi denir. İnternet uygulamalarında bunlara "güvenilir web" renkleri veya "güvenilir

browser" renkleri denir. 256 rengin herhangi bir sistem tarafından üretilebilen minimum renk sayısı olduğunu varsayarsak, bu renkleri gösteren, kabul edilmiş bir standart gösterime sahip olmak kullanışlı olur. Bu 256 rengin 40'ı farklı işletim sistemleri tarafından farklı olarak işlenir, geriye kalan 216 renk çoğu sistemlerde ortaktır. Bu 216 renk güvenilir renkler için, özellikle internet uygulamalarında, fiili standart haline gelmiştir. Bunlar, çoğu insanların baktığı rengin aynı görünmesinin istendiği her zaman kullanılırlar.

256 güvenilir rengin her biri, daha önce olduğu gibi, üç RGB değeri ile oluşur, fakat her değer yalnızca 0, 51, 102, 153, 204 veya 255 olabilir (burada bütün değerler 3'e bölünebilir). Bu yüzden bu değerlerin RGB üçlüleri bize $(6)^3 = 216$ olası değeri verir. Bu değerleri, onaltılık sayı sisteminde ifade etmek alışılmış kullanımdır. $(0)_{16} = (0000)_2$ ve $(F)_{16} = (1111)_2$'dir. Bu yüzden örneğin, $(FF)_{16} = (255)_{10} = (11111111)_2$ ve iki basamaklı onaltılık sayıyı gruplarsak bir 8 bit yani 1 byte oluşur.

6.2.2 CMY ve CMYK renk modelleri

Bölüm 6.1'de belirtildiği gibi camgöbeği, macenta, sarı ışığın ara renkleridir veya diğer deyişle pigmentlerin ana renkleridir. Örneğin camgöbeği pigmenti ile kaplanmış bir yüzey beyaz ışık ile aydınlatılınca, yüzeyden kırmızı ışık yansımaz. Yani camgöbeği, kırmızı ışığı, kırmızı, yeşil ve mavinin eşit miktarıyla oluşan yansıyan beyaz ışıktan çıkarır.

Yazıcılar ve fotokopiler gibi renkli pigmentleri kağıt üzerinde toplayan çoğu aygıtlar, içlerinde RGB' den CMY' ye dönüşüm için CMY data girişine ihtiyaç duyarlar. Bu dönüşüm şu basit işlem kullanılarak yapılır [18];

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{6.5}$$

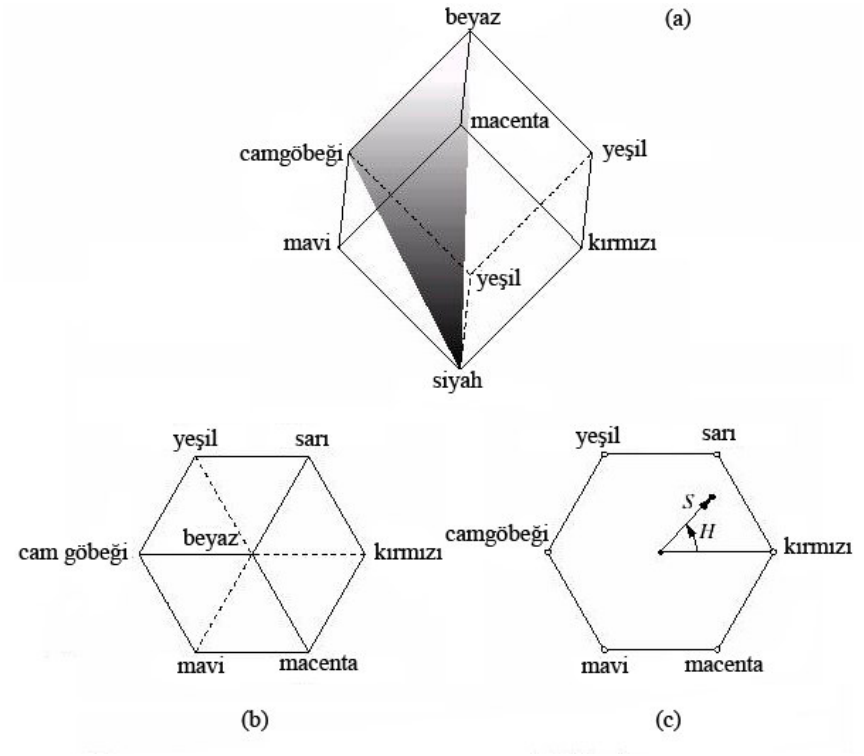
Burada yine bütün renk değerlerinin [0, 1] aralığında olduğu varsayılmıştır. Eş.6.5'den görülüyor ki saf camgöbeği ile hesaplanmış bir yüzeyden yansıyan ışık kırmızıyı içermez (yani denklemde C = 1- R): benzer şekilde, saf macenta yeşili yansıtmaz ve saf sarı da maviyi yansıtmaz. Ayrıca Eş.... şunu ortaya çıkarıyor ki RGB değerleri, bir grup CMY değerleri kullanılarak kolayca elde edilebilir; her CMY değeri 1'den çıkarılır. Genel olarak resim işlemede bu renk modeli, donanımsal çıktı elde etmek ile ilgili olarak kullanılmaktadır, dolayısıyla, tersi olan CMY' den RGB' ye dönüşüm işlemi pratikte pek kullanılmaz. Şekil 6.2(b)'ye göre pigment analarının (camgöbeği, macenta ve sarı) eşit miktarda karışımı siyahı oluşturması gerekir. Pratikte yazdırma işlemi için bunları birleştirirsek çamurumsu bir siyah ortaya çıkar. Dolayısıyla, gerçek siyahı (yazdırmada baskın renktir) elde etmek için CMYK renk modelini ortaya çıkaran, dördüncü bir renk, siyah, eklenir. Bu yüzden, satıcılar "dört-renkli yazımdan" bahsederken CMY' nin üç rengi artı siyahı kastetmektedirler.

6.2.3 HSI renk modeli

RGB ve CMY modelleri, donanım uygulamaları için idealdir. Ayrıca RGB sistemi, insan gözünün kırmızı, yeşil ve mavi ana renklerine çok hassas olduğu gerçeğine iyi bir şekilde uymaktadır. Maalesef RGB, CMY ve benzeri diğer renk modelleri, pratikte insanın algıladığı renkleri tanımlamada tam uygun değildirler. Örneğin, bir otomobilin rengini söylerken, onun rengini oluşturan ana renklerin yüzdelerini vermek yeterli değildir.

İnsan algısı bir renkli nesneyi, nesnenin renk özü, satürasyonu, ve parlaklığı ile tanımlar. Bölüm 6.1'de de belirtildiği gibi, renk özü (hue), saf rengi (saf sarı, turuncu veya kırmızı) tanımlayan bir renk özelliğidir. Satürasyon ise bir saf rengin beyaz ışıkla sulandırılma derecesinin ölçüsünü verir. Parlaklık pratikte ölçülmesi imkansız olan sübjektif bir tanımlayıcıdır. Işık şiddetinin renksiz ölçüsünü oluşturur ve renk hissini tamamlayan anahtar faktörlerden biridir. Işık şiddeti (gri seviyesi) tek renkli resimlerin en kullanışlı tanımlayıcılarından biridir. Bu özellik tam olarak ölçülebilir ve yorumlanabilir. HSI renk modeli, bir renkli resimdeki şiddet bileşenini, rengi

taşıyan bilgiden (renk özü ve satürasyon) ayırır. Sonuç olarak HSI modeli insanlar için doğal ve sezgisel olan renk tanımlamaları üzerine kurulu resim işleme algoritmaları geliştirmek için ideal bir araçtır. Şu şekilde özetlenebilir; RGB resim rengi üretmek için idealdir (renkli kamerayla resim yakalama veya renkli monitör ekranında resim görüntülemede ki gibi) fakat rengi tanımlamak için kullanımı daha sınırlıdır. Bir RGB resminden ışık şiddeti çıkarabilir.



Şekil 6.5. RGB ve HSI renk modelleri arasındaki ilişki, renk özü ve doygunluk bileşenlerinin gösterimi [18]

Renk özünün verilen bir RGB noktası nasıl belirlendiğini anlamak için, Şekil 6.5(a)'yi bakıldığında düzlem (siyah beyaz camgöbeği düzlemi) içerisinde siyah ve beyaz noktaların olduğu gerçeğinden; düzlem, ışık şiddeti eksenini de içerdiği anlaşılır. Ayrıca buradan, şiddet ekseni ve küpün sınırları ile tanımlanmış düzlem kısmındaki bütün noktaların aynı renk özüne (bu durumda camgöbeği) sahip olduğu sonucu da çıkmaktadır. Bölüm 6.1'de de bahsedildiği gibi üç renk ile oluşturulan

bütün renkler, o renkler ile tanımlanan üçgen içerisindedirler. Eğer o noktaların ikisi siyah be beyaz, ve üçüncüsü de bir renkli olursa, üçgen üzerindeki bütün noktalar aynı renk özüne sahip olur, çünkü siyah ve beyaz bileşenler renk özünü değiştirmezler. (tabi ki bu üçgendeki noktaların şiddeti ve satürasyonu farklı olur). Renk tonu düzlemi dikey şiddet ekseni etrafında döndürülerek farklı renk özleri elde edilir. Böylece, HSI uzayını oluşturmak için gereken renk özü, satürasyon ve şiddet değerleri, RGB renk küpünden elde edilebilir. Buradan hareketle herhangi bir RGB noktasının, HSI renk modelinde karşılık gelen noktaya çevrilebileceği sonuçu çıkar.

Şekil 6.5(a)'deki küp düzenlemesi ve ona karşılık gelen HSI renk uzayı ile ilgili olarak akılda tutulması gereken HSI uzayı bir dikey şiddet ekseni ve bu eksene dikey düzlemeler üzerindeki renk noktaları grubu ile gösterilir olduğudur. Düzlemler şiddet ekseni boyunca yukarı ve aşağı doğru hareket ettikçe, her düzlemin, küpün yüzeyleri ile kesişmesi ile tamamlanan sınırlar üçgensel veya altıgensel bir şekil alırlar. Bu durum, Şekil 6.5(b)'de gösterildiği gibi küpün gir-skala eksenine bakarak daha kolay görülebilir. Bu düzlemde ana renklerin 120^0 ile ayrıldığı görülmektedir. Ara renkler, ana renklerden 60^0 uzaktır, buda demektir ki ara renler arasındaki açı farkı da 120^0 dir. Şekil 6.5(c)'de aynı altıgensel şekil ve rasgele bir renkli nokta (bir nokta olarak gösterilmiş) görülmektedir. Noktanın renk özü bir frekans noktasından olan açıyla belirlenir. Genellikle kırmızı eksenden 0^0'lik bir açı o renk özünü belirtir ve renk özü oradan itibaren saatin tersi istikametinde artar. Satürasyon (düşey eksen) orijinden noktaya uzanan vektörün uzunluğudur. Burada görüldüğü gibi orijin, renk düzlemi ile şiddet ekseninin kesişmesi ile tanımlanır. HSI renk uzayının önemli bileşenleri dikey şiddet ekseni; bir renk noktasına olan vektörün uzunluğu, ve bu vektörün kırmızı eksen ile yaptığı açıdır.

6.3 Renkleri Modelleri Çevirme

6.3.1 Renkleri RGB'den HSI'ya çevirme

RGB renk formatında verilen bir resimde, her RGB piksel' inin H bileşenini elde etmek için Eş. 6.6 kullanılır;

$$H = \begin{cases} \theta & \text{eger } B \le G \\ 360 - \theta & \text{eger } B > G \end{cases} \tag{6.6}$$

Burada θ Eş.6.7 den elde edilir.

$$\theta = \cos^{-1}\left\{ \frac{\frac{1}{2}\left[(R-G)+(R-B)\right]}{\left[(R-G)^2+(R-G)(G-B)\right]^{1/2}} \right\} \tag{6.7}$$

Doygunluk

$$S = 1 - \frac{3}{(R+G+B)}\left[\min(R,G,B)\right] \tag{6.8}$$

Işık Şiddeti

$$I = \frac{1}{3}\left[R+G+B\right] \tag{6.9}$$

dir [18]. RGB değerlerinin [0, 1] aralığında olduğu ve θ açısının HSI uzayının kırmızı eksenine bağlı olarak ölçüldüğü (şekil 6.5(c)'de gösterildiği gibi) varsayılmıştır. Renk özü, Eş.6.6'dan çıkan bütün değerlerin 360^0 ile bölünmesi ile [0, 1] aralığında oluşturulabilir. Diğer iki HSI bileşeni, eğer verilen RGB değerleri [0, 1] aralığında ise, zaten bu aralık içindedirler.

6.3.2 Renkleri HSI'den RGB'ye çevirme

HSI'nın [0, 1] aralığında verilen değerleri için, aynı aralığa karşılık gelen RGB değerlerini bulmak amacıyla uygulanabilir denklemler H'nin değerlerine dayanır. Ana renklerin ayrımındaki 120^0 lik aralıkları gösteren, üç sektör vardır. Öncelikle H 360 ile çarpılarak renk özü orijinal aralığa olan [0^0, 360^0] aralığına döndürülür [18].

RG Bölgesi $(0° \le H < 120°)$: H bu bölgede iken RGB bileşenleri ,

$$B = I\ (1 - S\ ) \tag{6.10}$$

$$R = I\left[1 + \frac{S\cos H}{\cos(60° - H)}\right] \tag{6.11}$$

$$G = 3I - (R + B) \tag{6.12}$$

GB Bölgesi $(120° \le H < 240°)$: H bu bölgede ise;

$$H = H - 120° \tag{6.13}$$

dir. Bu durumda RGB bileşenleri Eş. 6.14 - Eş. 6.16 ile elde edilir.

$$R = I(1 - S) \tag{6.14}$$

$$G = I\left[1 + \frac{S\cos H}{\cos(60° - H)}\right] \tag{6.15}$$

$$B = 3I - (R + G) \tag{6.16}$$

BR Bölgesi $(240° \le H < 360°)$: H bu bölgede ise;

$$H = H - 240° \tag{6.17}$$

dir. Bu durumda RGB bileşenleri Eş. 6.18 - Eş. 6.20 ile elde edilir.

$$G = I(1 - S) \tag{6.18}$$

$$B = I\left[1 + \frac{S\cos H}{\cos(60° - H)}\right] \tag{6.19}$$

$$R = 3I - (G + B) \tag{6.20}$$

6.3.3 Gri seviyeden renkliye dönüşümler

Bu tip dönüşüm için uygun bir yaklaşım, herhangi bir giriş pikselinin gri seviyesinde üç bağımsız dönüşüm yapmaktır. Daha sonra üç sonuç ayrı ayrı bir renkli donanımın (monitör veya tv vb.) kırmızı, yeşil ve mavi kanallarında beslenir. Bu metot renk içeriği dönüşüm fonksiyonları ile ayarlanabilen karmaşık bir görüntü oluşturur. Bunlar görüntünün gri seviye değerlerindeki dönüşümler olup pozisyon fonksiyonu değillerdir. Gri seviyelerin parçalı doğrusal fonksiyonları renkleri oluşturmak için kullanılır. Diğer taraftan bu metot, teknik esnekliği vermesi beklenen düzgün doğrusal olmayan fonksiyonlara dayanabilir.

6.4 Tam-Renk Görüntü İşleme Temelleri

Tam-renk görüntü işleme yaklaşımları iki ana kategoriden oluşur. İlk kategoride, her bir bileşen görüntüsü tek tek işleme tabi tutulur ve daha sonra tek tek işlenmiş bileşenlerden işlenmiş renkli görüntü bileşiği oluşturulur. İkinci kategoride, doğrudan renk pikselleriyle çalışılır. Çünkü tam-renkli görüntülerin en az üç bileşeni vardır, renk pikselleri gerçekten vektörlerdir. Örneğin, RGB sisteminde, her bir renk noktası RGB koordinat sisteminde merkezden o noktaya uzanan bir vektör olarak yorumlanabilir. RGB renk uzayında $\bar{c}$ herhangi bir vektörü temsil etsin:

$$\bar{c} = \begin{bmatrix} c_R \\ c_G \\ c_B \end{bmatrix} = \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{6.21}$$

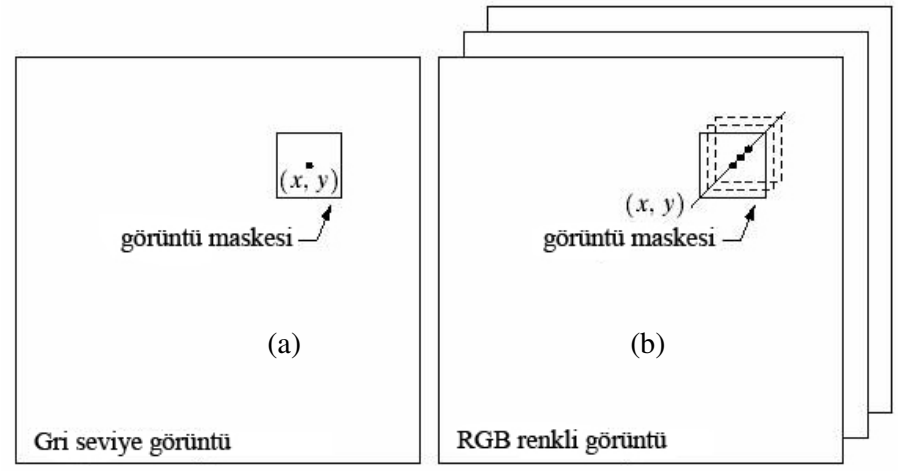
$\bar{c}$'nin bileşenleri noktadaki renkli görüntünün RGB bileşenleridir. Eş.6.21 ile renk bileşenlerinin, (x,y) koordinatlarının bir fonksiyonu olduğu ifade edilmektedir [18].

$$\bar{c}(x,y) = \begin{bmatrix} c_R(x,y) \\ c_G(x,y) \\ c_B(x,y) \end{bmatrix} = \begin{bmatrix} R(x,y) \\ G(x,y) \\ B(x,y) \end{bmatrix} \tag{6.22}$$

MxN boyutundaki bir görüntü için, MN gibi vektörler vardır, $\bar{c}(x,y)$, x = 0,1,2,...,M-1, y=0,1,2,...,N-1.

Eş.6.22'nin, bileşenleri x ve y'de uzaysal değişkenler (piksel düzlemi) olan bir vektörü ifade ettiğini açıkça göz önünde bulundurmak önemlidir. Piksellerin şimdi renk pikselleri olması gerçeği, standart gri skala görüntü işleme metotları kullanarak renkli görüntünün her bir bileşen görüntüsünü ayrı ayrı işleyerek renkli görüntü işlemeye izin veren bir faktör ortaya çıkarır. Buna rağmen, tek renk bileşeni işleme sonuçları her zaman renkli vektör uzayında doğrudan işleme ile eşit olmaz.

Tek-renkli bileşen ve vektör tabanlı işlemenin eşit olması için, iki şart sağlanmalıdır: İlk olarak, işlem hem vektörlere hem de sayısallara uygulanabilir olmalıdır. İkinci şart ise, vektörün her bir bileşenindeki işlemin diğer bileşenlerden bağımsız olmasıdır. Şekil 6.6 gri-skala ve tam-renkli görüntülerin piksel komşuluk uzayı işlemesini göstermektedir. Prosesin komşuluk ortalama olduğu varsayılmıştır. Şekil 6.6 (a)'da ortalama, komşuluktaki tüm piksellerin gri seviyelerinin toplanması ve komşuluktaki toplam piksel sayısına bölünmesi ile bulunur. Şekil 6.6 (b)'de ortalama, komşuluktaki tüm vektörlerin toplamı ve her bir bileşenin komşuluktaki toplam vektör sayısına bölümü ile elde edilir.



Şekil 6.6. Gri-skala ve RGB renkli görüntüler için uzaysal maskeler [18]

6.5 Renk Dönüşümleri

Renk dönüşümleri olarak adlandırılan teknikler, bileşenlerin modeller arasındaki dönüşümüne (RGB'den HSI'ya ve HSI'dan RGB'ye değişim dönüşümü gibi) karşı olarak renkli bir görüntünün bileşenlerinin tek renk modeli bağlamında işlenmesi ile ilgilidir.

6.5.1 Formülasyon

Bölüm 3'teki gri-seviye dönüşüm teknikleri ile, Eş.6.23 kullanılarak renk dönüşümleri modellenebilir:

$$g(x,y) = D[f(x,y)] \tag{6.23}$$

Burada f(x,y) renkli giriş görüntüsünü, g(x,y) dönüştürülmüş veya işlenmiş renkli çıkış görüntüsünü ve D ise (x,y)'nin görüntü koodinatları komşuluğu üstünden f'deki bir işlemi ifade etmektedir. Şekil 6.6(b)'de gösterildiği gibi görüntüyü ifade etmek için renkli görüntü koordinat uzayından seçilen piksel değerleri üçlü ve dörtlüdür (üç veya dört değerli gruplar).

$$s_i = D_i(r_1, r_2,...r_n), \qquad i=1,2,...n \tag{6.24}$$

Burada, notasyonal basitlik için, r_i ve s_i herhangi bir (x,y) noktasındaki f(x,y) ve g(x,y)'nin renk bilşenlerini ifade eden değişkenlerdir, n renk bilşenlerinin sayısıdır ve $\{D_1, D_2,...,D_n\}$ ise s_i'yi oluşturmak için r_i üzerinde işlem yapan *dönüşüm* veya *renk eşleme fonksiyonları* kümesidir. Buradaki n tane dönüşümün, D_i, Eş6.23'deki tek dönüşüm fonksiyonu D'yi oluşturmak içindir. f ve g'nin piksellerini tanımlamak için seçilen renk uzayı n'nin değerini belirler. Örneğin eğer RGB renk uzayı seçilirse, n = 3 ve r_1, r_2 ve r_3 giriş görüntüsünün sırasıyla kırmızı, yeşil ve mavi bileşenlerini gösterir. Eğer CMYK veya HSI renk uzayları seçilirse, n = 4 veya n = 3 olur.

Son olarak, renk tonu (hue) bileşenlerini yorumlamada aşağıda belirtilen durumlardan kaynaklanana zorluklar oluşabilmektedir.

1) HSI modelinde 0º ve 360º'nin birleşim yerinde devamsızlık vardır.
2) Renk tonu (hue) 0 doyumda tanımlı değildir (örneğin, beyaz, siyah ve saf gri için).

Herhangi bir renk uzayı bileşeni Eş.6.24 ile birleşiminde kullanılabilir. Teoride, herhangi bir dönüşüm herhangi bir renk modeline uygulanabilir. Buna rağmen pratikte, bazı işlemler belirli modellere daha uygun olur. Verilen bir dönüşüm için, resimler arasındaki çevrim bedeli uygulanacak renk uzayı hakkındaki karara bağlıdır. Örneğin, Eş.6.25 kullanarak herhangi bir görüntünün yoğunluğunu değiştirilebilir.

$$g(x,y) = kf(x,y) \tag{6.25}$$

Burada $0<k<1$'dir. HSI renk uzayında, bu Eş.6.26 daki basit dönüşümle yapılabilir.

$$s_3 = kr_3 \tag{6.26}$$

Burada $s_1=r_1$ v $s_2=r_2$'dir. Sadece HSI yoğunluk bileşeni r_3 değiştirilmiştir. RGB renk uzayında, üç bileşenin dönüştürülmesi gerekmektedir:

$$s_i = kr_i \qquad i = 1,2,3 \tag{6.27}$$

CMY uzayı benzer bir doğrusal dönüşüm kümesi gerektirmektedir:

$$s_i = kr_i + (1-k) \qquad i = 1,2,3 \tag{6.28}$$

HSI dönüşümü daha az sayıda işlem gerektirmesine rağmen, RGB veya CMY(K) görüntüden HSI uzayına dönüşüm için gerekli hesaplamalar basit dönüşümün avantajlarını dengelemektedir. Çevrim hesaplamaları yoğunluk dönüşümlerinden

daha yorucudur. Buna rağmen seçilen renk uzayına aldırmadan, çıkış aynıdır. Tanımlanan her bir dönüşümün sadece renk uzayındaki bir bileşene bağlıdır. Örneğin, Eş.6.27'deki kırmızı çıkış bileşeni, s_1, yeşil (r_2) ve mavi (r_3) girişlerden bağımsızdır, sadece kırmızı (r_1) girişe bağlıdır. Bu tip dönüşümler, en kolaylar arasındadır ve renk işleme araçlarında çoğunlukla kullanılır. Başlangıcında da belirtildiği gibi tek-renkli bileşen esaslı gerçekleştirilebilirler [18].

6.5.2 Histogram işleme

Bölüm 3.3'ten de hatırlanacağı gibi histogram eşitleme gri seviye (intensity) değeri üniform hale getirilmiş görüntü elde etmeyi sağlayan bir dönüşümdür. Tek renkli görüntüde bunu uygulamak daha iyi sonuçlar vermektedir. Fakat renk görüntü bileşenlerden oluştuğu için gri seviyede verilen tekniğin uyarlanması gerekir. Eğer renkli görüntünün bileşenleri için ayrı ve bağımsız işlem uygulanırsa yanlış sonuçlar elde edilir. Renkli görüntünün renk bileşenlerinin ışık şiddeti seviyelerini birleştirip işlemleri onun üzerinde uygulamak ve renk bileşenlerini hiç değiştirmemek daha uygun bir yaklaşımdır. Bunun için renk modeli olarak HSI modeli seçmek bu işlemde daha iyi ve doğru sonuç alınmasını sağlar. HSI renk modelinde renk bileşeni ve ışık şiddeti bileşeni ayrı ayrı ifadesi gri seviye görüntüye daha çok benzeyen bir yapıdır. Buda histogram işlemeyi daha kolay ve daha uygun şekilde yapabilmeyi sağlar.

6.5.3 Renkli görüntü yumuşatma

Bölüm 3.4'te de belirtildiği gibi yumuşatma işlemi, tüm değerleri bir olan filtre maskesinin tüm görüntü pikselleri üzerinde uygulanması ile elde edilmektedir. Her pikselin komşuluğunun ortalaması alınarak (maske elemanının filtre olarak uygulanmasıyla) yumuşatılmış görüntü elde edilmekteydi. Renkli görüntüde gri seviyeden temel farkı gri seviye skaler değeri yerine renkli görüntünün bileşen vektörleri kullanılmaktadır. Bu vektör bileşenleri Eş.2.29 dur. RGB renkli görüntüde (x, y) noktasının komşularının koordinatları S_{xy} ile gösterilsin bu durumda RGB bileşen vektörlerinin bu komşuluk için ortalaması Eş.6.30 ile elde edilir [18].

$$\bar{c}(x, y) = \frac{1}{K} \sum_{(s,t) \in S_{xy}} c(s,t) \tag{6.29}$$

$$\bar{c}(x, y) = \begin{bmatrix} \frac{1}{K} \sum_{(s,t) \in S_{xy}} R(s,t) \\ \frac{1}{K} \sum_{(s,t) \in S_{xy}} G(s,t) \\ \frac{1}{K} \sum_{(s,t) \in S_{xy}} B(s,t) \end{bmatrix} \tag{6.30}$$

Bu vektörün her bir bileşeni her düzlemin yumuşatmasının bir birinden bağımsız olarak elde edildiği skaler görüntüdür. yani piksel komşuluk ortalaması kullanılarak yumuşatma her bir renk düzleminin işlenmesi ile yapılır

6.5.4 Renkli görüntü keskinleştirme

Renkli görüntüyü keskinleştirme için Laplacian operatörü kullanılabilir. Laplacian vektör bileşenleri giriş vektöründe ki her bir skaler bileşenin Laplace ile tanımlanan vektördür. RGB renk sisteminde bir *c* vektörünün laplace'si Eş.6.31' dir.

$$\nabla^2 [c(x, y)] = \begin{bmatrix} \nabla^2 R(x, y) \\ \nabla^2 G(x, y) \\ \nabla^2 B(x, y) \end{bmatrix} \tag{6.31}$$

Burada da görüldüğü gibi her bir renk bileşeninin laplace'si alınarak renkli görüntü keskinleştirmesi uygulanır.

6.5.5 RGB görüntüde bölümlendirme

HSI renk modeli RGB modelinden rengin renk bileşenlerine uygun olarak yeniden ifade edilebilmesi, aynı şekilde doygunluk bileşeni kullanarak görüntünün renk bileşeninde ki istenilen rengin uzak bölgelerinin filtrelenebilmesi nedenlerinden dolayı bölümlendirme işleminde çok daha kolay uygulanabilir ve çok daha iyi

sonuçlar vermektedir. HSI modelinde ışık şiddeti herhangi bir renk bileşeni bilgisi içermediğinden renkli görüntü bölümlendirmede pek kullanılmamaktadır.

Bölümlendirmek istenilen rengin tahmini ortalama değeri, bu rengi temsil eden verilen bir örnek renk noktaları setinden elde edilir. Bu ortalama renk değeri "a" RGB vektörü ile ifade edilsin. Bölümlendirme de hedef verilen resim için her bir RGB pikselinin renginin istenen aralıkta olup olmadığının sınıflandırmasıdır. Bu karşılaştırmayı yapmak için benzerlik ölçüsüne sahip olmak gereklidir. Ölçüm Euclidean uzaklığı ile alınabilir. z RGB'de bir nokta olsun. z ve "a" arasındaki mesafe, tanımlanan D_0 gibi bir eşik değerinden az ise z ve a benzerdir denir. z ile a arasındaki Euclidean mesafe Eş.6.32 ile elde edilir.

$$\begin{aligned} D(z,a) &= \|z-a\| \\ &= \left[(z-a)^T(z-a)\right]^{1/2} \\ &= \left[(z_R-a_R)^2+(z_G-a_G)^2+(z_B-a_B)^2\right]^{1/2} \end{aligned} \tag{6.32}$$

$D(z,a) \le D_o$ mantıksal koşulana göre bölümlendirilmiş ikili düzey görüntü elde edilir. Eş.6.32'nin genelleştirilmiş formu

$$D(z,a) = \left[(z-a)^T C^{-1}(z-a)\right]^{1/2} \tag{6.33}$$

dir. Burada C bölümlendirmek istenilen rengin örnek ifadelerini içeren kovaryans matrisidir [18].

6.5.6 Renk kenar belirleme

Bölüm 3.4'te gradient operatörünün uygulanışı gösterilmiştir. Fakat burada gradient işlemi vektör büyüklükler üzerinde gösterilmemiştir. RGB görüntüde bileşik gradient görüntüyü elde etmek için her bir resim düzleminin ayrı ayrı işlenmesi yanlış sonuçlar verir. Gradient işlemini vektörel nicelikler üzerinde uygulamak için Di Zenzo metodu kullanılabilir.

Gradient işlemi skalar fonksiyon olan f(x,y) için uygulanabilirken, vektör fonksiyon için uygulanabilir değildir. Gradient f(x,y) skaler fonksiyonu için f'nin (x, y) koordinatlarında maksimum değişim oranında ki doğrultusunu veren vektördür.

r, g, b sırasıyla RGB uzayının birim vektörü olsunlar. Bu vektörler Eş.6.34 ve Eş.6.35 ile ifade edilebilir.

$$u = \frac{\delta R}{\delta x} r + \frac{\delta G}{\delta x} g + \frac{\delta B}{\delta x} b \tag{6.34}$$

$$v = \frac{\delta R}{\delta y} r + \frac{\delta G}{\delta y} g + \frac{\delta B}{\delta y} b \tag{6.35}$$

Bu vektörlerin vektörel çarpımları g_{xx}, g_{yy}, g_{xy} olsun. Bu çarpımlar Eş.6.36, - Eş.6.38 ile ifade edilmiştir.

$$g_{xx} = u \bullet u = u^T u = \left|\frac{\delta R}{\delta x}\right|^2 + \left|\frac{\delta G}{\delta x}\right|^2 + \left|\frac{\delta B}{\delta x}\right|^2 \tag{6.36}$$

$$g_{yy} = v \bullet v = v^T v = \left|\frac{\delta R}{\delta y}\right|^2 + \left|\frac{\delta G}{\delta y}\right|^2 + \left|\frac{\delta B}{\delta y}\right|^2 \tag{6.37}$$

$$g_{xy} = u \bullet v = u^T v = \frac{\delta R}{\delta x}\frac{\delta R}{\delta y} + \frac{\delta G}{\delta x}\frac{\delta G}{\delta y} + \frac{\delta B}{\delta x}\frac{\delta B}{\delta y} \tag{6.38}$$

$\bar{c}\,(x, y)$ için maksimum değişim oranının doğrultusu Eş.6.39 dur ve bu doğrultuda ki maksimum değişim değeri Eş.6.40 dir.

$$\theta = \frac{1}{2}\tan^{-1}\left[\frac{2g_{xx}}{(g_{xx} - g_{yy})}\right] \tag{6.39}$$

$$F(\theta) = \left\{\frac{1}{2}\left[(g_{xx} + g_{yy}) + (g_{xx} + g_{yy})\cos 2\theta + 2g_{xy}\sin 2\theta\right]\right\}^{\frac{1}{2}} \tag{6.40}$$

Bu gösterim Di Zenzo yazılımı olarak bilinir [18].

7. GÖRÜNTÜ İŞLEME UYGULAMASI ve SONUÇLARI

Bu bölümde kurulan uygulama düzeneği, sunulan görüntü işleme tekniklerinin uygulanması ve elde edilen sonuçlar bulunmaktadır. Uygulamamın ilk bölümünde kurulan düzenekle alınan görüntülerden, domino taşları görüntü işleme teknikleri kullanılarak bilgisayar ortamında taşlar üzerindeki noktalara göre tanımlanmıştır.

İkinci bölüm uygulamada ise küresel grafit içeren dökme demir, martenzit çelik ve işlemsiz çelik görüntüleri kullanılmıştır. İkinci bölümde birinci bölüm uygulamadaki işlemlere eklenti yapılarak görüntüler içerisindeki işlemsiz çelik için iki fazdan birinin toplam piksel alanı, küresel grafit için ise yüzde grafit oranı, grafit piksel alanı, görüntü işleme teknikleri ile bilgisayar ortamında belirlenmiştir.

7.1 Birinci Bölüm Uygulama (Domino Taşlarını Tanıma)

7.1.1 Sistem

Kurulan uygulama düzeneği aşağıdaki donanımlardan oluşmuştur.

- Panasonic NV-GS75 3 CCD MiniDV Kamera ve kamera sabitleme sehpası
- Harici aydınlatma kaynağı
- IEEE1394 FireWire Arayüzü
- AMD Athlon64+ 3GHz işlemcili, üzerinde Windows XP işletim sistemi ve görüntü işleme araçlarını içeren yazılım yüklü bir PC.

CCD kamerayı Domino taşlarının dizili olduğu masaya dik bir şekilde monte edilmiştir. Bunun için kamera sabitleme sehpası kullanmıştır. Bu sehpa ile kameranın düşey konumu ayarlanarak taşların dağılı olduğu tüm yüzeyi görmesi sağlanmıştır.

Harici aydınlatma kaynağı düzgün dağılımlı bir ortam aydınlatması kullanılmıştır. Bu tip aydınlatma özellikle gölge etkisini ortadan kaldırmak için uygulanır.

Bu uygulamada dijital bir kamera modeli kullandığı için örnekleme ve niceleştirme (sampling and quantization) işlemini direk olarak kamera gercekleştirmektedir. Kullanıcıya sadece sayısal görüntüyü almak kalır ki bu işlem için PC üzerindeki IEEE 1394 arayüzü kullanılmıştır. Kamera ile alınan görüntüler aynı zamanda miniDV bir kasete de yedeklenmiştir. Bu bize görüntüleri daha sonra inceleme olanağı da sunmuştur. Resim 7.1' de kurulan uygulama düzeneği görülmektedir.



Resim 7.1. Kullanılan sistem

Bilgisayar

Görüntü işlemede kullanılan ana donanımlar işlemcilerdir. Bunun için bilgisayarlar veya sayısal sinyal işleme (digital singnal processing) donanımları kullanılır. Bu uygulamada video yakala, görüntü yakalama, görüntü işleme verileri yedekleme işlemleri için AMD Athlon64 işlemci, 512 MB RAM, 80GB HDD ve ATI Rodeon 9600XT 128MB RAM li ekran sahip bir PC kullanılmıştır. Bilgisayar, kamera ile IEEE 1394 portundan bağlandıktan sonra Kameradaki video görüntülerinin almak için Panasonic MotionDV Studio 5.3E LE for DV yazılımı kullanılmıştır. Bu yazılımla MPEG 2 formatında (720X576) görüntü kaydetme yapılabilmektedir. Ayrıca kameranın standart snapshot özelliği yine bu yazılımla kullanılarak (640X480) büyüklüğünde resim yakalama işlemide yine bu yazılımla gerçekleştirilmiştir. İlk olarak gerekli bağlantılar kurulduktan sonra 25 adet kalibrasyon örneği görüntüsü, 25 adet 640*480 24bit görüntü ve 25 adet 720*576 24bitlik görüntüden oluşan bir set HDD üzerine yedeklenmiştir. Görüntü isleme yazılımı, Kamera Kalibrasyon yazılımı, kullanılarak, kameranın niceliksel bilgileri belirlenip, sonrasında taşlar üzerinde tek tek işlem yapılarak tanıma aşamasına geçilmiştir.

CCD Camera

Bu uygulamada kullanılan kameranın iki önemli özelliği bulunmaktadır. Bunlar 3 CCD li olması ve Digital görüntü almasıdır. 3 CCD teknolojisinde ışık bir prizma ile 3 ana renge bölünür (RGB) ve her renk ayrı bir CCD ile toplanır. Renkler orjinaline sadık kalınarak düşük renk değişim seviyelerinde bile çok iyi bir şekilde tekrar oluşturulur. Her CCD' nin bağımsız sinyal işleme özelliği; daha düşük gürültü seviyesi, piksel interpolasyonuyla daha temiz ve yüksek kontrastlı görüntüler elde etme, daha gerçeğe yakın renk ve detay olanakları sağlar. Kullanılan kameranın en önemli özelliği ise görüntünün direk olarak sayısal formatta elde edilmesidir. Böylece görüntü yakalama işlemi için özel bir donanım kullanma (capture card veya frame grabber card) zorunluluğu ortadan kalkmıştır. Görüntüler zaten sayısal formatta olduğundan direk olarak IEEE 1394 arayüzünden PC ortamına aktarılmıştır.

Üzerinde çalışılan görüntüler ise kamera paketiyle gelen Panasonic Motion Studio 5.3E LE yazılımı ile alınmıştır. Bu yazılım kameranın snopshot özelliğini kullanarak direk 640*480 boyutlarında yakalama olanağı sunmaktadır. 720*576 boyutundaki görüntüler ise direk MPEG II formatındaki video görüntüsünden, video player yardımı ile alınmıştır. Kameranın kendine öz (intrinsic) bilgileri ve dışa bağlı (extrinsic) bilgileri 640*480 formatındaki görüntüler aracılığıyla tespit edilmiştir. Kameranın odaklanma özelliği otomatik modda bırakılmıştır. Alınan resim ve video görüntüleri bilgisayar ortamına kaydedilmiştir. Kamera, sabitleme sehpası aracılığıyla domino taşlarının bulunduğu düzleme dik olacak şekilde düşeyden 60 cm yukarıya sabitlenmiş 48*36 cm^2 lik bir alanı net görecek şekilde otomatik focus modunda ve büyütme yapılmadan kullanılmıştır.

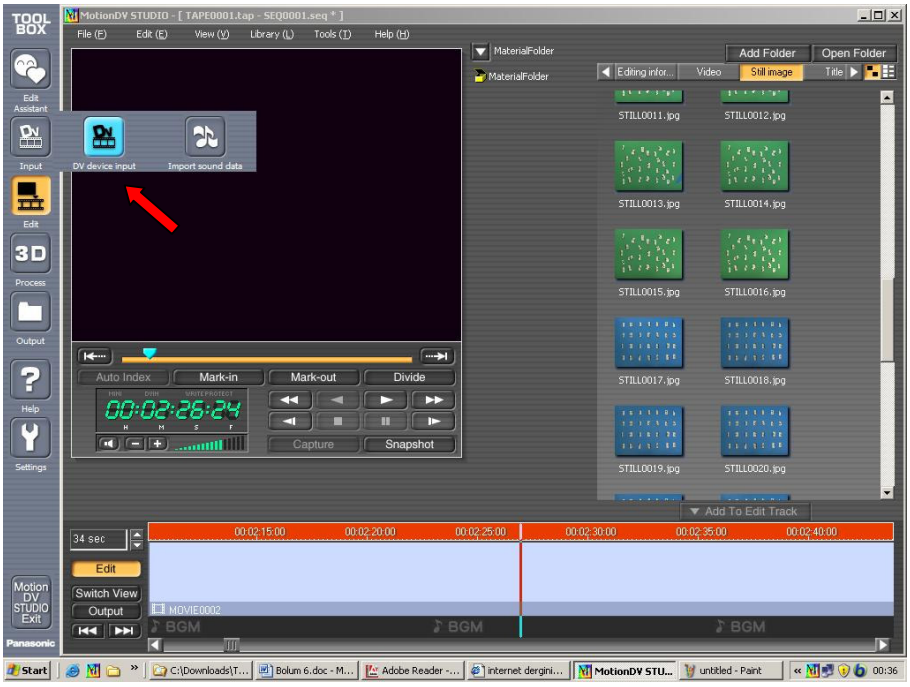
IEEE1394

IEEE1394 dijital video kayıt cihazları, yüksek çözünürlüklü dijital kameralar, sabit disk sürücüleri, DVD-ROM sürücüleri ve yazıcılar gibi birçok elektronik aygıt için popüler bir arayüzdür. Birinci bölüm uygulamada dijital bir kamera kullanıldığı için iletim ortamı olarak bu teknoloji tercih edilmiştir. 1394 arabiriminin özellikleri aşağıdaki gibi özetlenebilir.

- Anında Takma/Çıkarma, PnP: 1394 aygıtları, aygıt takma/çıkarma işlemlerini tespit edebilir ve konfigürasyonunu otomatik olarak yapar.
- Seri Veri Yolu: Veri aktarımı için iki farklı sinyal çifti kullanılmaktadır.
- Eş süreli Aktarım: Kamera gibi video uygulamalarında, sabit bit oranına yakın aktarım gereklidir. Eş süreli (real-time) aktarım, veri kesinliği yerine ayrılmış aktarım oranı sözü verir.
- Direk Aktarım: Aynı 1394 veri yolunda bulunan aygıtların birbirleri ile iletişim kurmaları için sunucu sistemi etkilememe şansları vardır.
- Kablo Gücü: Kendi elektriğini sağlayamayan aygıtlar için, kablo gücü, 1394'ün altı pinlik kablosu aracılığıyla mümkündür.

- Arka Yüzey ve Kablo Ortamları: 1394 sadece harici bağlantılar için değil, dahili PC aygıtları için de tasarlanmıştır.
- Ölçeklenebilir Performans: 1394 aygıtlarının aktarım hızı, azami 100Mbps, 200Mbps veya 400Mbps olabilir.

Bu donanımı kamera ile kullanılırken kamera paketiyle gelen video stream sürücüsü kullanılmıştır. Özelliklerinde de belirtildiği gibi tak çıkar desteği olan bir arayüzdür. IEEE 1394 arayüzününden gelen bilgiyi görüntülemek için Motion studio yazılımından DV device input seçeneğini seçmek yeterli olmuştur. Resim 7.2'de Motion Studio yazılımının arayüzü görülmektedir.



Resim 7.2. Capture yazılımı arayüzü

Görüntü işleme Yazılımı

Görüntü üzerindeki görüntü işleme metotlarını uygulamak için üçüncü parti bir yazılım kullanılmıştır. Üçüncü parti yazılım sayısal hesaplama ortamını kullanan, pek çok fonksiyonun bir araya gelmesinden oluşmaktadır. Buda görüntü işleme

tekniklerinin uygulanmasında son derece büyük kolaylık sağlamaktadır. Görüntü işleme yazılımları aşağıdaki işlemler için kullanılabilir:

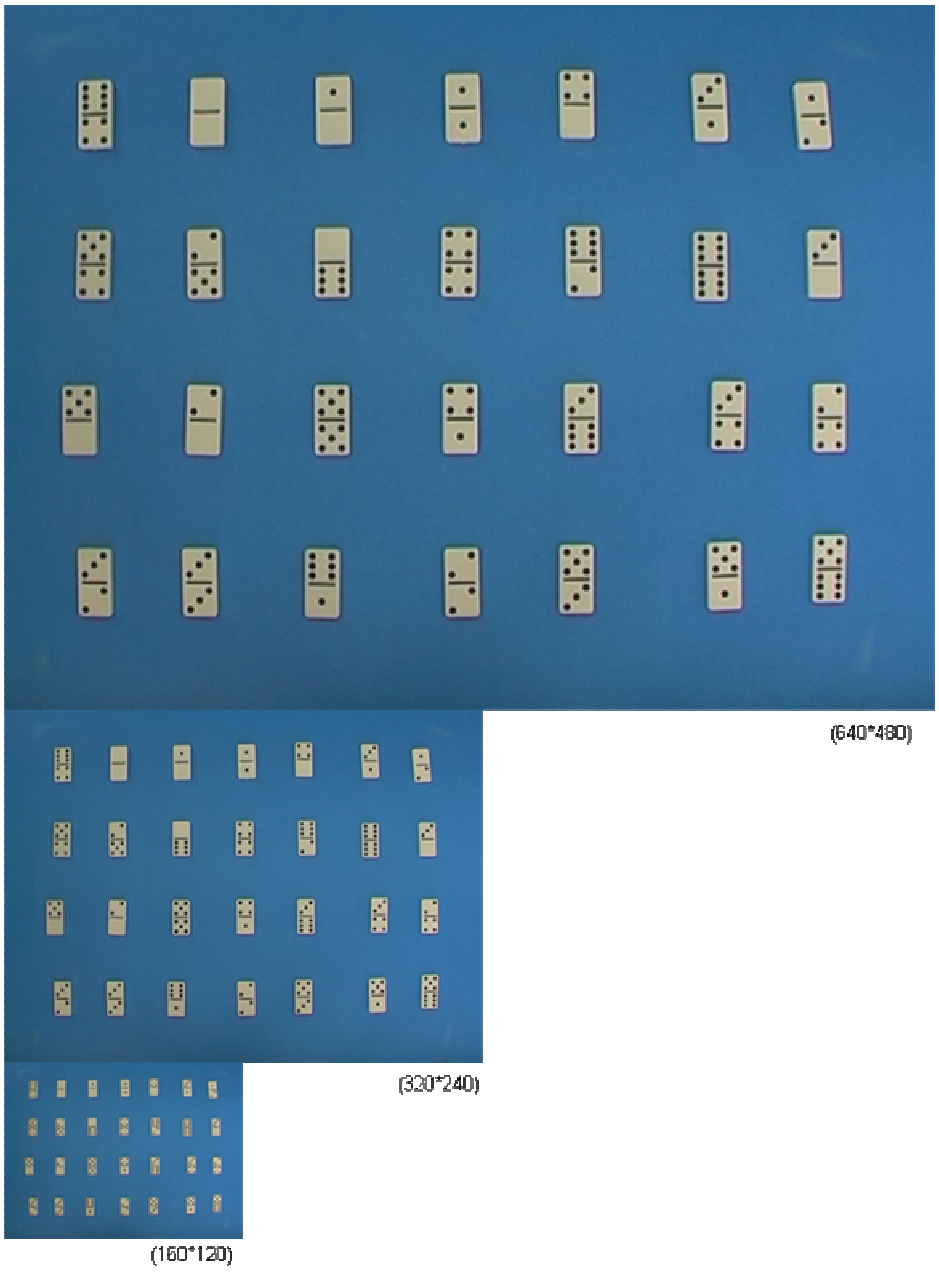
- Piksel tabanlı transformasyon,
- İkili düzey resim işlemleri
- Komşuluk ve bölgesel işlemler
- Lineer Filtreleme ve fitre oluşturma
- Dönüşümler (Fourier Trans., Discrete Cosine Trans.)
- Görüntü geliştirme ve analizi (histogram işleme, kenar çıkarma vb.)
- Resim kontrol kayıtlaması
- Netleştirme (görüntü bulanıklık modeli belirleme, bulanıklık iyileştirme)
- İlgi bölgesi işlemleri (Region of interest)

Hazırlanan uygulamada gri seviye dönüşümleri, ikili düzey işlemleri, kenar belirleme ve özellik çıkarma işlemleri, nesne analizi, ilgi bölgesi seçme işlemleri ve görüntü tanım bölgesi doğrusal filtreleme işlemleri kullanılmıştır. Kullanılan yazılım komutları ve kaynak kodu ekte verilmiştir.

7.1.2. Çözünürlük

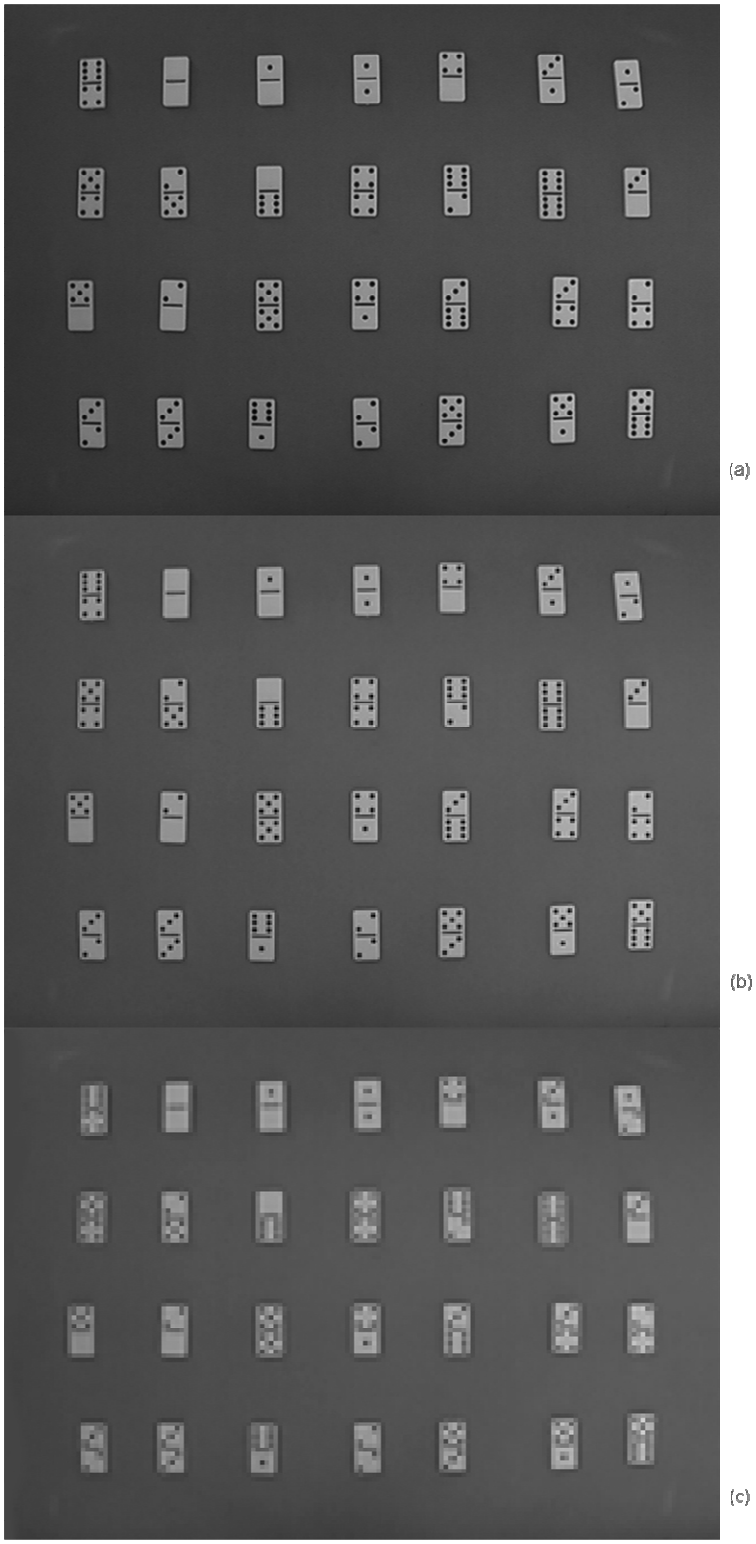
Bir görüntünün çözünürlüğü iki boyutlu piksellerin sayısal büyüklüğünün ifadesidir. Bir resmin çözünürlüğünü belirlemedeki ana faktör örnekleme (sampling) dir . Çözünürlük için en çok kullanılan tanım, resmi oluşturan satır çiftinin (satır ve bitişiğindeki boşluk) en küçük mesafesidir. Renk çözünürlüğünü belirleyen ana faktör ise nicelikleştirme (quantization) işlemidir. Renk seviyesindeki en küçük değişim değerini ifade eder. Genelde renk derinliği 2'nin üstleri şeklinde ifade edilir. Genelde kullanılan renk seviyeleri 2^8=256 renk, 2^{16}=65536 renk ve 2^{24}=16777216 renk dir. Görüntü çözünürlüğü nekadar iyi olursa nesnenin kesinliğide o derece iyi olacaktır. Buda elde edilen sonuçların hassasiyetini etkiler. Aynı şekilde renk seviyeside düşük olursa görüntü içerisinde detaylar yine azalır ve eş düzey eğrileri

belirginleşir. Resim 7.3'de değişik çözünürlük seviyelerine sahip domino görüntüleri görülmektedir.



Resim 7.3. Değişik çözünürlükler

Resim 7.4 ise değişik çözünürlükdeki resimlerin aynı piksel alanında ifadesi görülmektedir.



Resim 7.4. (a) 640*480 lik görüntü,(b) 320*240'lik görüntü, (c) 160*120' lik görüntülerin 640*480 piksel olarak gösterimi

7.1.3. Görüntü yakalama ve görüntü formatı

Bitmap, bilgisayar ekranında (display space) bir resmi görüntülemek için gereken tüm renk bilgilerinin ekranı oluşturan her piksel için verildiği resim formatlarına verilen genel addır. GIF, JPEG, BMP, PCX örnek Bitmap formatlarıdır.

Bitmap formatları, resmin çözünürlüğüne bağlı sabit bir tanımlama ile oluşturulur (Raster tekniği). Bu haliyle, eğer resim ölçeklendirilirse (küçültme/büyütme) bir çok ayrıntı kaybedilebilir. Vektör bazlı grafik formatlarında ise, tanımlamalar bağıl olduğu için ölçeklendirmelerde kayıp önlenir (CAD formatları, Post Script (PS, EPS) formatları gibi).

BMP,GIF,JPEG,TIFF gibi resim formatları arasındaki temel fark sıkıştırma algoritmaları, kodlama gibi özelliklerinden kaynaklanmaktadır.

JPEG: Çok renkle (256 renkten fazla) çalışıldığı zaman GIF formatının yapısından dolayı kullanamaz. BMP olarak saklanırsa, bu durumda çok disk alanı kaplamaktadır. Tiff ise jpeg formatına nazaran daha az sıkıştırma oranına sahiptir. Yüksek seviyede renk düzeyine sahip uygulamalar için saha uygundur. Onun yerine JPEG (Joint Photographics Experts Group) tercih edilebilir. Standart JPG formatında, resmin kalitesinden bir miktar ödün vererek sıkıştırma uygulanır. Böylece dosya boyu bir hayli düşer. Özellikle 24 bit true color uygulamalarda resim kalitesinin düştüğünü anlamak mümkün değildir. JPEG'den ne kadar sıkıştırma istendiği (0-100 arası bir faktör) seçilebilir ama genellikle 5-95 arası kullanılmaktadır.

Birinci bölüm uygulamada çözünürlüğü 640*480 24 bit, ve 720*576 24 bit olan JPEG formatında resimler yakalanmıştır. Bu çözünürlükler sayısal kameranın direk desteklediği değerlerdir.

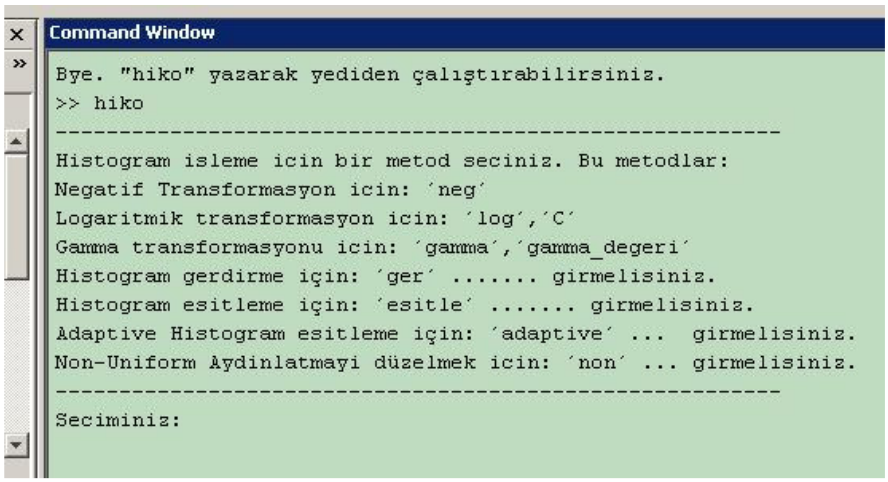
7.1.4. Görüntü işleme

Bu bölümde önceki bölümlerde anlatılan tekniklerin (görüntü iyileştirme geliştirme, görüntü analizi) uygulama sonucu ve nesne tanıma işleminin sonucunu 4 farklı

durum (640*480 ve 720*576 çözünürlüklerinde domino taşlarının düzenli ve düzensiz dağıltılmış durumları) için sunulacaktır.

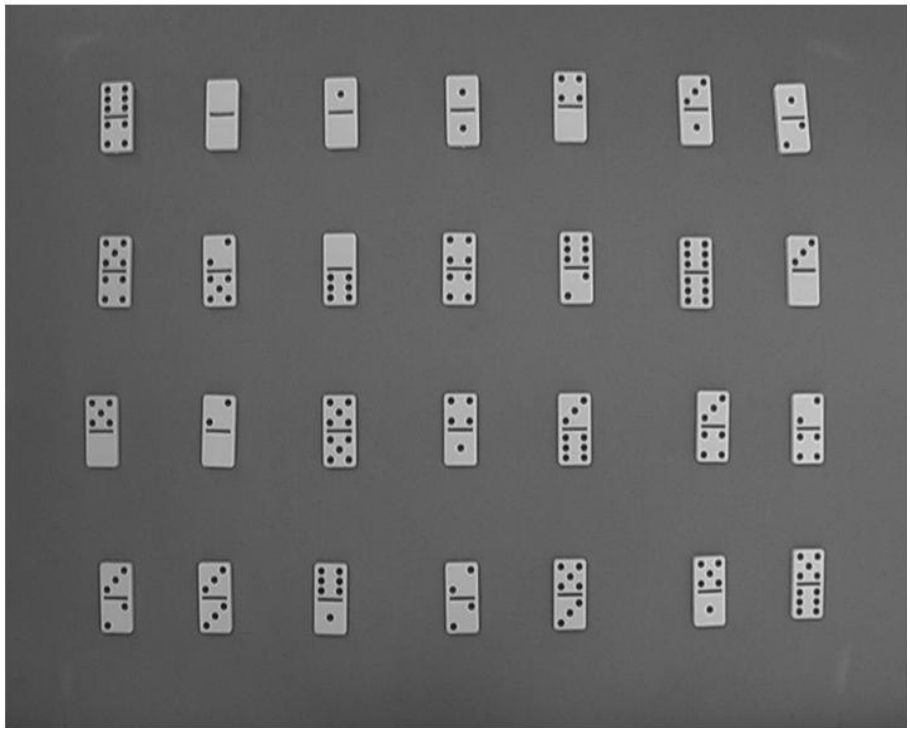
Histogram işleme

Histogram işleme bölüm 3.2 ve 3.3 de anlatılan teknikleri kapsamaktadır. Hazırlanan yazılımda Histogram işleme menüsünde Şekil 7.1 de görünen seçenekler vardır.

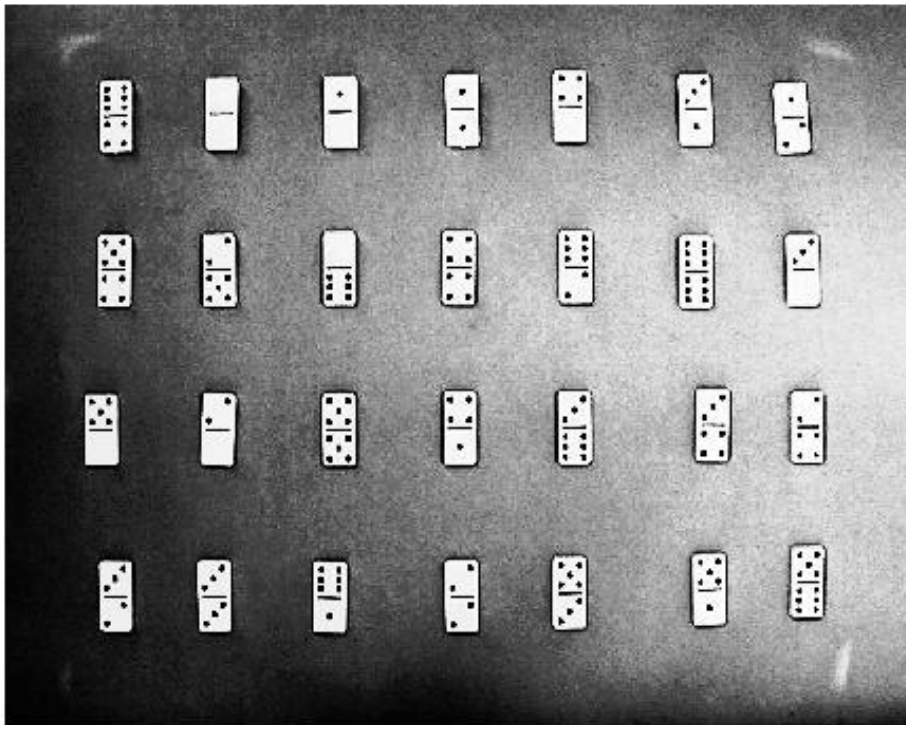


Şekil 7.1. Histogram işleme menü seçenekleri

Bu bölümde elde edilen sonuç bir sonraki menü seçeneği için girdi olmaktadır. Buradaki seçenek şu şekilde kullanılabilir. Domino taşları olarak beyaz taş üzerine siyah noktaları olan bir model kullanılmıştır. Dolayısıyla görüntüdeki siyah bölgeleri beyaz, beyazları da siyah göstermek için negatif dönüşüm uygulanabilir. Bu dönüşüm fark edilmeyen ayrıntıları daha belirginleştirmek için kullanılmaktadır. Logaritmik ve üssel dönüşümler Eş. 3.4 ve denklem Eş. 3.6 ile elde edilmiştir. Burada C ve gamma değerleri kullanıcı tarafından belirlenmezse sırasıyla yerlerine 1 ve 0,5 değerleri kullanılmıştır. Histogram germe Eş. 3.5 ile verilmiştir. Bu seçenekle kontrast artırılmaktadır. Histogram eşitleme işleminde bölüm 3.4 de anlatıldığı gibi çıkışta tüm gri seviye bandını kaplayan bir resim elde edilmekteydi. Bu seçeneği Resim 7.5(a)'daki görüntü için uygulandığında Resim 7.5'deki sonuç elde edilmiştir. Orjinal ve işlenmiş görüntünün histogramları Şekil 7.6'da görülmektedir.

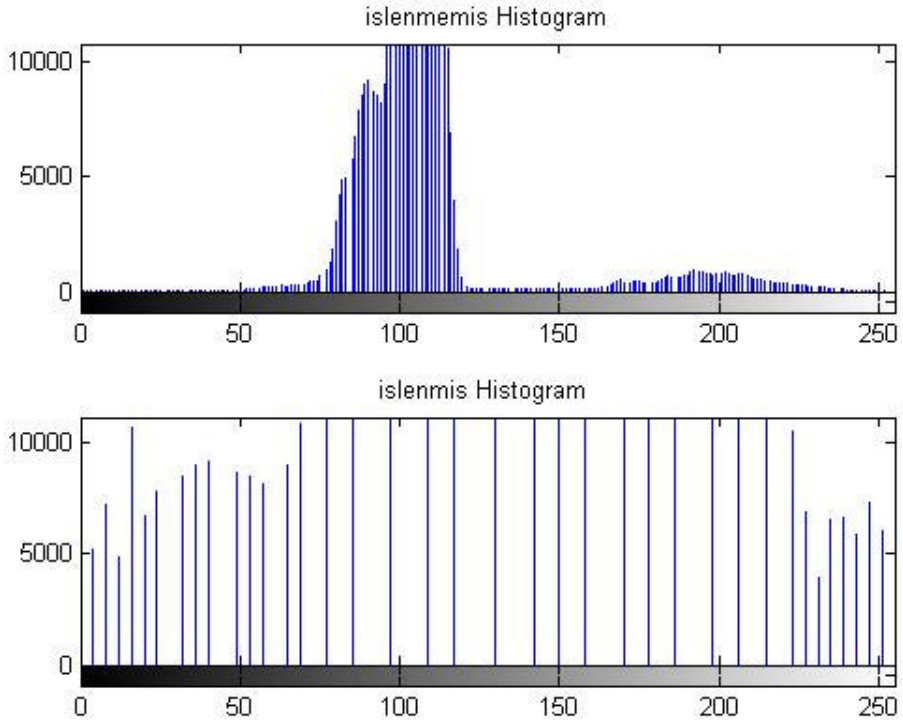


(a)



(b)

Resim 7.5. (a) Orjinal resim , (b) Histogram işlenmiş resim



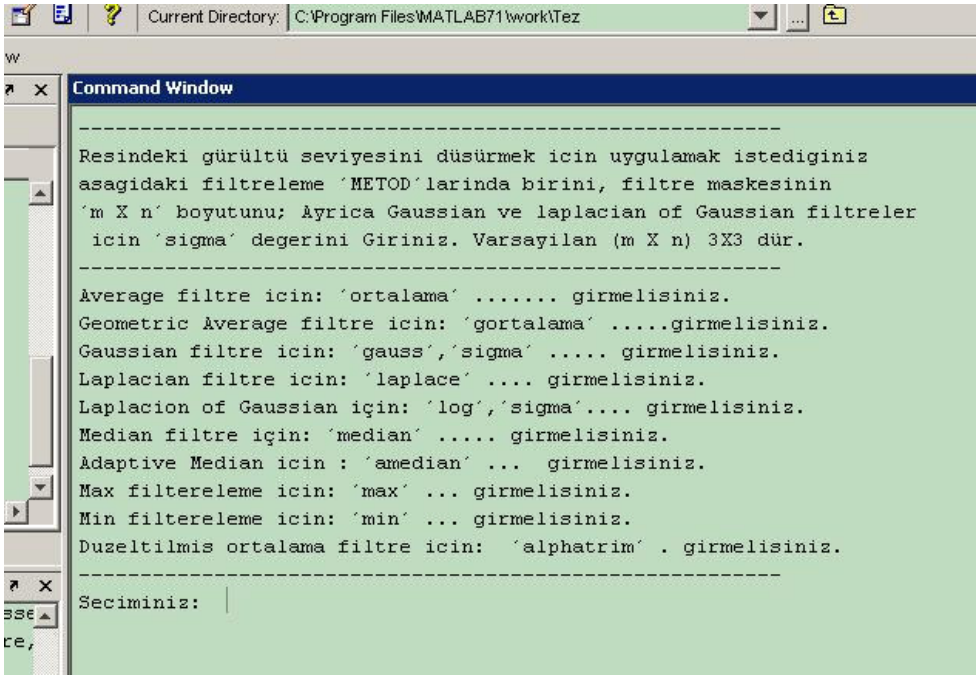
Şekil 7.2. Orjinal görüntüye ait histogram ve eşitlenmiş histogram

Eldeki görüntülerde eğer düzensiz bir aydınlatma problemi varsa Non-Uniform Aydınlatmayı düzeltme seçeneği, görüntüye filtreleme işlemi uygulamadan önce daha iyi sonuçlar vermektedir. Birinci bölüm uygulamada kullanılan görüntülerde düzensiz aydınlatma gözlenmiştir. Histogram eşitlenmiş görüntüde bu düzensiz aydınlatma bariz şekilde belirgindir. Işık kaynağının olduğu taraftan diğer uça doğru görüntü içerisindeki gri seviye değerleri değişim göstermiştir.

Filtreleme

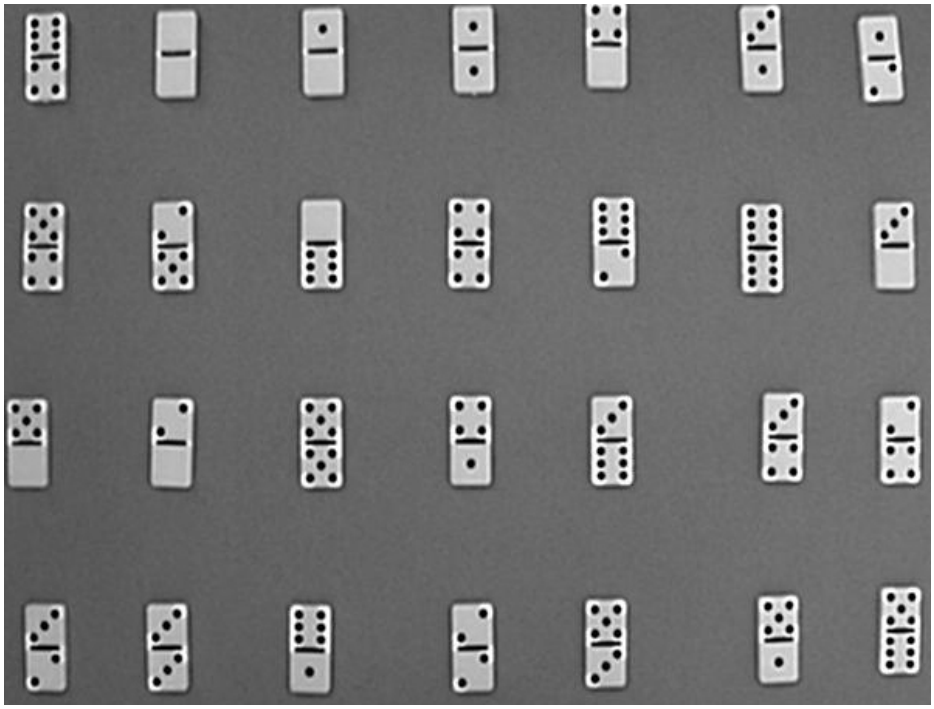
Bölüm 3.4'de de belirtildiği gibi görüntü filtreleme teknikleri görüntüdeki gürültü ve bozuklukları bastırmaktır. Filtreleme için görüntü tanım bölgesinde uygulanan lineer filtreler kullanılmıştır. Hazırlan yazılımın filtre menusu Şekil 7.3'de görülmektedir. Keskinleştirme ve yumuşatma filtrelerinden birini kullanarak görüntü, gürültülerden arındırabilir. Gaussian ve Laplacian filtre için sigma (standart sapma) ve alpha

(maske biçimi) standart olarak 0.5 dir. Filtreleme için oluşturulan maske boyutu için varsayılan değer ise 3x3 dür.

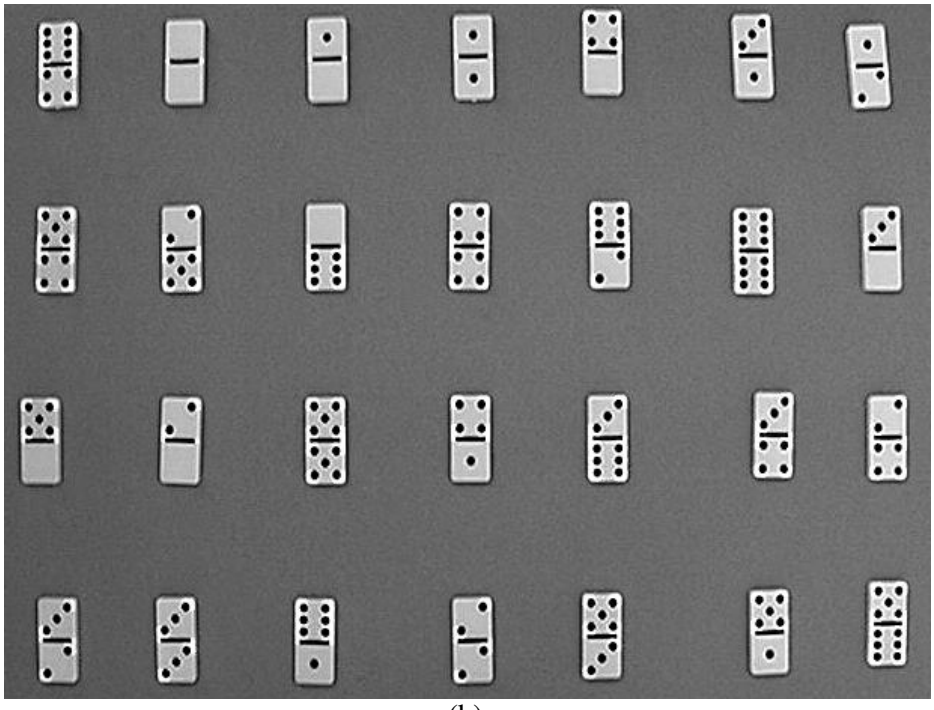


Şekil 7.3 Filtre menüsü

Burada max, min, adaptive median filtreler görüntü tanım bölgesinde uygulanan non-lineer filtrelerdir. Yumuşatma filtreleri eşikleme işleminde daha başarılı sonuçlar vermiştir. Resim 7.5 (a)'daki görüntüye ait ortalama filtreleme sonucu Resim 7.6 (a)'daki gibidir. Ortalama filtre resim içerisindeki keskin geçişleri yumuşatmaktadır. Bu daha bulanık bir görüntü oluşmasına neden olur. Ayrıca nesne sınırları da bu duruma bağlı olarak biraz genişleme gösterir. Resim 7.6(b) ve Resim 7.7 (a)'de 5*5 boyutlarında 0.2 ve 2 sigma değerleri için Gaussian filtre sonuçu görülmektedir. Yüksek sigma değerleri için daha bulanık bir sonuç elde edilirken düşük sigma değerlerinde sınırlar daha belirgindir. Laplacian filtre ile Resim 7.7 (b)'de görüldüğü gibi daha keskin bir görüntü elde edilmiştir. Log (laplacian of gaussian) filtre ise kenar belirleme için kullanıllır. Bu durum Resim 7.8 (a)'de açık bir şekilde belirgindir. Median filtrenin sonucu ise Resim 7.8 (b)'de dir. Filtrelenmiş resim bir sonraki aşamada bölümledirme için giriş görüntüsü olarak kullanılmaktadır.

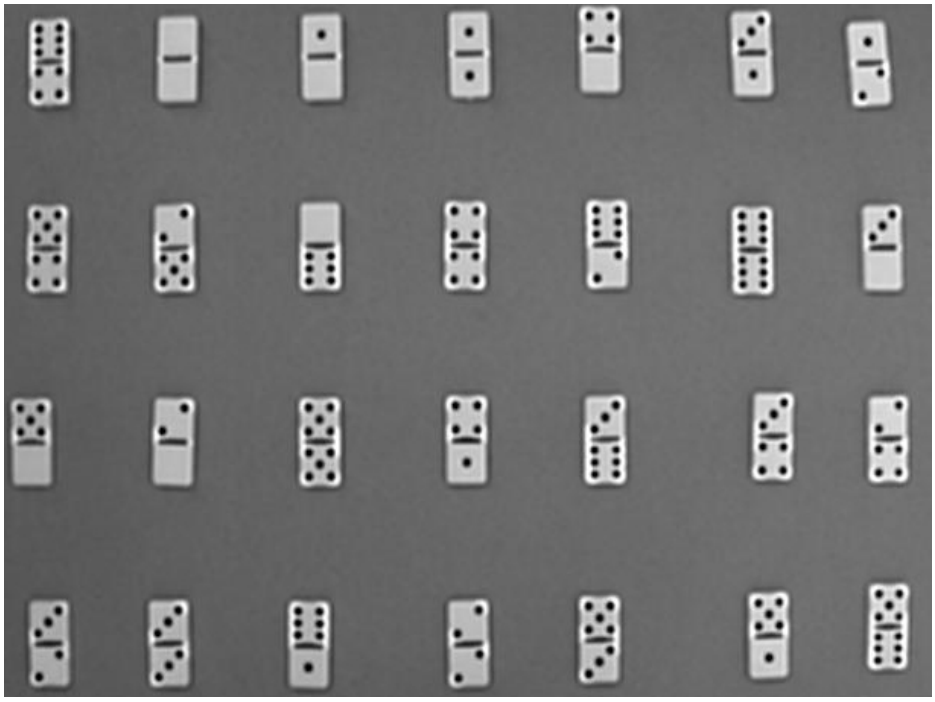


(a)

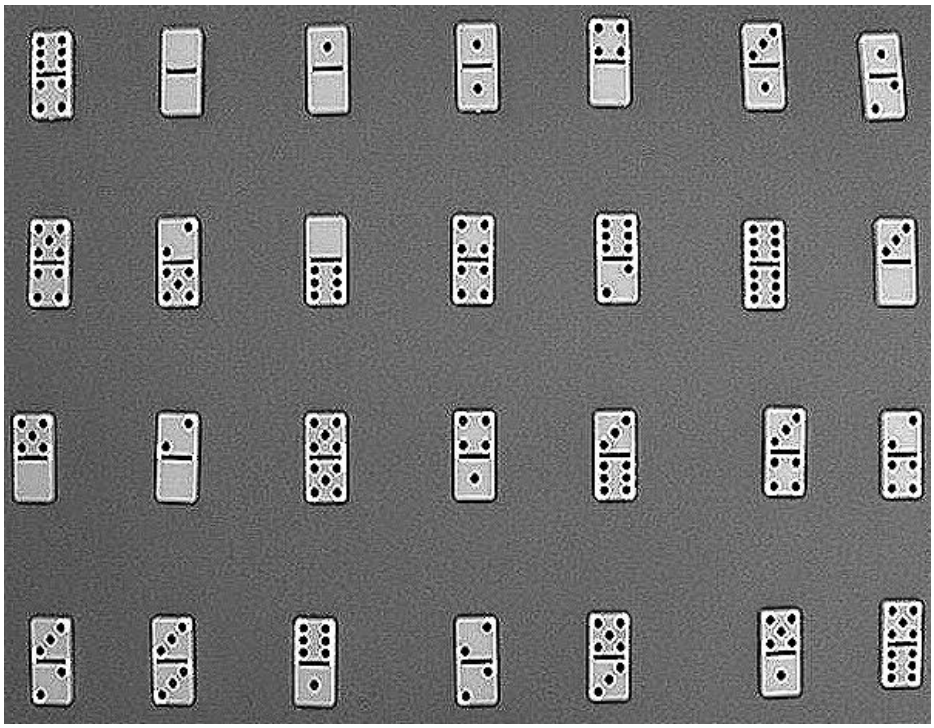


(b)

Resim 7.6. (a) Ortalama filtre uygulanmış görüntü (b) 5*5 boyutlarında sigma=0.2 için Gaussian filtrelemenin sonucu

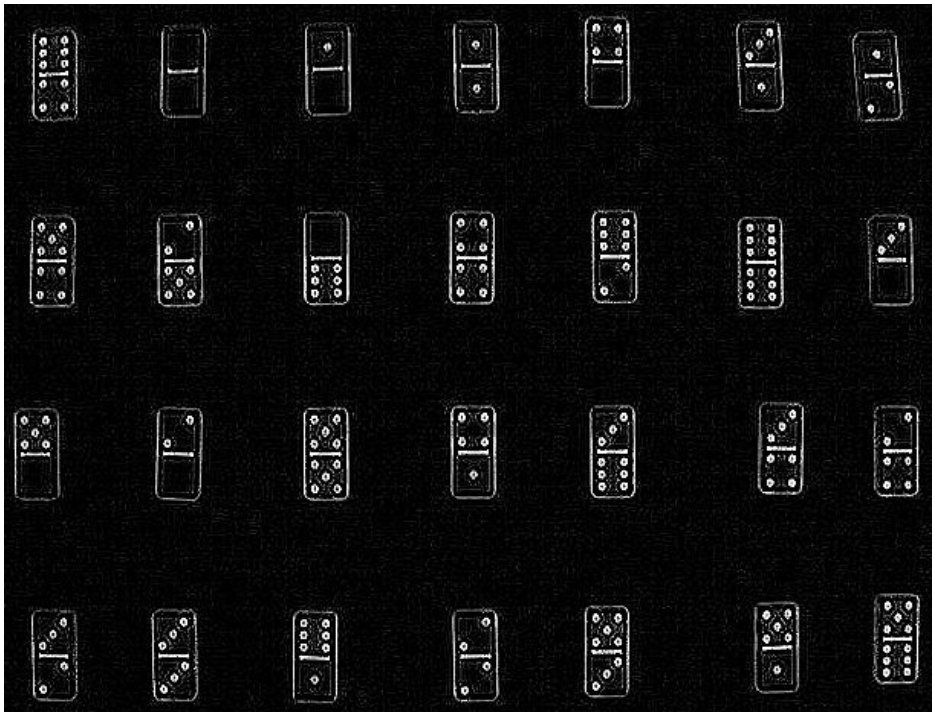


(a)

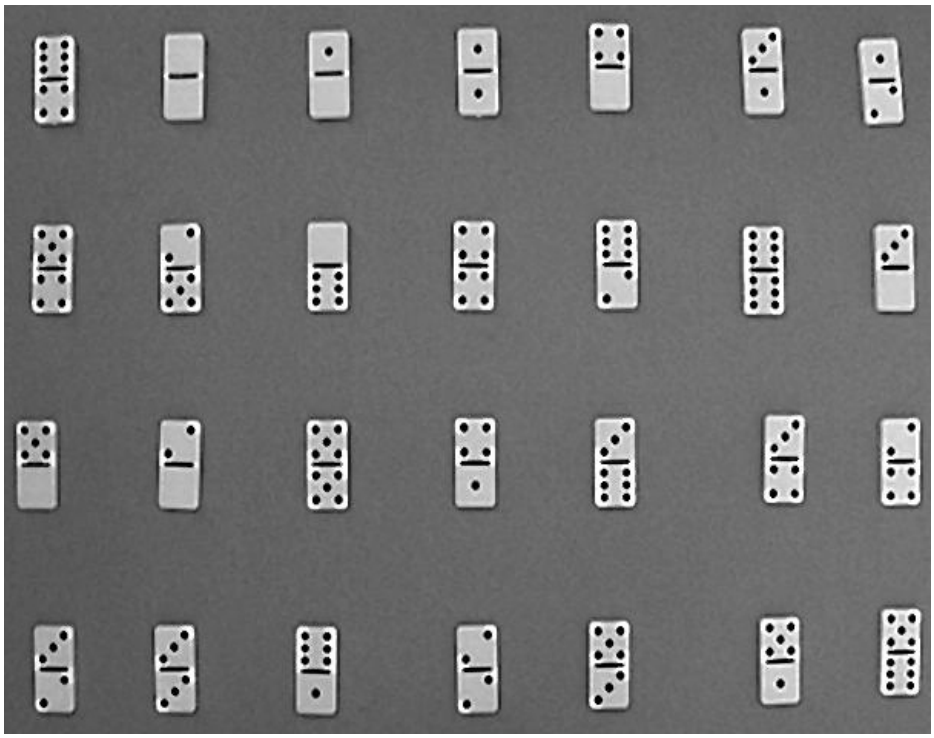


(b)

Resim 7. 7. 5*5 boyutlarında sigma=2 için orijinal görüntünün Gaussian filtreleme sonucu (b) Laplacian filtrelenmiş görüntü (alpha=0)



(a)

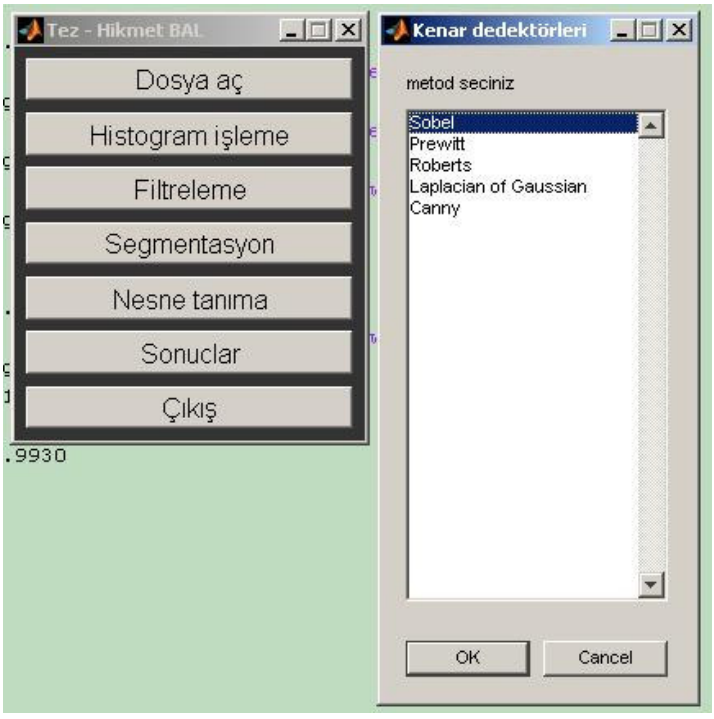


(b)

Resim 7. 8 (a) Log (laplacian of gaussian) filtrelenmiş görüntü (b) Median filtre sonuçu

Bölümlendirme (Segmentasyon)

Bölümlendirme işlemi Bölüm 5' de anlatıldığı gibi nesne ile geri planı ayırmak ve görüntüden nesneleri seçmek için kullanılmaktadır. Bu uygulamada kenar belirleme işlemleri için Bölüm 5.1 ve 5.2 de anlatılan kenar birleştirme sınır belirleme, Hough dönüşümü algoritmaları kullanılmıştır. Eşikleme için ise genel eşikleme metodu uygulanmıştır. Şekil 7.4'de Bölümlendirme menü içeriği görülmektedir. Burada görüldüğü gibi kenar belirleme metotlarından biri uygulandıktan sonra resim otomatik olarak eşikleme yapılmakta ve sonuçta ikili bir görüntü elde edilmektedir. Kenar belirleme işlemiyle elde edilen görüntü taşların eğimini bulmak için kullanılmıştır. Taşların dış sınırlarının iyi bir şekilde elde edilmesi, hough dönüşümü ile taşların eğimlerin belirlenmesinde önemli etki yapmaktadır.



Şekil 7.4. Bölümlendirme menusu

Kenar belirleme birleştirme operatörlerinden "Sobel" operatörü resmin birinci türevini alarak gradient vektörün genliğini kullanmaktadır. Aynı zamanda görüntüye eşiklemede uygulayarak kenarların belirgin bir şekilde görünmesini sağlarlar. Prewitt

ve Roberts operatörleride benzer işlem uygulamaktadır. Eşikleme için tek değer kullanmaktadırlar. LoG işlemi için Bölüm 3.4.3'de anlatılılan teknik uygulanmıştır. LoG işleminde eşikleme yapılmadığı takdirde filtreleme benzeri sonuç elde edilir. LoG operatöründe nesne bölgelerinin sınırları daha büyük görünür. Canny operatörü güçlü bir kenar dedektörüdür. Bu metot şu şekilde uygulanmaktadır;

- Görüntü, tanımlana sigma değerine göre gaussian filtre kullanılarak önce yumuşatılır.
- Her noktada gradient vektörün genliği ve doğrultusu belirlenir. Bir kenar noktası, gradient vektör yönünde güçü bölgesel maksimum olan noktadır.
- Bir önceki aşamada belirlenen kenar noktaları bir sırt oluşturur. Bu sırt üzerindeki en yüksek değer (tepede olanlar) kenar noktası olarak belirlenir ve diğerlerine sıfır atanır. Böylece çıkışta ince bir çizgi elde edilir. Bu işlem maksimum olmayan değerlerin bastırımı 'nonmaximal suppression' olarak bilinir. T1<T2 olacak şekilde eşikleme uygulanır.
- Son olarak 8'li komşuluk ilişkisi kullanılarak kenar pikselleri birleştirilir.

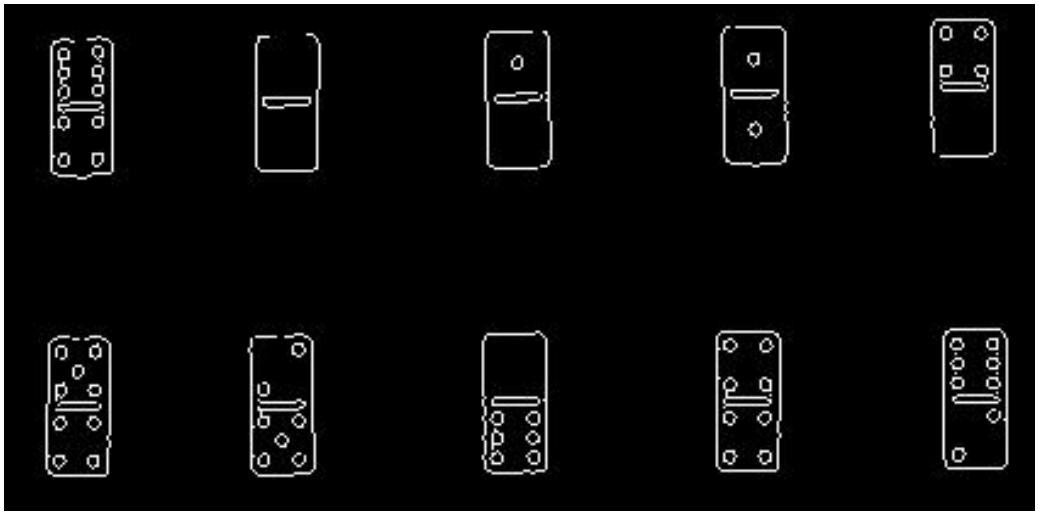
Canny operatörünün diğer bir üstünlüğü iki eşik değeri kullanmasıdır. Bu çalışmada en çok bu metot tercih edilmiştir. Bölümlendirmede kullanılan eşik değerleri Çizelge 7.1 deki gibidir. Filtrelenmiş görüntü 5*5 boyutlu σ=1 değerli Gaussian fitle uygulanarak elde edilmiştir.

Çizelge 7.1 Orjinal görüntü ve filtrelenmiş görüntü için eşik değerleri

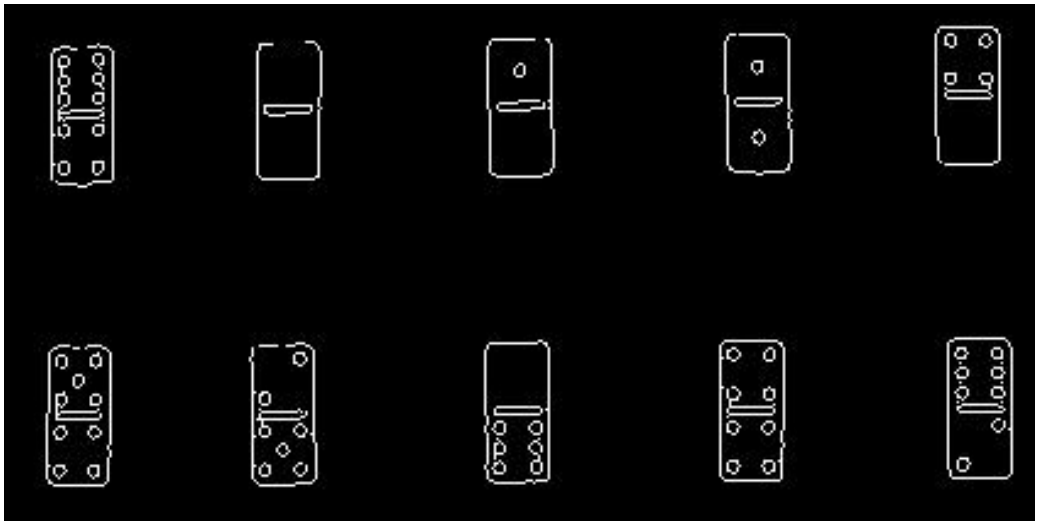
	Eşikleme Metodu	Eşik değeri
Orjinal Görüntü	Genel Eşikleme	135
	İterative eşikleme	173
Filtrelenmiş Görüntü	Genel Eşikleme	151
	İterative eşikleme	198

Resim 7.9(a),(b) ve Resim 7.10 (a),(b),(c) de kenarları sırasıyla sobel, prewitt, roberts, log, canny operatörleri ile belirlenmiş görüntüler görülmektedir. Kenar

belirleme için Resim 7.8(b)'de Median filtre uygulanmış görüntü kullanılmıştır. Resim 7.11(a)'de ise genel eşikleme sonrası elde edilen nesne sınırları ile iç sınırlar gösterilmektedir. Kırmızı renkle çizilen yerler dış sınırları, yeşil renkle çizilen yerler ise iç sınırları ifade etmektedir. Resim 7.11 (b)'de ise genel eşikleme ile elde edilmiş ikili görüntü görülmektedir.

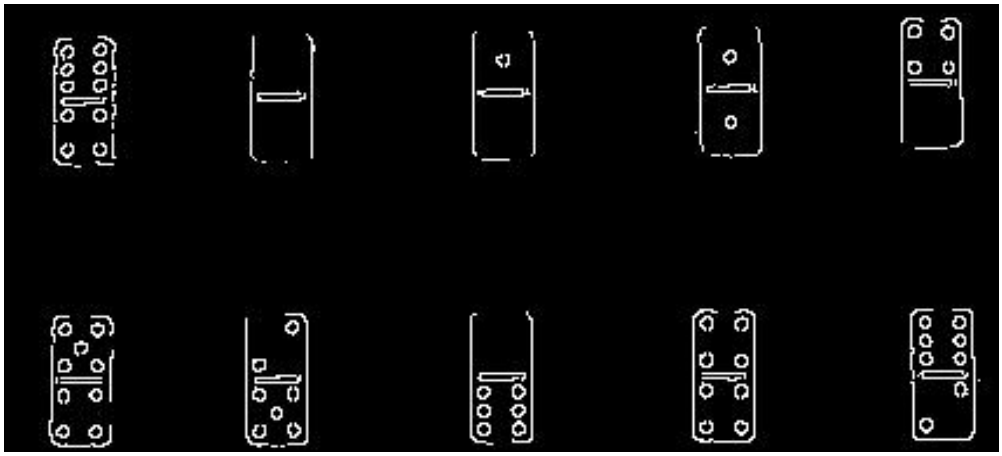


(a)

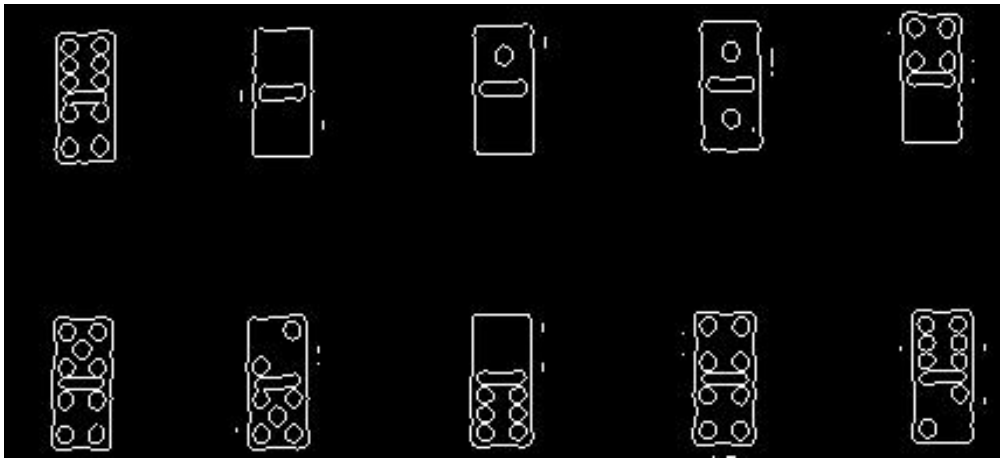


(b)

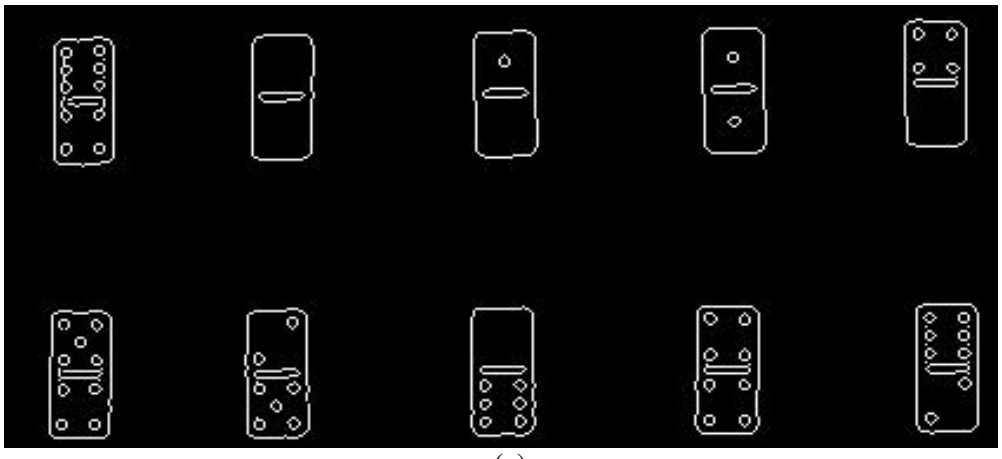
Resim 7.9. (a) Sobel operatörü ile elde edilmiş kenarlar. (b) Prewitt operatörüyle elde edilmiş kenarlar.



(a)

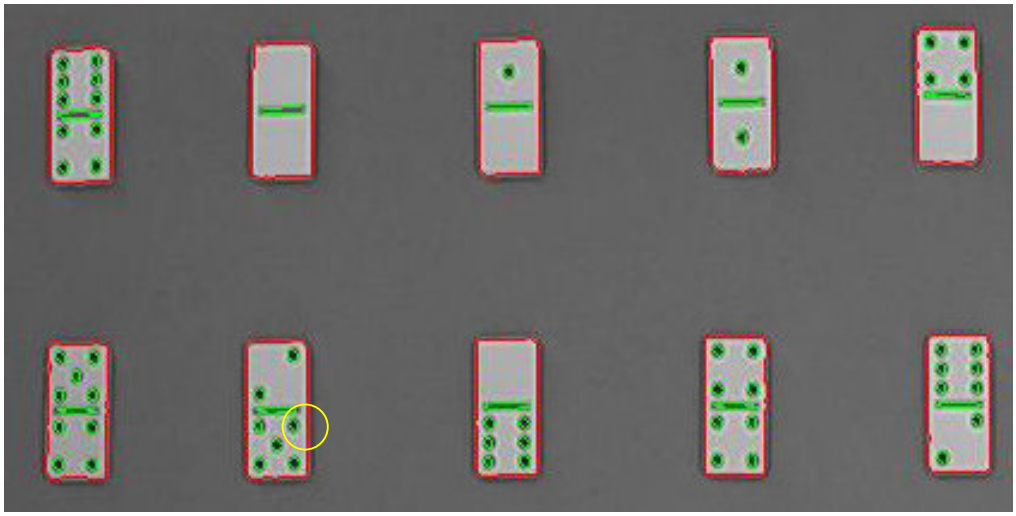


(b)

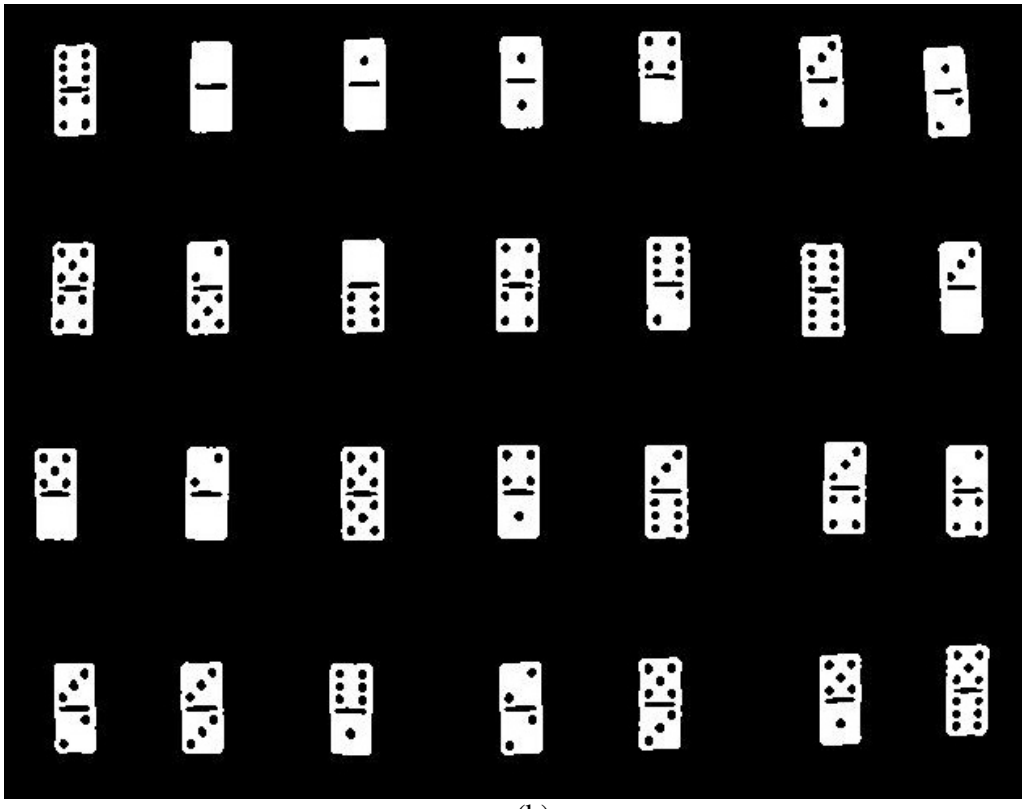


(c)

Resim 7.10. (a) Roberts operatörü ile, (b) LoG operatörü ile, (c) Canny operatörü ile elde edilmiş kenarlar



(a)



(b)

Resim 7.11. (a) Genel eşikleme ile nesne sınırları (kırmızı) ve iç sınırlar (yeşil). (b) Genel eşikleme ile elde edilmiş ikili görüntü

Nesne tanıma

Bu bölümde, şu ana kadar görüntü üzerinde uygulanılan geliştirme, bölümlendirme işleminden sonra elde edilen görüntüden, elde edilen özellikler ve bu özelliklere göre nesnenin tanıma sonuçları sunulmuştur. Kolaylık sağladığı için nesnenin yapısal özelliklerinden faydalanılmıştır. Bu yapısal özelliklerin en belirgini tüm taşların ve noktaların resim içerisinde aynı (birbirine çok yakın) piksel alanı ile ifade edilmiş olmalarıdır. Nesne tanıma için aşağıdaki işlem basamakları uygulanmıştır.

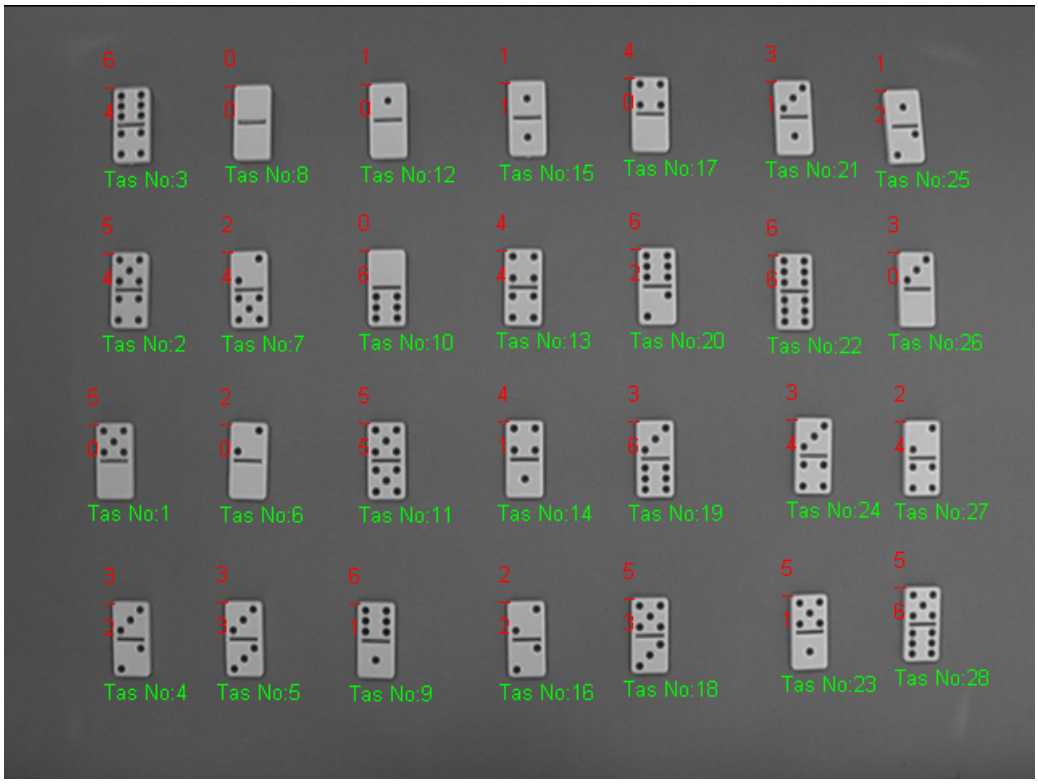
- İşlenmiş görüntüyü al (iyileştirilmiş filtrelenmiş bölümlendirilmiş).
- İkili düzey görüntüyü etiketle (Labelling).
- Etiketlenmiş görüntüden Taş sayısını çıkar.
- Herbir taşın kenarlarını hough dönüşümü ile belirle.
- Taş kenarlarının eğriliğine göre düzeltme açılarını bul.
- İkili resimden herbir taşı al ve düzeltme açısına göre dik konuma getir.
- Taş içindeki noktaların alanlarını sentroidlerini belirle.
- Sentroid ve alan kriterlerine göre alttaki ve üstteki nokta sayısını belirle.
- Elde edilen sonuçları kaydet.

Elde edilen değerler ikili formatta (binary)bir dosyaya kaydedildiğinden sonuçlar daha sonra tekrar incelenebilmektedir. Yukarıda verilen işlem basamaklarının dört farklı seçenek için sonuçları ilgili çizelgede sunulmuştur. Çizelgede taşların elde edilen değeri, taşların eğim açıları (dikey eksene göre) ve koordinat bilgileri bulunmaktadır. Koordinat olarak taşın sentroidi verilmiştir. Bu dört seçenek şunlardır:

- 640*480 çözünürlük, Dominoların dizilişi düz
- 640*480 çözünürlük, Dominoların dizilişi dağınık
- 720*576 çözünürlük, Dominoların dizilişi düz
- 720*576 çözünürlük, Dominoların dizilişi dağınık

Tanıma işlemi sonrasında orjinal gri görüntüde, taş numaraları taşın alt ortasına yeşil renk ile, taş değerleri taşın sol üst köşesine kırmızı renk ile etiketlenmiştir.

Durum 1: Görüntü 640*480 çözünürlükte ve domino taşları düzenli yerleştirilmiştir. Resim 7.12 de domino taşlarının uygulama yazılımı tarafından etiketlenmiş görüntüsü ve Çizelge 7.2 de elde edilen değerler görülmektedir.

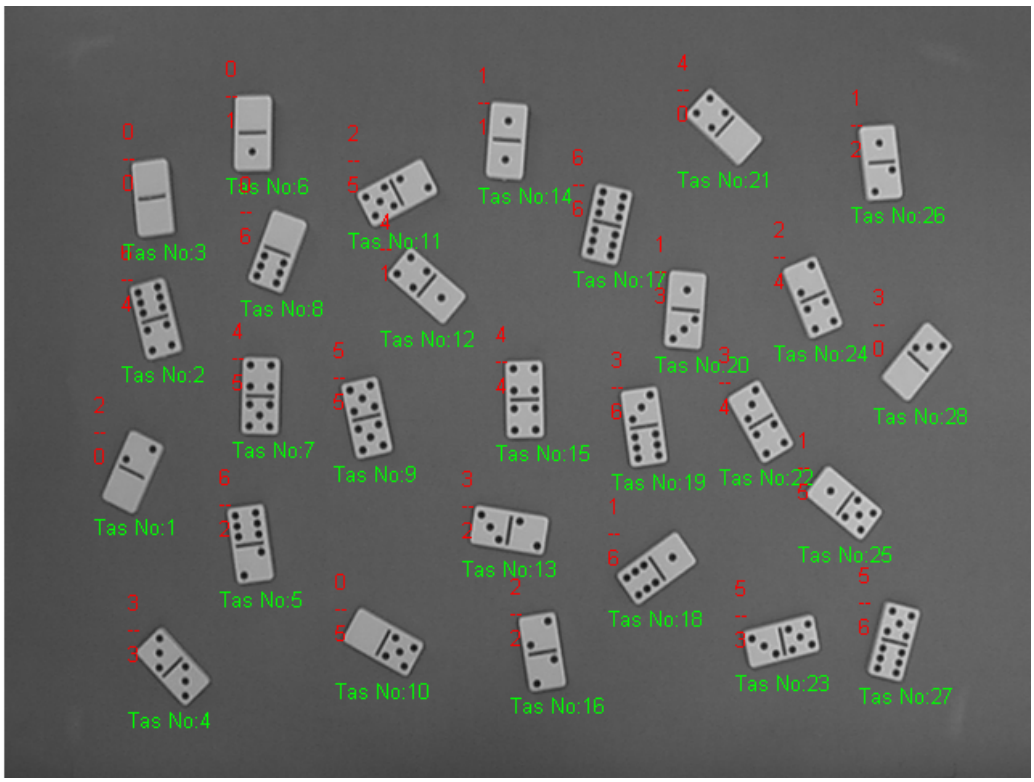


Resim 7.12. Durum1 için elde edilen sonuç

Çizelge 7.2. Durum 1 için elde edilen değerler

	Taş Açısı (◦)	Taş Merkezi koordinatı (piksel)	Ölçülen Taş Değeri
Domino 0/0	1	(154, 173)	0/0
Domino 0/1	-0,5	(238, 71)	1/0
Domino 0/2	0	(152, 283)	2/0
Domino 0/3	2	(566,176)	3/0
Domino 0/4	-1	(401, 68)	4/0
Domino 0/5	-1	(69, 282)	5/0
Domino 0/6	1	(237, 176)	0/6
Domino 1/1	-0,5	(324, 70)	1/1
Domino 1/2	-5	(558,75)	1/2
Domino 1/3	-3	(489, 70)	3/1
Domino 1/4	0	(323, 281)	4/1
Domino 1/5	0	(499, 388)	5/1
Domino 1/6	-0,5	(231, 393)	6/1
Domino 2/2	-2	(324, 393)	2/2
Domino 2/3	-2	(79, 393)	3/2
Domino 2/4	0	(569, 281)	2/4
Domino 2/5	0	(152, 176)	2/4
Domino 2/6	0	(405, 174)	6/2
Domino 3/3	-2	(149, 393)	3/3
Domino 3/4	0	(502, 280)	3/4
Domino 3/5	0	(400, 391)	5/3
Domino 3/6	0	(404, 282)	3/6
Domino 4/4	0	(322, 175)	4/4
Domino 4/5	1,5	(78, 177)	5/4
Domino 4/6	0	(79, 75)	6/4
Domino 5/5	1	(237, 232)	5/5
Domino 5/6	1,5	(569, 384)	5/6
Domino 6/6	0	(490, 178)	6/6

Durum 2: Görüntü 640*480 çözünürlükte ve domino taşları rasgele bir şekilde yerleştirilmiştir. Resim 7.13 de domino taşlarının uygulama yazılımı tarafından etiketlenmiş görüntüsü ve Çizelge 7.3 de elde edilen değerler görülmektedir.

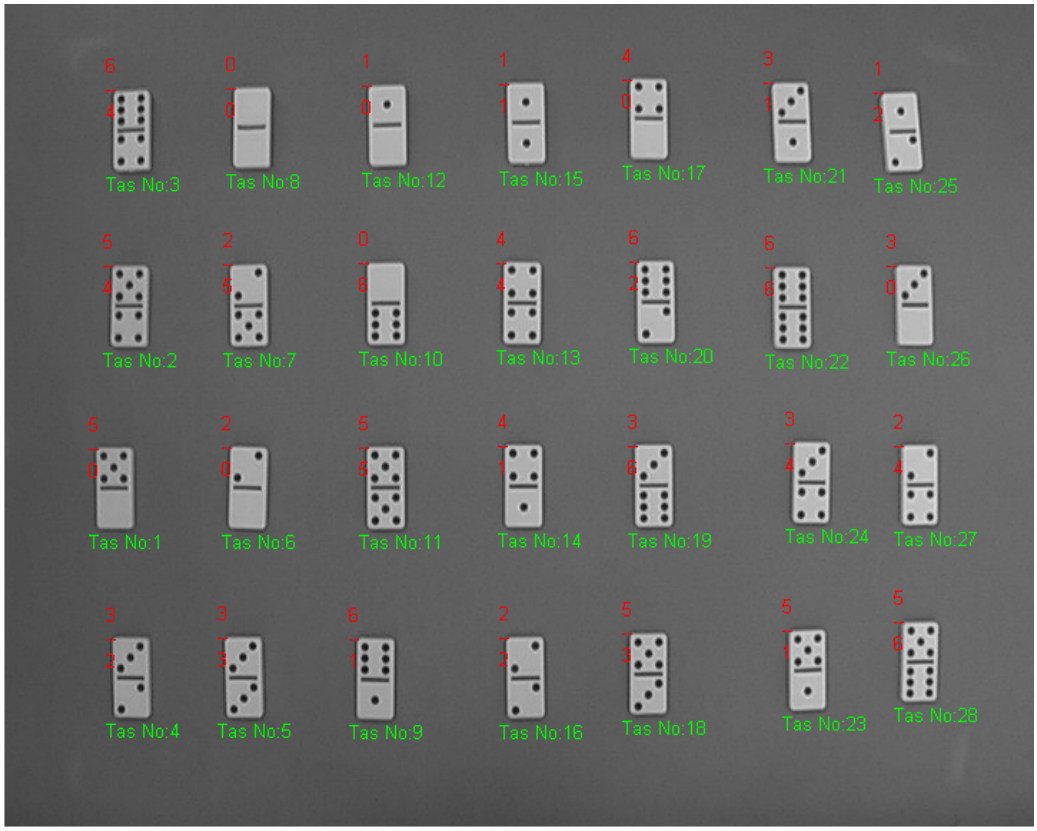


Resim 7.13. Durum 2 için elde edilen sonuç

Çizelge 7.3. Durum 2 için elde edilen değerler

	Taş Açısı (◦)	Taş Merkezi koordinatı (piksel)	Ölçülen Taş Değeri
Domino 0/0	-5,5	(92, 119)	0/0
Domino 0/1	-1	(155, 79)	1/0
Domino 0/2	23,5	(80, 289)	2/0
Domino 0/3	38,5	(568, 221)	3/0
Domino 0/4	-47	(448, 75)	4/0
Domino 0/5	-60	(236, 396)	5/0
Domino 0/6	18,5	(170, 153)	0/6
Domino 1/1	4	(313, 84)	1/1
Domino 1/2	-6	(546, 97)	1/2
Domino 1/3	4	(424, 189)	1/3
Domino 1/4	-49	(263, 174)	4/1
Domino 1/5	-49,5	(523, 309)	1/5
Domino 1/6	55,5	(406, 350)	1/6
Domino 2/2	-9,5	(335, 401)	2/2
Domino 2/3	-79,5	(314, 326)	3/2
Domino 2/4	-22,5	(503, 181)	2/4
Domino 2/5	62	(245, 116)	2/5
Domino 2/6	-8,5	(153, 334)	6/2
Domino 3/3	-41,5	(105, 411)	3/3
Domino 3/4	-28	(471, 258)	3/4
Domino 3/5	75,5	(484, 395)	5/3
Domino 3/6	-8,5	(398, 262)	3/6
Domino 4/4	0	(324, 245)	4/4
Domino 4/5	0	(160, 242)	4/5
Domino 4/6	-13,5	(94, 195)	6/4
Domino 5/5	-12	(226, 256)	5/5
Domino 5/6	15,5	(554, 395)	5/6
Domino 6/6	11,5	(375, 136)	6/6

Durum 3: Görüntü 720*576 çözünürlükte ve domino taşları düzenli yerleştirilmiştir. Resim 7.14 de domino taşlarının uygulama yazılımı tarafından etiketlenmiş görüntüsü ve Çizelge 7.4 de elde edilen değerler görülmektedir.

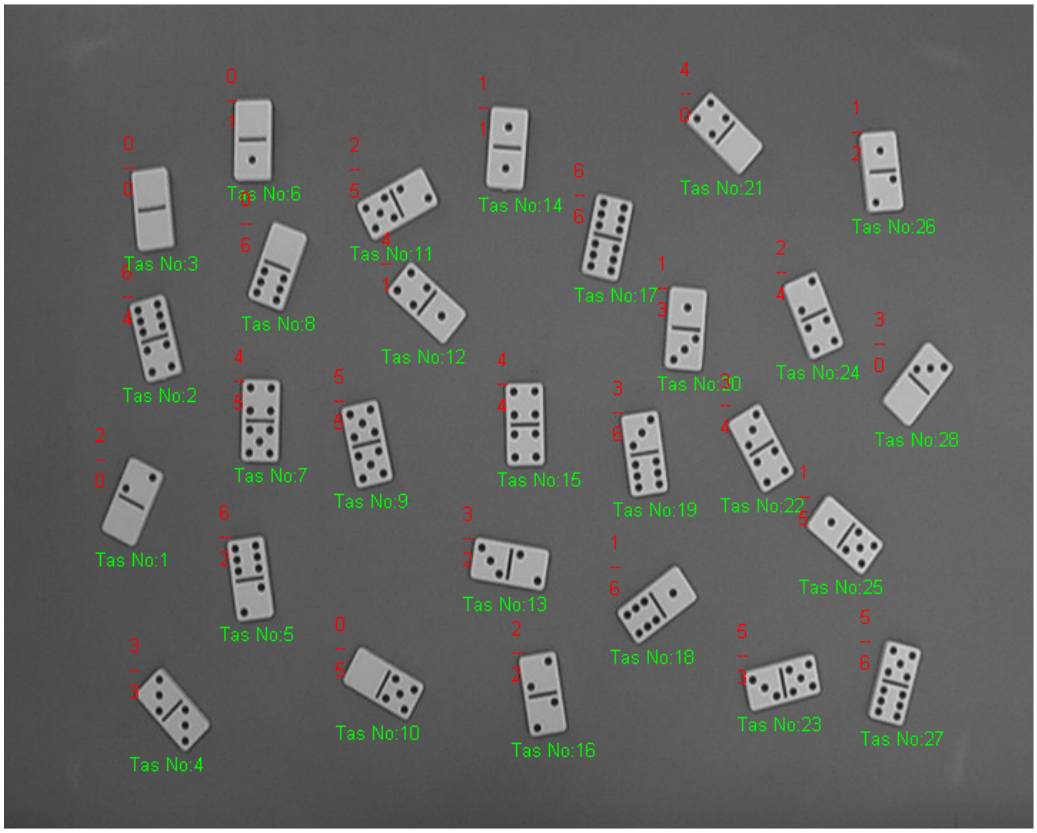


Resim 7.14. Durum 3 için elde edilen sonuç

Çizelge 7.4. Durum 3 için elde edilen değerler

	Taş Açısı (◦)	Taş Merkezi koordinatı (piksel)	Ölçülen Taş Değeri
Domino 0/0	-1	(173, 87)	0/0
Domino 0/1	0	(268, 86)	1/0
Domino 0/2	1,5	(170, 339)	2/0
Domino 0/3	0	(636, 211)	3/0
Domino 0/4	-1	(451, 81)	4/0
Domino 0/5	-1	(77, 339)	5/0
Domino 0/6	0	(267, 211)	0/6
Domino 1/1	-0,5	(365, 84)	1/1
Domino 1/2	-4,5	(627, 91)	1/2
Domino 1/3	-2	(550, 84)	3/1
Domino 1/4	-1	(363, 338)	4/1
Domino 1/5	0	(552, 466)	5/1
Domino 1/6	1,5	(260, 472)	6/1
Domino 2/2	-2	(365, 472)	2/2
Domino 2/3	-2	(89, 472)	3/2
Domino 2/4	-1	(640, 337)	2/4
Domino 2/5	-1	(171, 212)	2/5
Domino 2/6	-1	(456, 209)	6/2
Domino 3/3	-2	(168, 472)	3/3
Domino 3/4	1	(564, 336)	3/4
Domino 3/5	1,5	(171, 212)	5/3
Domino 3/6	0	(454, 338)	3/6
Domino 4/4	0	(362, 210)	4/4
Domino 4/5	1,5	(88, 212)	5/4
Domino 4/6	0	(89, 89)	6/4
Domino 5/5	-1	(267, 339)	5/5
Domino 5/6	-1	(640, 461)	5/6
Domino 6/6	-1	(551, 213)	6/6

Durum 4: Görüntü 720*576 çözünürlükte ve domino taşları rasgele bir şekilde yerleştirilmiştir. Resim 7.15 de domino taşlarının uygulama yazılımı tarafından etiketlenmiş görüntüsü ve Çizelge 7.5 de elde edilen değerler görülmektedir.



Resim 7.15. Durum 3 için elde edilen sonuç

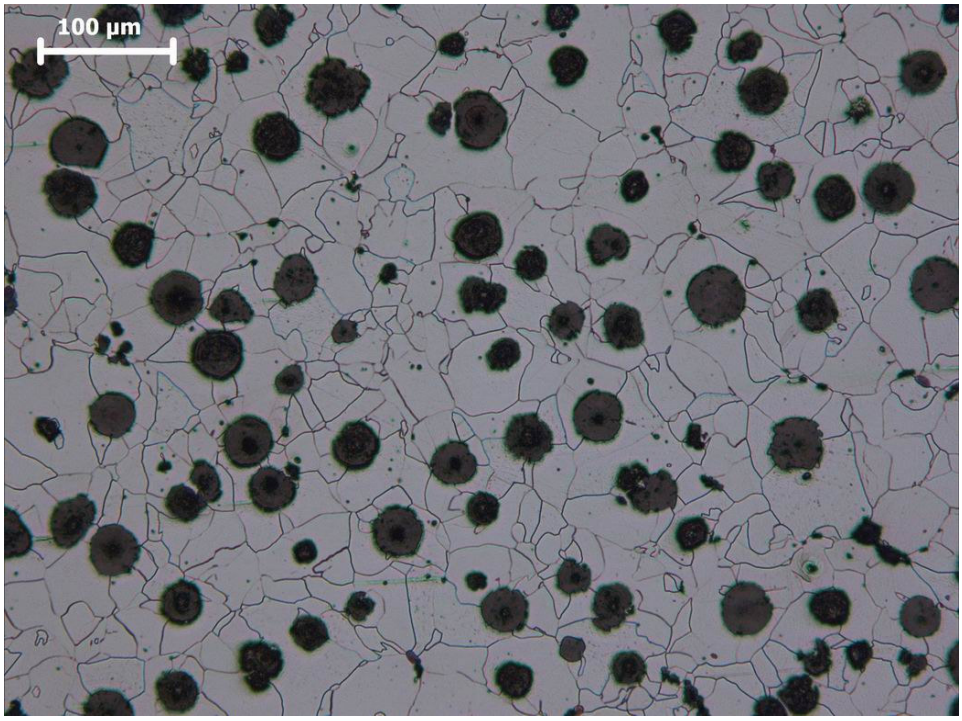
Çizelge 7.5. Durum 4 için elde edilen değerler

	Taş Açısı (◦)	Taş Merkezi koordinatı (piksel)	Ölçülen Taş Değeri
Domino 0/0	-5	(104,143)	0/0
Domino 0/1	-1	(174, 95)	0/1
Domino 0/2	22,5	(90, 348)	2/0
Domino 0/3	37,5	(639, 265)	3/0
Domino 0/4	-45	(504, 90)	4/0
Domino 0/5	-58,5	(266, 475)	0/5
Domino 0/6	19,5	(191, 183)	0/6
Domino 1/1	4	(352, 101)	1/1
Domino 1/2	-6	(614, 117)	1/2
Domino 1/3	4	(477, 227)	1/3
Domino 1/4	-46,5	(296, 208)	4/1
Domino 1/5	-48	(588, 371)	1/5
Domino 1/6	53	(457, 420)	6/1
Domino 2/2	-8,5	(377,482)	2/2
Domino 2/3	-79	(354, 391)	3/2
Domino 2/4	-20,5	(566, 217)	2/4
Domino 2/5	60	(275, 139)	2/5
Domino 2/6	-8	(172, 401)	6/2
Domino 3/3	-40	(118, 493)	3/3
Domino 3/4	-27	(530, 310)	3/4
Domino 3/5	74	(544, 474)	5/3
Domino 3/6	-7,5	(448, 314)	3/6
Domino 4/4	-0,5	(364, 293)	4/4
Domino 4/5	1,5	(179, 291)	4/5
Domino 4/6	-13,5	(106, 234)	6/4
Domino 5/5	11,5	(254, 307)	5/5
Domino 5/6	14,5	(623, 474)	5/6
Domino 6/6	12	(422, 163)	6/6

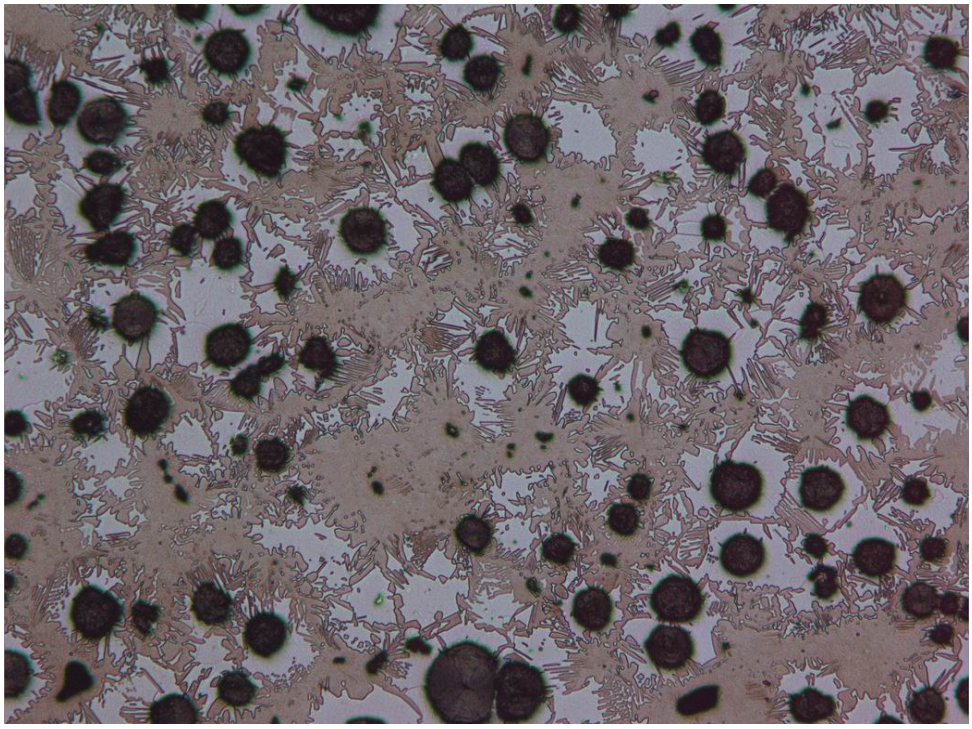
Elde edilen görüntüler için hazırlanan yazılım sorunsuz çalışmış ve doğru sonuçlar vermiştir. Çizelge 7.2'de de sunulduğu gibi sadece tekbir domino taşının altdeğeri yanlış okunmuştur. Bunun nedeni Resim 7.12(a)'da da görüldüğü gibi alttaki beş noktadan sağ üsttekinin 4 lü komşuluk ilişkisine göre orta ayıraca bağlantılı görünmesidir. Durum 1 deki hata öncelikli olarak yetersiz aydınlatmadan kaynaklanmaktadır. Bu sorun yüksek çözünürlüklü durum için öz konusu olmamıştır. Çünkü aynı bölge, daha yüksek çözünürlükte daha çok yoğunlukta örneklendiği için elde edilen görüntüde aynı bölge daha fazla piksel ile ifade edilmiştir. Bunun sonucunda ayrık bölgeleri, bölümlendirmek ve etiketlemek daha doğru sonuçlar vermiştir.

7.2 İkinci Bölüm Uygulama (Malzeme Görüntü İşleme)

İkinci grup uygulamada ise görüntü işleme teknikleri ile malzeme özellikleri belirlenmiştir. Uygulamanın bu bölümünde 2 farklı tip malzeme görüntüsü üzerinde çalışılmıştır. Bu görüntülerden ilki küresel grafit içeren dökme demir ve martenzit çeliği diğeri ise normalize edilmiş çeliktir. İlk görüntü üzerinde uygulanan işlemde amaç malzeme içerisindeki küresel grafit yüzdesini, küresel grafit alanını (piksel olarak), ve ortalama küresel grafit alanını (piksel olarak) bulmaktır. İkinci tip görüntü de ise amaç çelik görüntüsündeki fazlardan birinin yüzey alanının bulunmasıdır. Uygulamada kullanılan malzeme görüntüleri jpg sayısal resim formatında hazır olarak alınmış herhangi bir deney düzeneği kurulmamıştır. Resim 7.16 ve Resim 7.17'de kullanılan resimler görülmektedir.

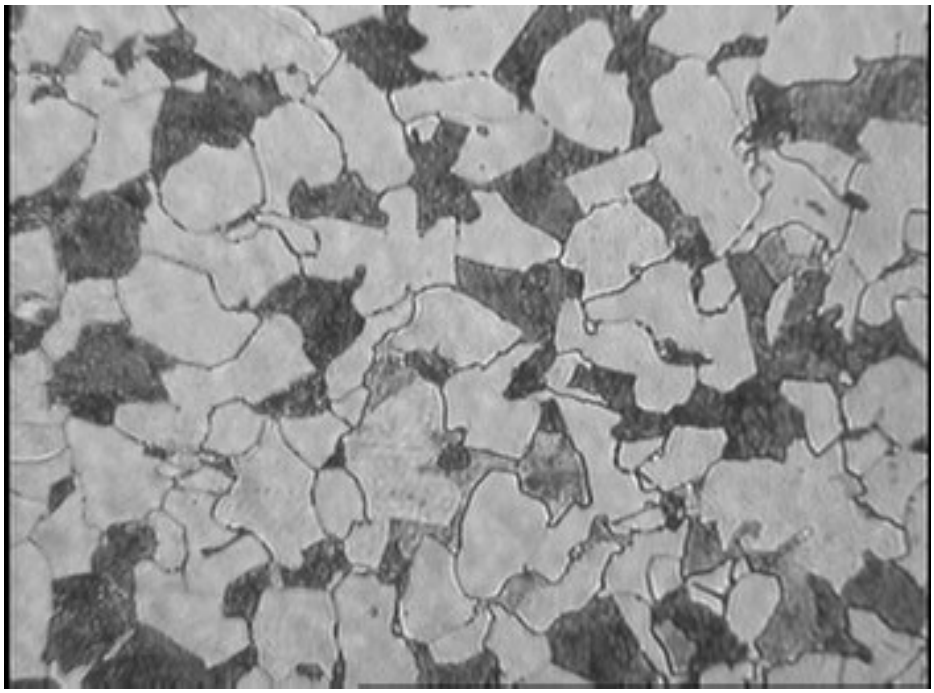


(a)



(b)

Resim 7.16. (a) Küresel grafitli dökme demir. (b) Küresel grafitli martenzit çeliği



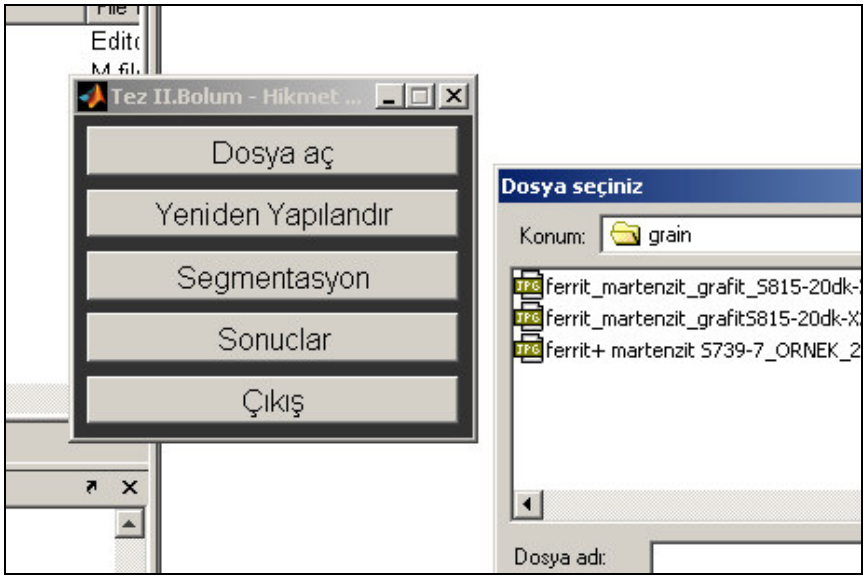
Resim 7.17.. İşlemsiz 1030 çeliği

7.2.1 Malzeme görüntü işleme

Malzeme görüntü işlemede birinci bölüm uygulamada kullanılan eşikleme filtreleme gibi görüntü işleme teknikleri yanında yapısal görüntü işleme watershed bölümlendirme teknikleri kullanılmıştır. Eşikleme ve watershed bölümlendirme sonrası elde edilen görüntüler üzerinde etiketleme ve özellik çıkarma işlemleri uygulandıktan sonra elde edilen sonuçlar özel bir görüntü işleme yazılımında elde edilen sonuçlarla karşılaştırılmıştır. Sonuçlar tablo halinde sunulmuştur.

Oluşturulan yazılımın ana menüsü Şekil 7.5 de görülmektedir. Oluşturulan yazılımda her aşama bir öncekinde elde edilen verileri girdi olarak kullanmaktadır. Yazılımın ilk aşaması işlem yapılacak dosyanın seçilmesidir. Bunun için menuden “Dosya aç” seçeneği kullanılmaktadır. Açılan dosya otomatik olarak gri seviyeye dönüştürülmektedir. Bu bölümde önemli olan resim üzerinde işlem uygulanacak

alanın seçilmesidir. İlgilenilen bölge (Region of Interest – ROI) seçildikten sonra tüm işlemler bu bölge üzerinde gerçekleştirilmektedir.

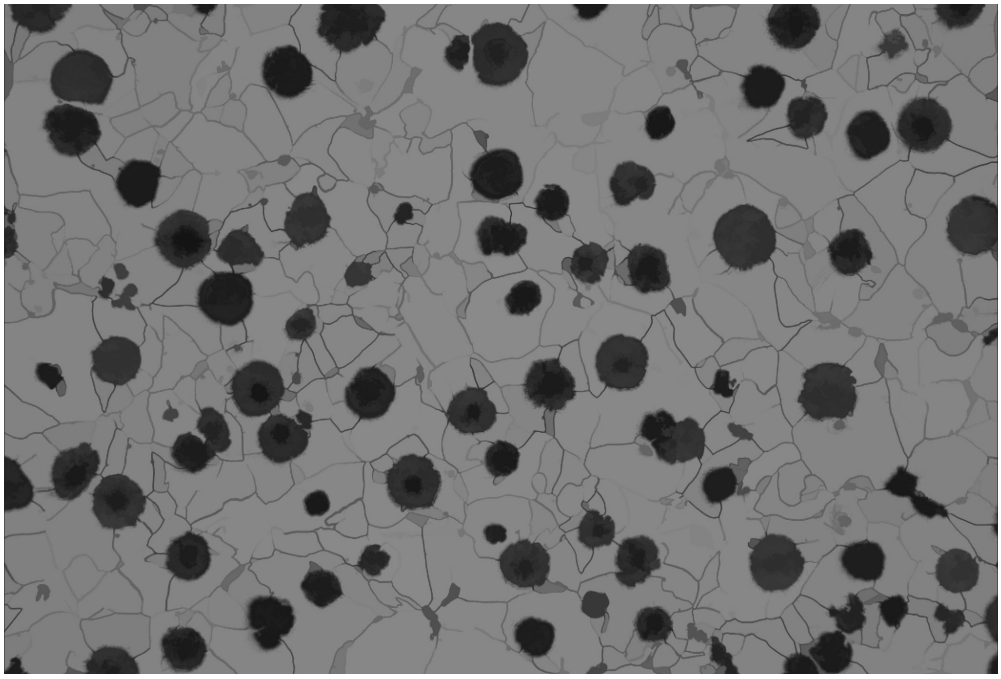


Şekil 7.5. İkinci bölüm uygulama ana menu görüntüsü

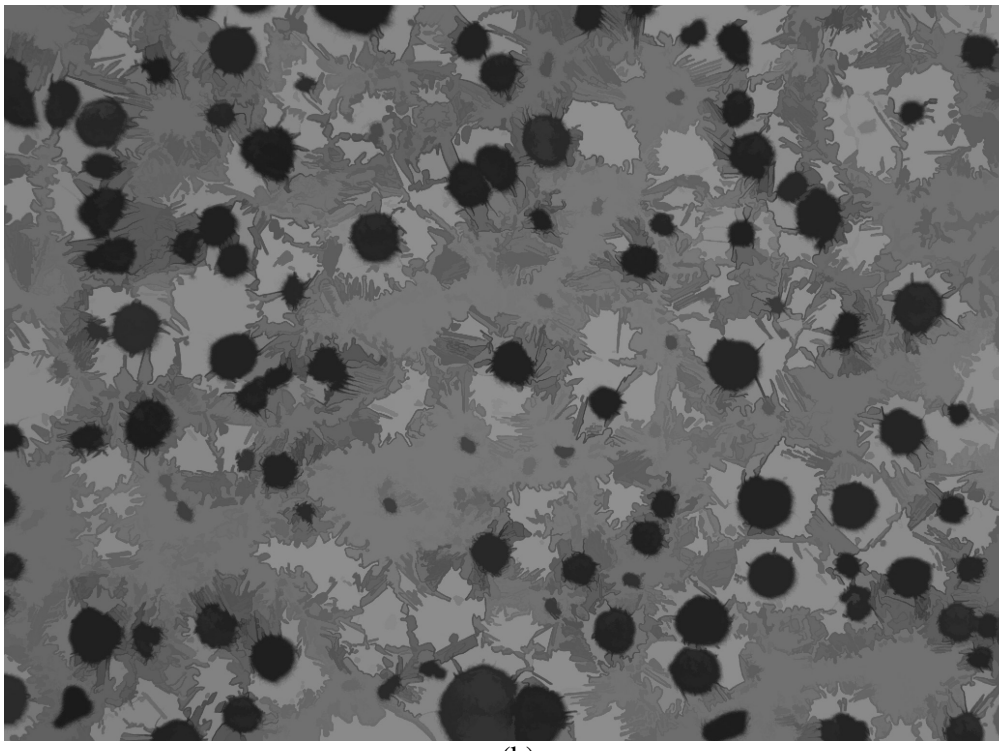
7.2.3 Yeniden yapılandırma

Görüntü yeniden yapılandırma Bölüm 4.2 deki işlemleri kapsamaktadır. Görüntü yeniden yapılandırma malzeme bölümlendirmesi yapmadan önce görüntü içerisindeki piksel gri seviye değerlerinin yeniden düzenlenmesi işlemi olarak özetlenebilir. Bunun için ikili düzey veri yapısına sahip yapısal eleman kullanılmaktadır. Oluşturulan yazılımda bu yapısal elemanın büyüklük ve biçimi otomatik olarak seçilmektedir.

Yeniden yapılandırma sonrası gri seviye değerlerindeki ani ve aşırı oynamalar (küçük beyaz nokta veya küçük siyah nokta gibi gürültüler ve aynı nesne içerisindeki gölge veya benzeri etkilerden dolayı oluşan piksel gri seviye değerindeki değişimler) yok edilerek bölümlendirmeye daha elverişli bir görüntü elde edilmiş olur. Resim 7.16 (a),(b) ve Resim 7.17 deki görüntülere ait yeniden yapılandırma sonuçları Resim 7.18 (a),(b) ve Resim7.19' da görülmektedir.

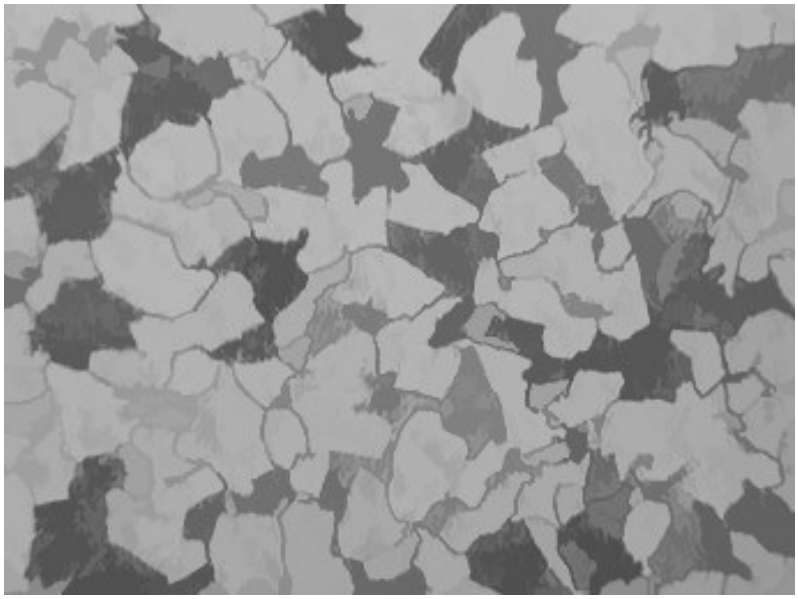


(a)



(b)

Resim 7.18. (a) Yeniden yapılandırılmış küresel grafitli dökme demir görüntüsü (b) Yenide yapılandırılmış martenzit çeliği görüntüsü



Resim 7.19. Yeniden yapılandırılmış işlemsiz çelik görüntüsü

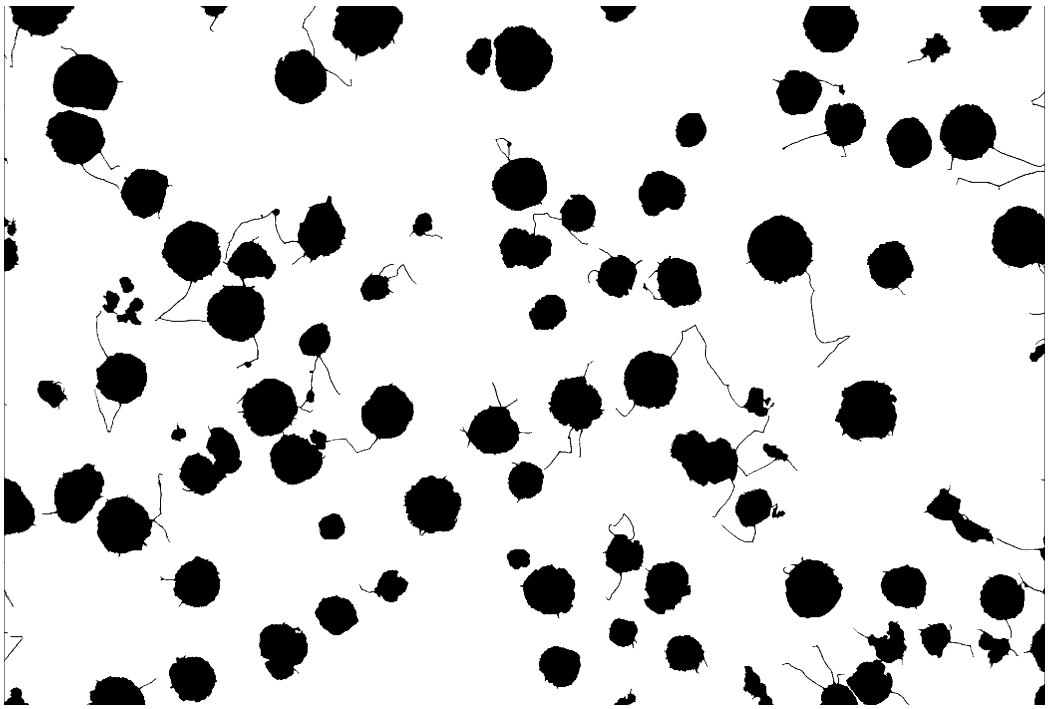
7.2.4 Bölümlendirme

Bölümlendirme işlemi görüntü içerisinde istenen bölgelerin geriplandan ayrılması için kullanılmaktadır. İkinci bölüm uygulamada bölümlendirme işlemi için bölüm 5'de anlatılan Marker kontrollü Watershed ve genel eşikleme teknikleri kullanılmıştır. Yazılımdaki bölümlendirme menüsü Şekil 7.6'de görülmektedir. Buradaki menüden bölümlendirme için uygulanmak istenen teknik seçilmelidir.

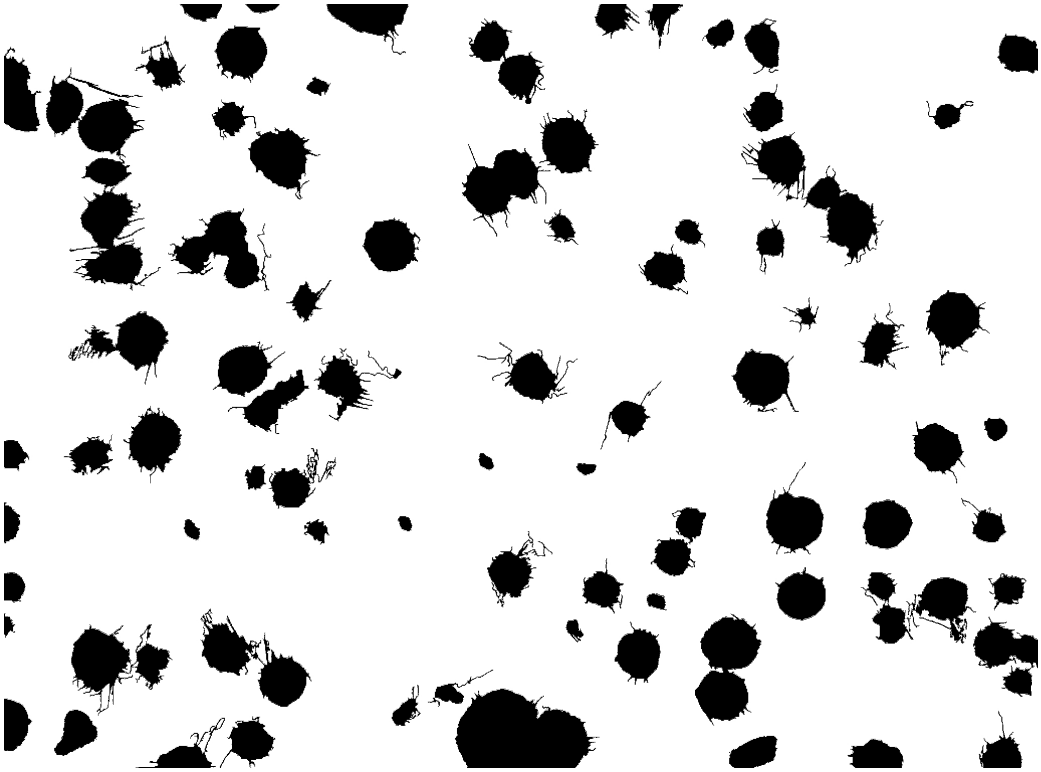
Eşikleme ile bölümlendirme işlemi kullanıcı kontrolünde uygulanmaktadır. Otomatik belirlenen eşik değerine göre kullanıcı eşik değerini değişirerek bölümlendirme sonuçunu seçilen taneler beyaz çizgi ile etrafları çizilerek seçimin doğru ve yeterli olup olmadığı kullanıcıya sorulmaktadır. Eğer herhangi bir eşik değeri verilmezse otomatik belirlenen eşik değerine göre işlemler yapılmaktadır. Resim 7.20 ve Resim 7.21'de yeniden yapılandırılmış görüntünün, eşikleme ile elde edilen ikili düzey görüntüleri görülmektedir.



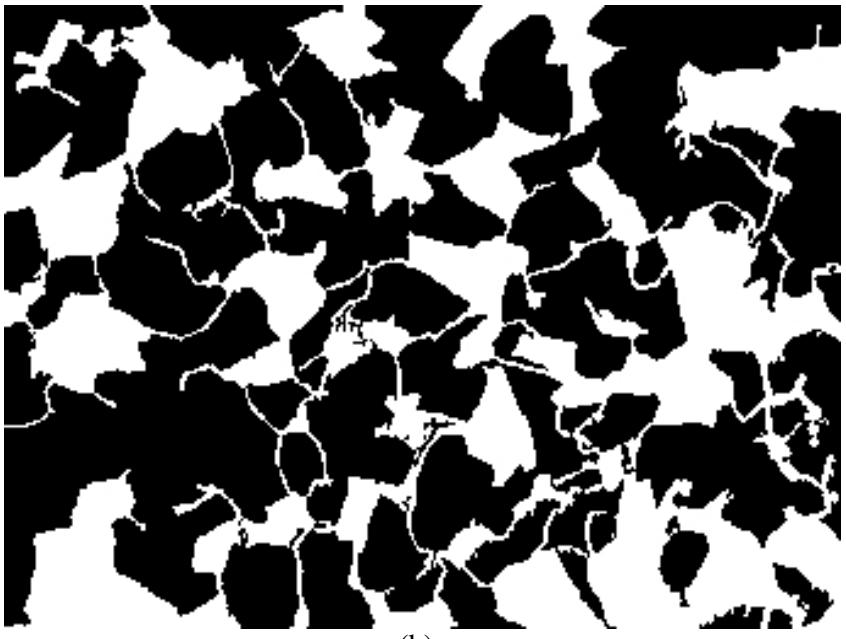
Şekil 7.6. Bölümlendirme menüsü



Resim 7.20. Dökme demir görüntüsündeki küresel grafitler



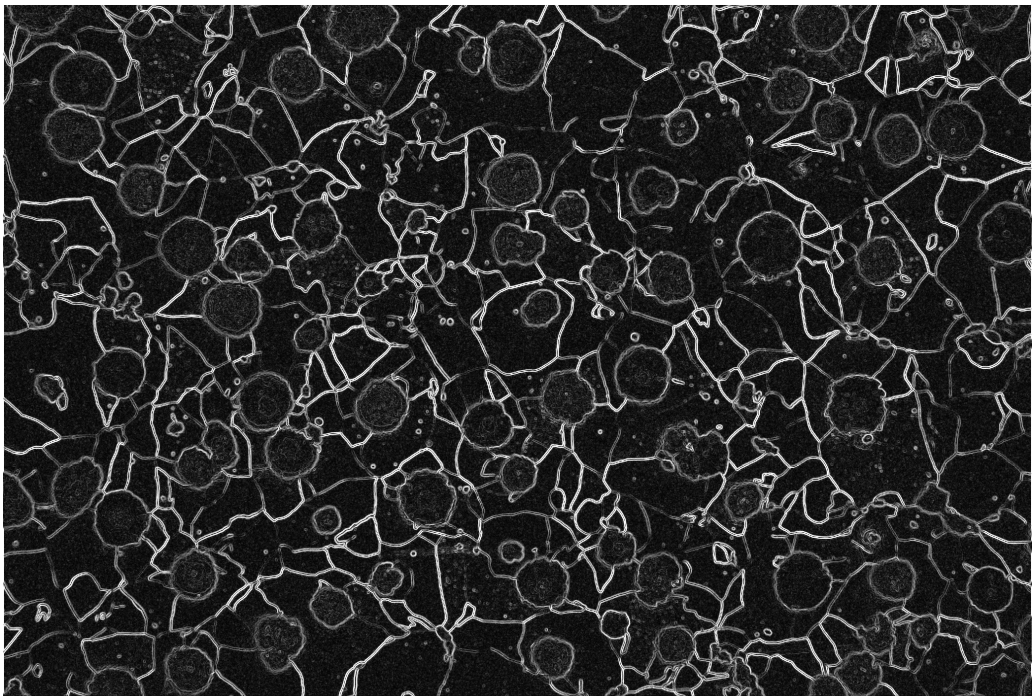
(a)



(b)

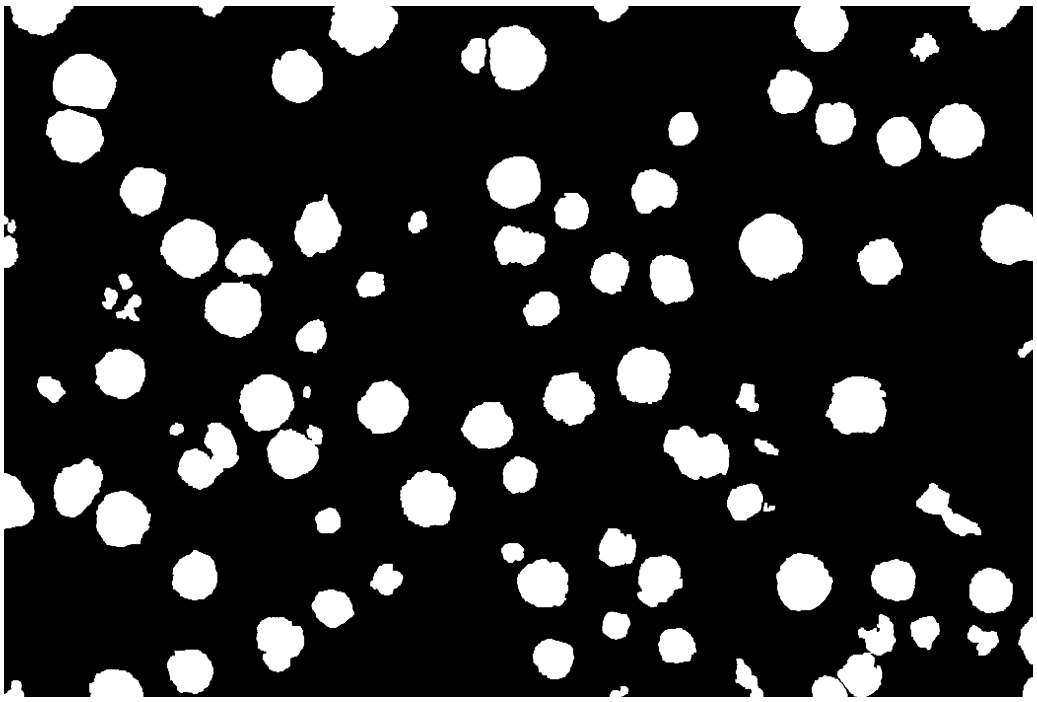
Resim 7.21. (a) ikili düzey martenzit çeliği görüntüsü (b) ikili düzey çelik görüntüsü

Watershed bölümlendirmesi direk olarak uygulandığında aşırı bölümlendirme sonuçunu vermektedir. Bu nedenle marker kontrollü watershed tekniği uygulanmıştır. Marker kontrollü Watershed tekniği kullanılırken ön işlem olarak gri seviye görüntünün gradient genliği elde edilmiştir. Gradient genliği alınmış görüntüde sınırlar yüksek gri seviye değeri sahipken diğer her yere düşük gri seviye değeri atanır. Böylece nesne sınırları ile watershed sınırlarının daha iyi çakışması sağlanır. Resim 7.22, Resim 7.16 (a)'da ki görüntünün gradient genlik görüntüsüdür.

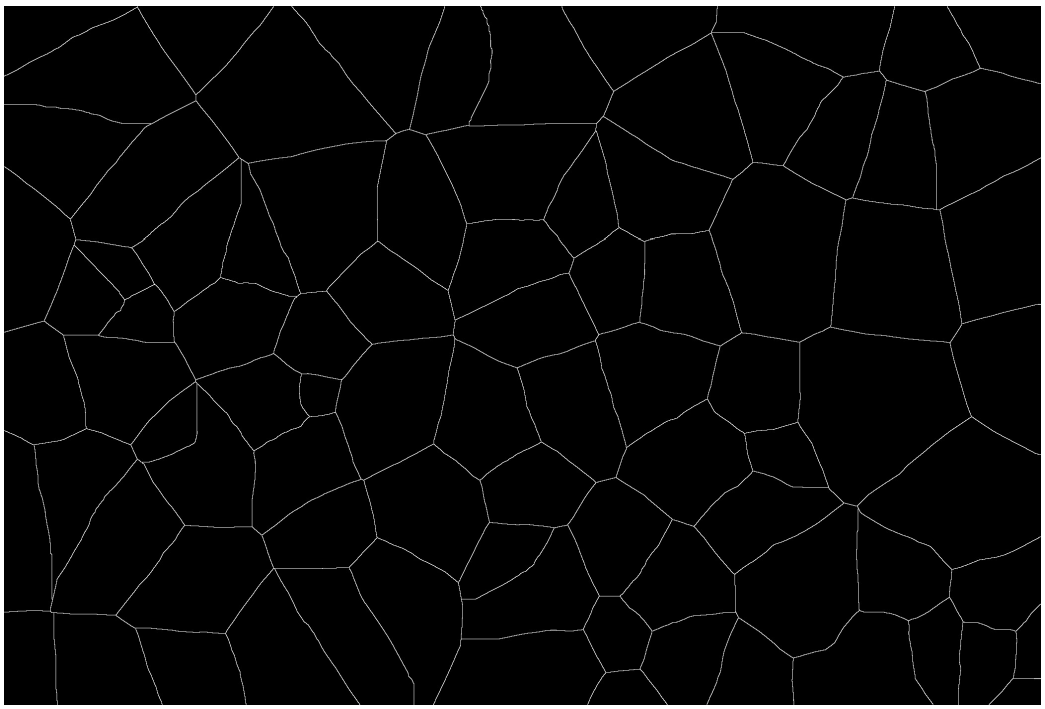


Resim 7.22. Küresel grafitli dökme demire ait gradient genlik görüntüsü

Watershed bölümlendirme, gradient genlik görüntüsü bulunduktan sonra buna önplan ve geriplan görüntüsüde birleştirilerek elde edilen görüntüde uygulanmıştır. Bu işlemin asıl nedeni görüntü içerisindeki yeniden yapılandırılmış görüntüde de belirgin olan tane sınırlarının watershed sınırları olarak görülmesini engellemektir. Birleştirilmiş görüntüyü uygulama yazılımı otomatik olarak elde etmektedir. Böylece daha doğru watershed bölümlendirme elde edilmektedir. Resim 7.23 de önplan görüntüsü, Resim 7.24 de ise geriplan görüntüsü verilmiştir.

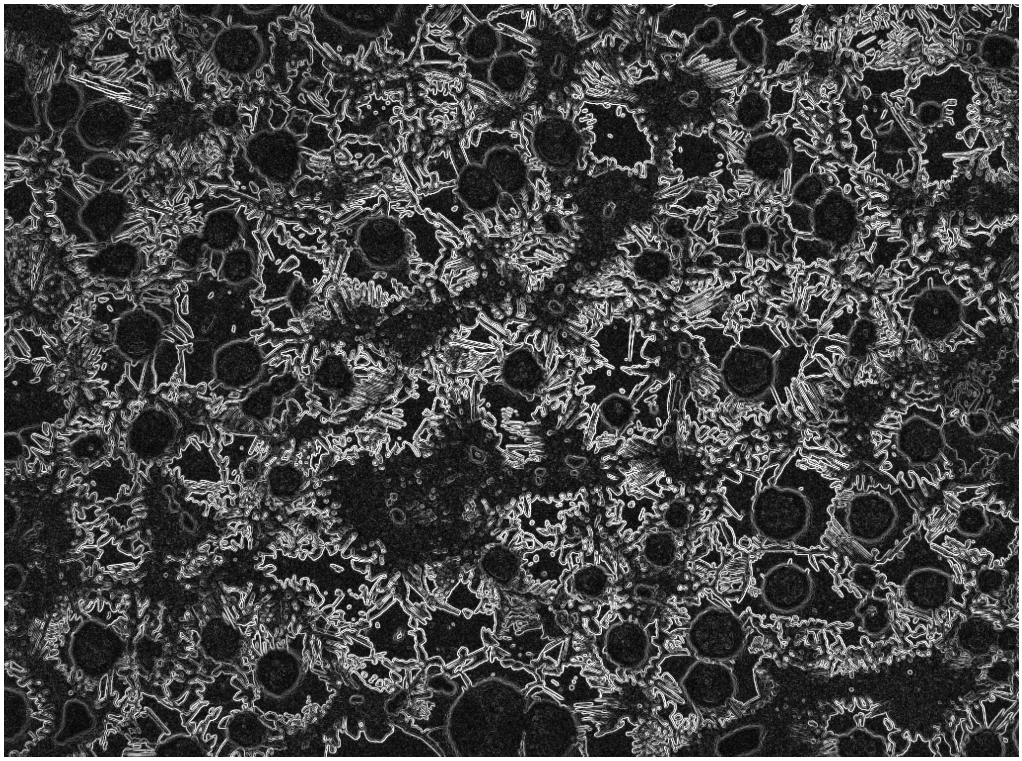


Resim 7.23. Küresel grafitli dökme demir için önplan görüntüsü

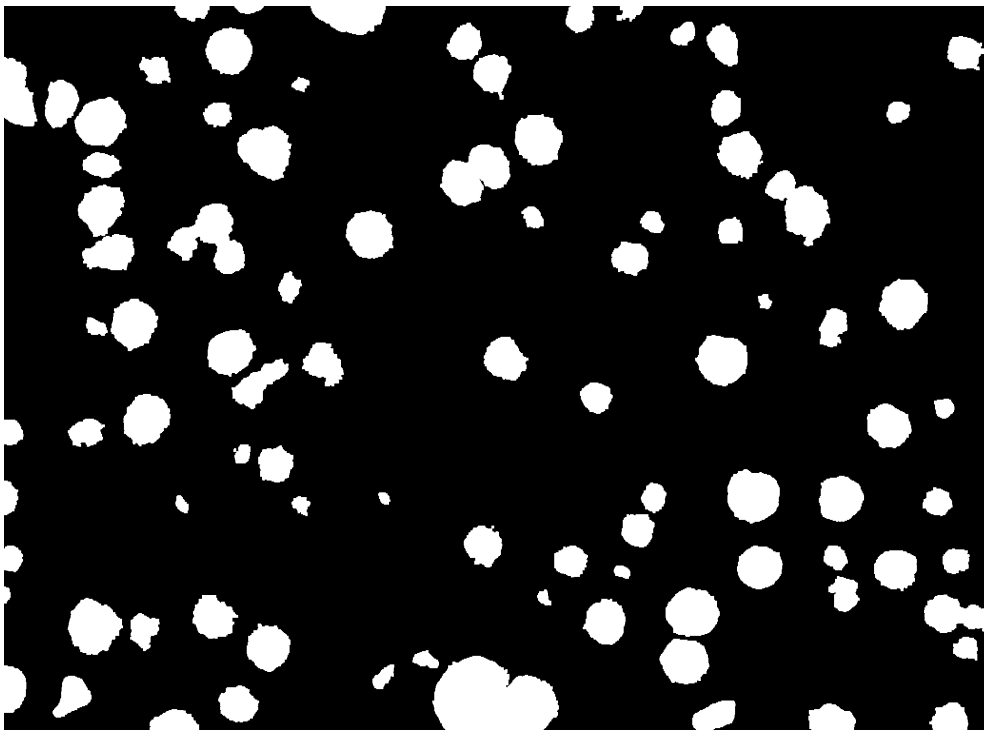


Resim 7.24. Kürsel grafitli dökme demir için geriplan görüntüsü

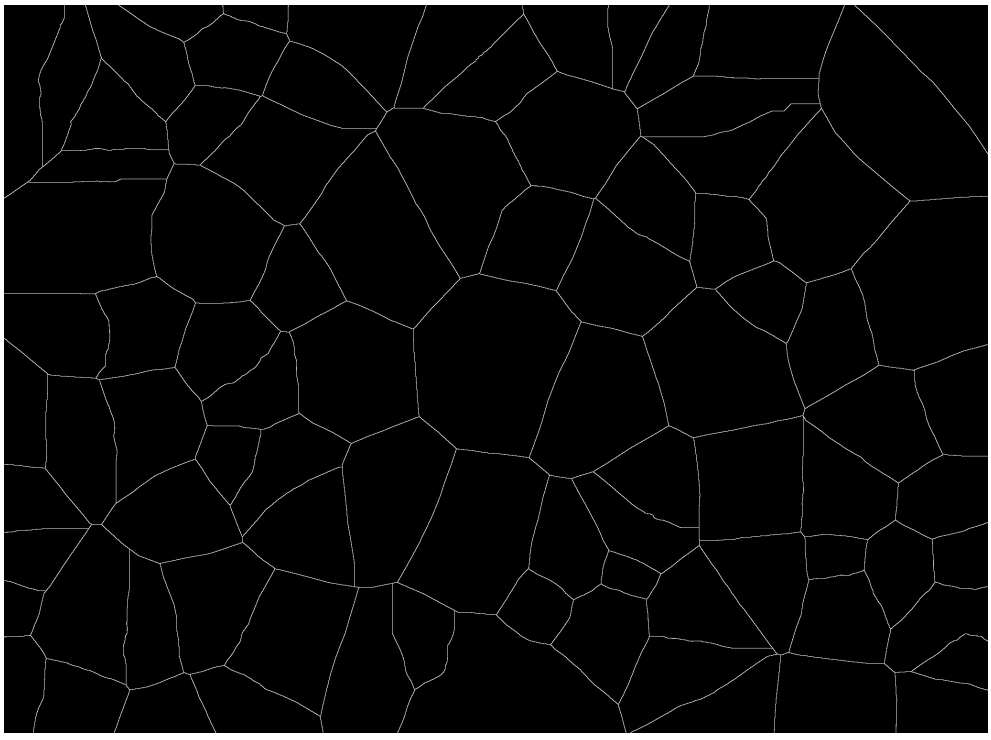
Küresel grafitli martenzit çeliği için gradient genlik önplan ve geriplan görüntüleri sırasıyla Resim 7.25- Resim 7.27 de dir. Watershed bölümlendirmesi sadece küresel grafit içeren (birinci tip) görüntüler için uygulanmıştır. Tane alanının belirlendiği görüntülerde (ikinci tip görüntüler) sadece eşikleme tekniği kullanılarak bölümlendirme yapılmıştır.



Resim 7.25. Martenzit çeliği için gradient genliği görüntüsü



Resim 7.26. Martenzit çeliği için önplan görüntüsü

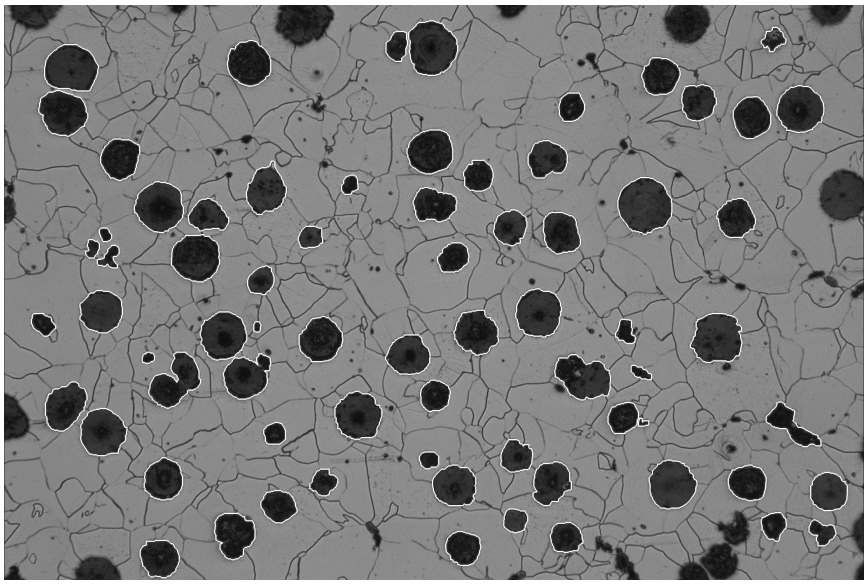


Resim 7.27. Martenzit çeliği için geriplan görüntüsü

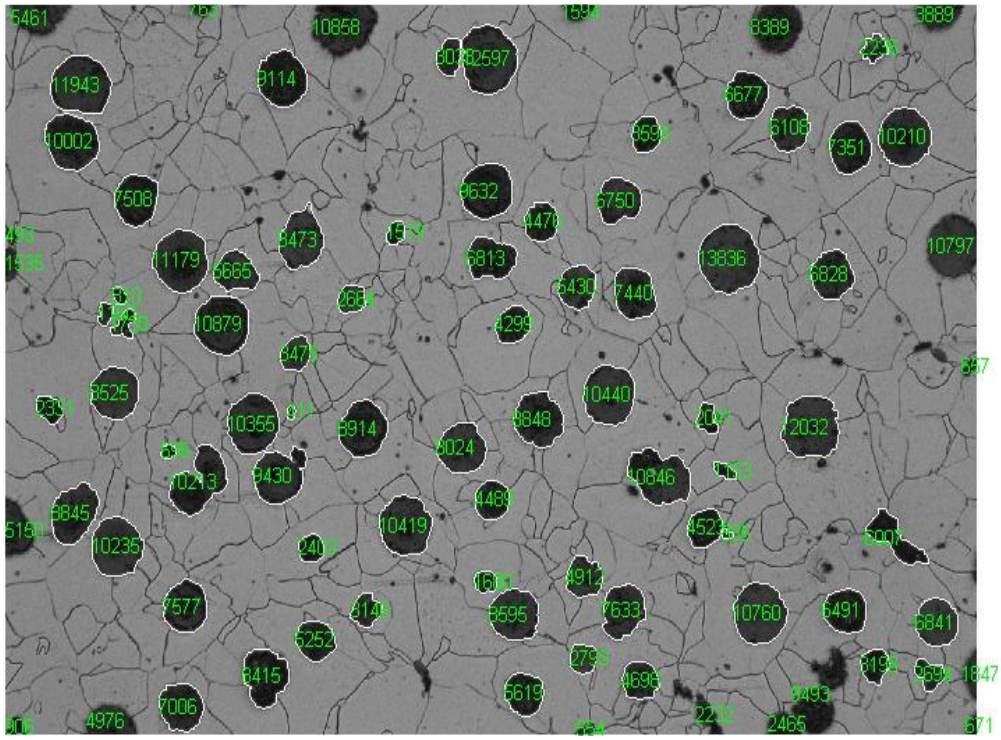
7.2.5 İkinci bölüm uygulama sonuçları

Uygulama sonuçları en son menü seçeneği olarak sunulmuştur. Sonuçlar butonuna basıldığında öncelikli olarak Bölümlendirme sonrası elde edilen görüntü içindeki taneler etiketlenerek özellikleri belirlenir ve etiketlenmiş görüntü kullanıcıya sunulur. Burada en büyük ve en küçük tane alanları görülmektedir. Bu etiketlemeye göre kullanıcı ortalama tane büyüklüğü içi hariç tutmak istediği tane piksel alan sınır değerini girerek ortalama tane alanı ve toplam tane alanı piksel cinsinden hesaplanarak kullanıcıya gösterilir. Ayrıca toplam küresel grafit piksel alanı, toplam resim alanına bölünerek grafit yüzdesi hesaplanabilir. Toplam tane alanı hesaplanırken tüm tanelerin piksel alanı işleme katılmış, tane ortalama alanı hesaplanırken sınırda kalan yarım taneler işleme katılmamıştır. Küresel grafitli dökme demir için Resim 7.28 ve Resim 7.29 da eşikleme tekniği ile Resim 7.30 ve Resim 7.31'de marker kontrollü watershed tekniği ile elde edilen sonuçlar sunulmuştur. Her iki teknikle elde edilen tane sınırları watershed için kırmızı, eşikleme için sarı renkte olacak şekilde Resim 7.32'de gösterilmiştir.

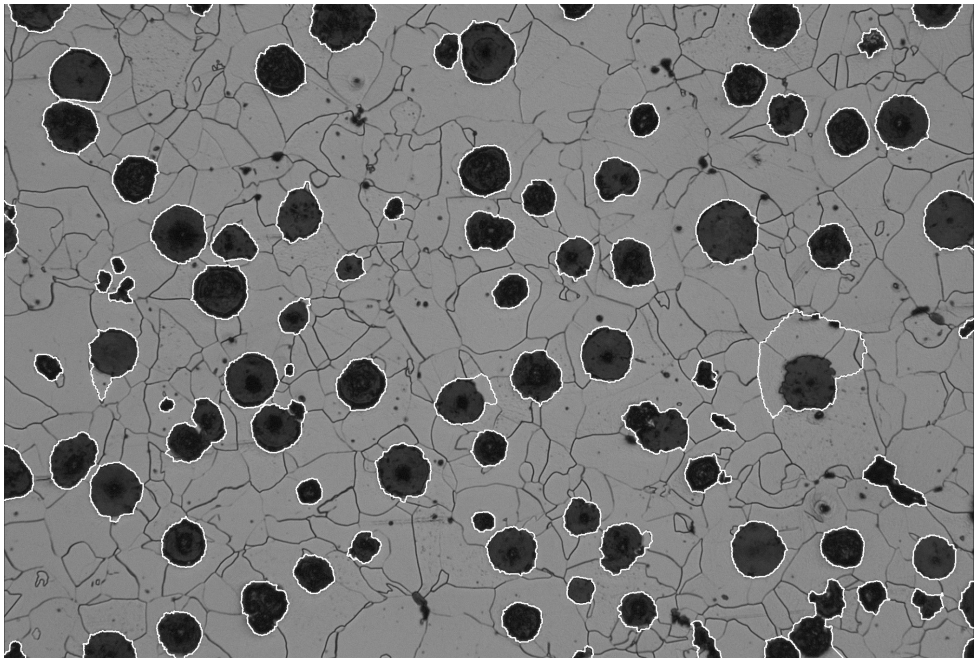
Küresel Grafitli Dökme Demir için elde edilen sonuçlar



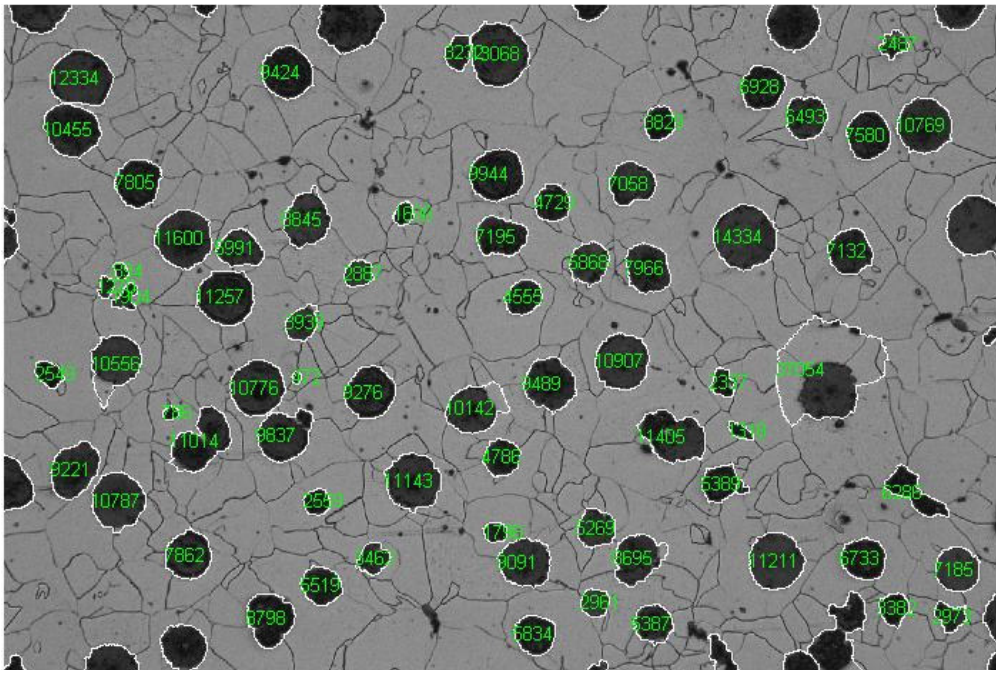
Resim 7.28. Eşikleme ile elde edilen tane sınırları beyaz olarak çizili



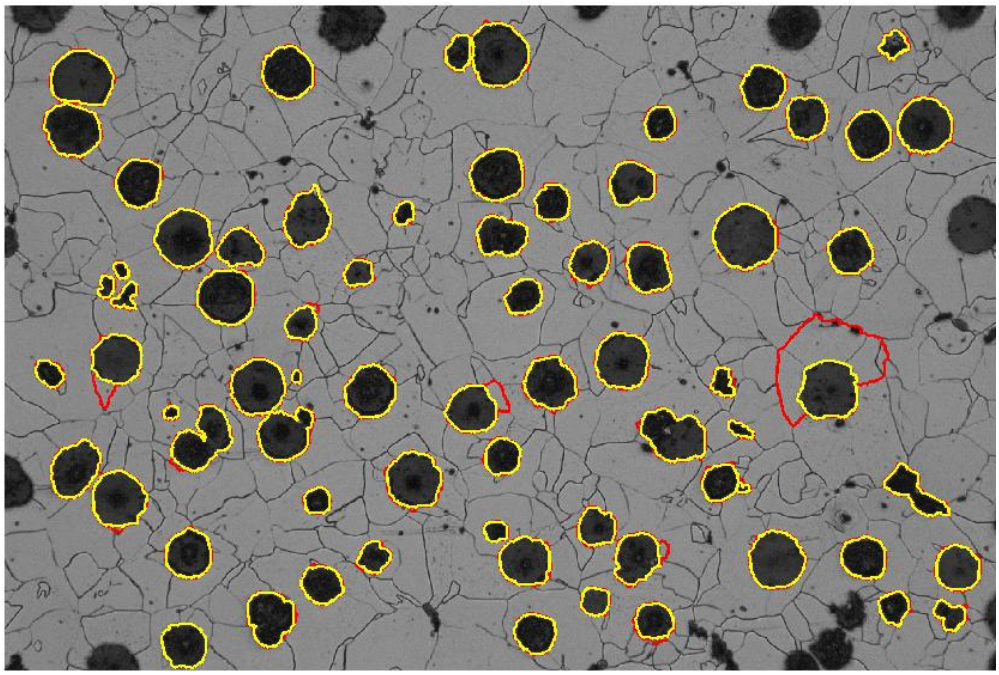
Resim 7.29. Küresel grafitli dökme demir için eşikleme sonrası elde edilen tane boyutları



Resim 7.30. Watershed tekniği ile elde edilen tane sınırları beyaz ile gösterilmiştir



Resim 7.31. Küresel grafitli dökme demir için watershed sonrası elde edilen tane boyutları



Resim 7.32. Watershed ve Eşikleme ile bulunan tane sınırlarının karşılaştırması. (kırmızı watershed ile bulunan sınırlar, sarı Eşikleme ile bulunan sınırlar)

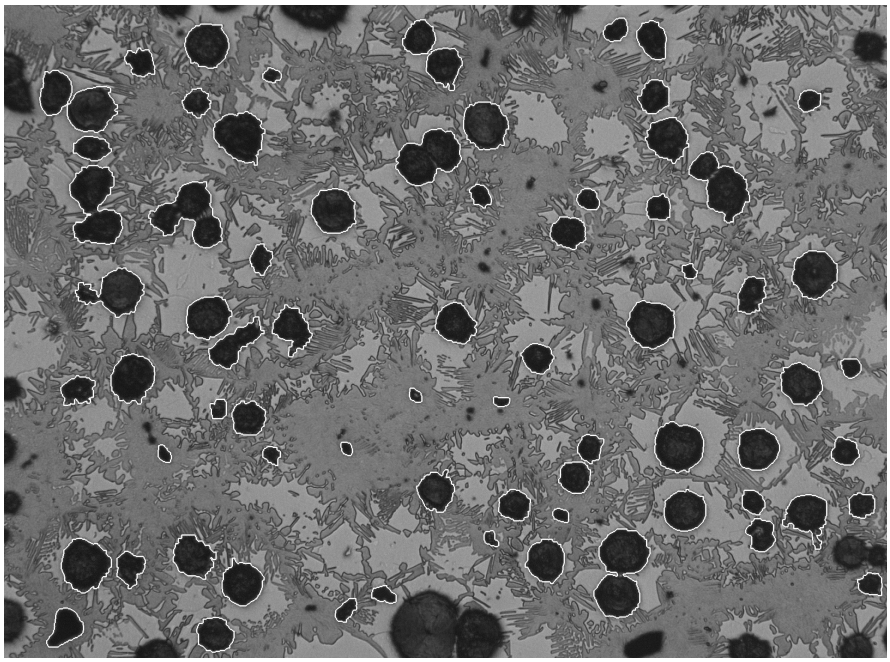
Elde edilen değerler ise Çizelge 7.6 deki gibidir.

Çizelge 7.6. Küresel Grafitli Dökme Demir için sonuçlar

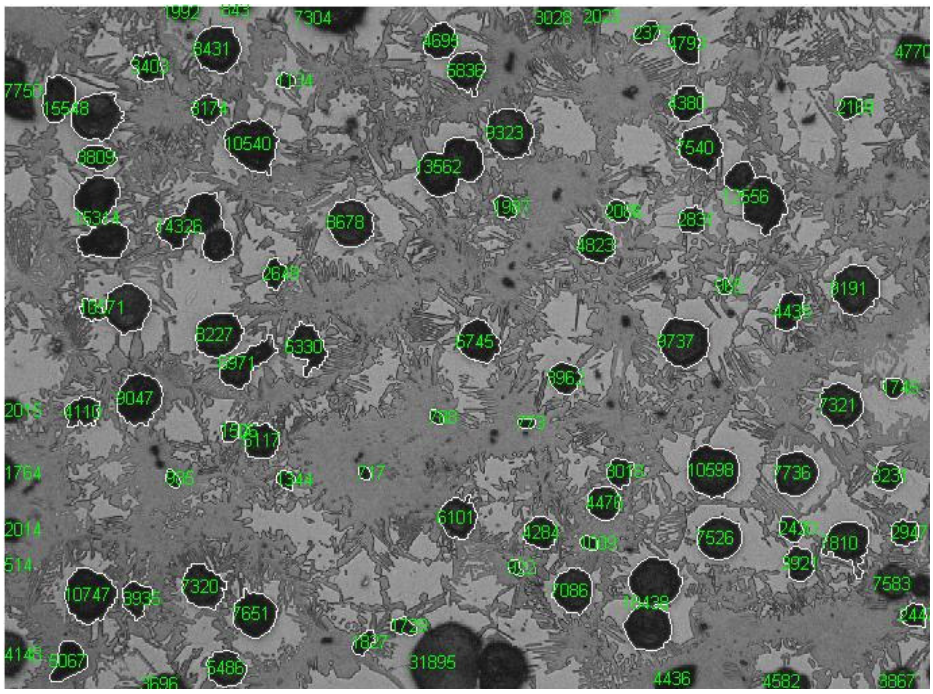
	Eşikleme ile	Watershed ile
Ortalama tane alanı (piksel)	8676	8885
Grafit Tanelerini Toplam Alanı (piksel)	525025	530113
Resim toplam alanı (piksel)	2925288	2925288
Resim içindeki grafit yüzdesi	%17,95	%18,12

Küresel Grafitli Martenzit Çeliği için elde edilen sonuçlar

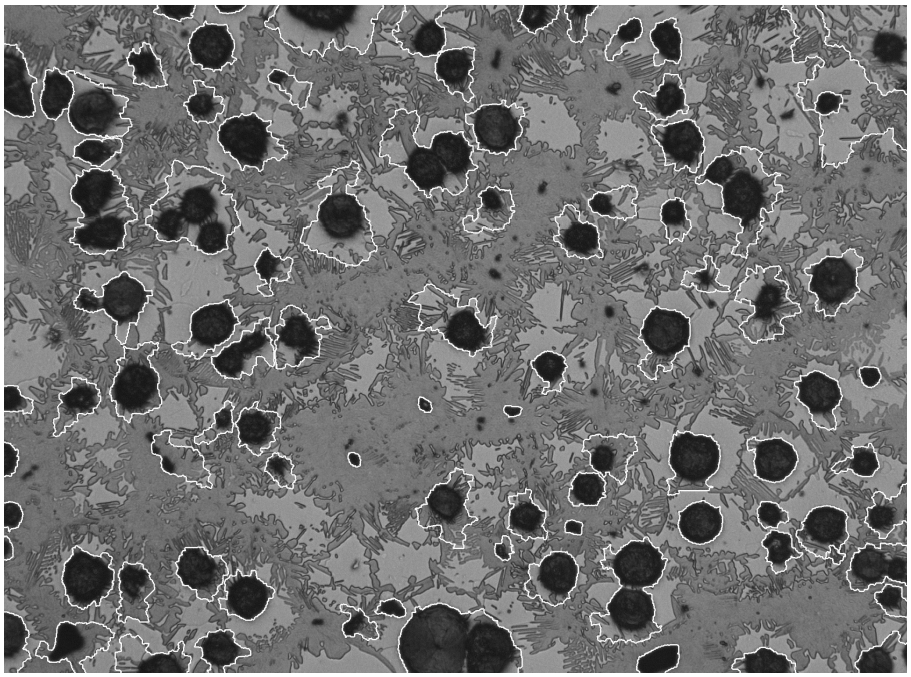
Küresel grafitli martenzit çeliği için eşikleme ve watershed ile elde edilen sonuçlar sırasıyla Resim 7.33, Resim 7.34'de ve Resim 7.35, Resim 7.36'da dir. Çizelge 7.7 de martenzit çeliğine ait sonuçlar sunulmuştur. Her iki teknikle elde edilen tane sınırları watershed için kırmızı, eşikleme için sarı renkte olarak Resim 7.37 de gösterimiştir.



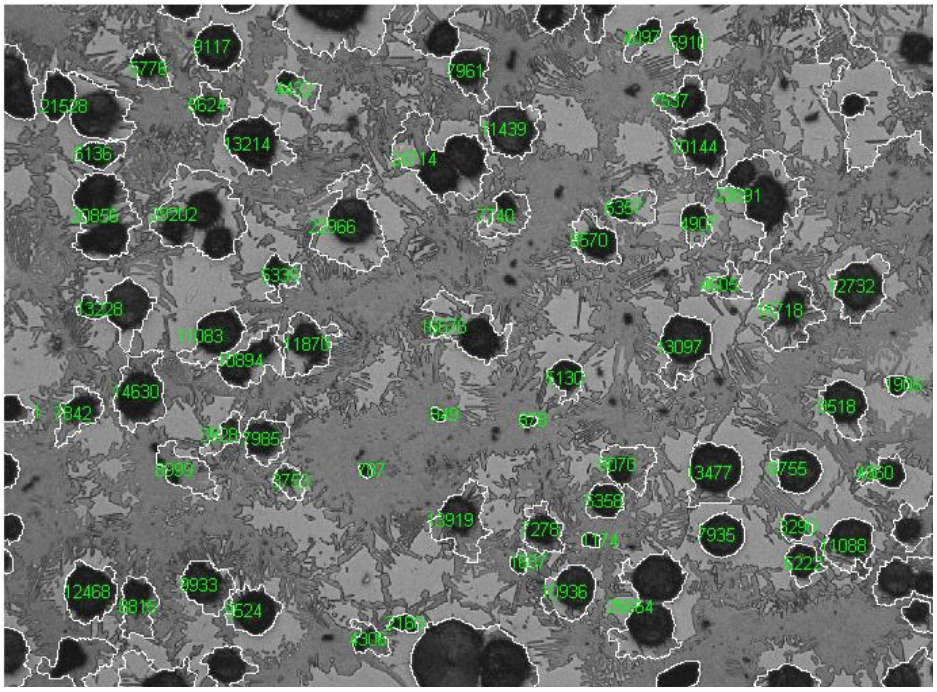
Resim 7.33. Eşikleme ile elde edilen tane sınırları beyaz olarak çizilmiştir



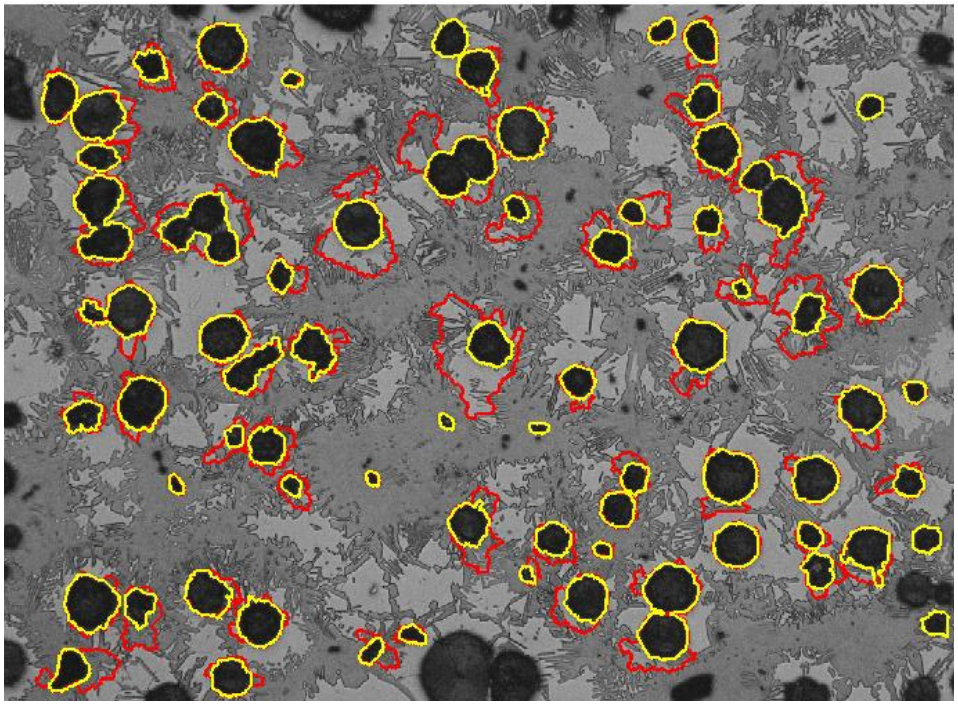
Resim 7.34. Küresel grafitli martenzit çeliği için eşikleme sonrası elde edilen tane boyutları



Resim 7.35. Watershed tekniği ile elde edilen tane sınırları beyaz renkle çizilmiştir



Resim 7.36. Küresel grafitli martenzit çeliği için watershed sonrası elde edilen tane boyutları



Resim 7.37. Watershed ve Eşikleme ile bulunan tane sınırlarının karşılaştırması. (kırmızı watershed ile, sarı Eşikleme ile bulunan sınırlar)

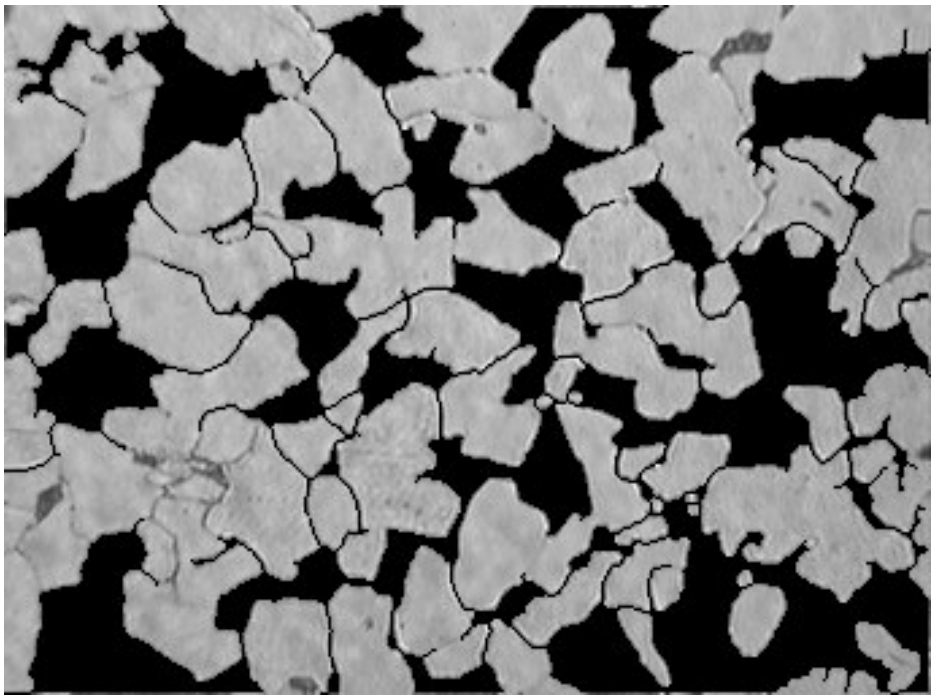
Elde edilen değerler ise Çizelge 7.7 deki gibidir.

Çizelge 7.7. Küresel grafitli Martenzit çeliği için sonuçlar

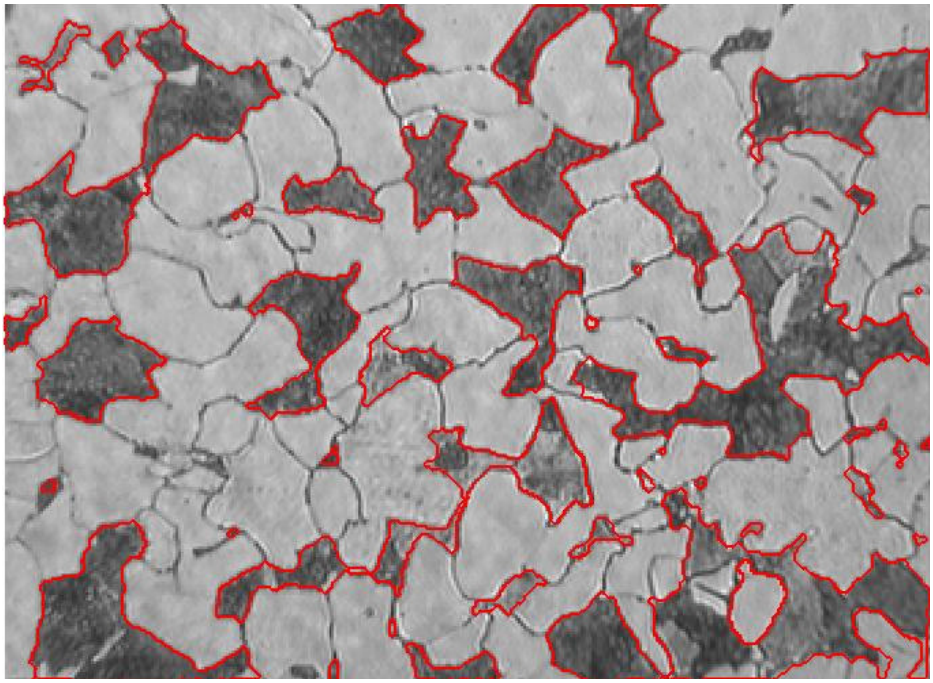
	Eşikleme ile	Watershed ile
Ortalama tane alanı (piksel)	8439	9266
Grafit Tanelerini Toplam Alanı (piksel)	495505	503793
Resim toplam alanı (piksel)	3131739	3131739
Resim içindeki grafit yüzdesi	%15,82	%16,09

İşlemsiz Çelik için elde edilen sonuçlar

İşlemsiz çeliğe ait tane büyüklüğü görüntülerine ait sonuçlar ise Resim 7.38 ve Resim 7.39 da dir.



Resim 7.38. Bölümlendirme sonucunun orijinal görüntüde gösterimi



Resim 7.39. Belirlenen Siyah faza ait bölge sınırları kırmızı ile çizilmiştir

Çizelge 7.8. İşlemsiz çelik için sonuçlar

	Eşikleme ile
Siyah fazın alanı (piksel)	26468
Resim toplam alanı (piksel)	87122
Resim içindeki siyah faz yüzdesi	%30,38

Küresel Grafitli dökme demir ve işlemsiz çeliğe ait orijinal görüntüler hata kontrol maksadıyla Leica image processing yazılımında işleme tabi tutulmuşlardır.

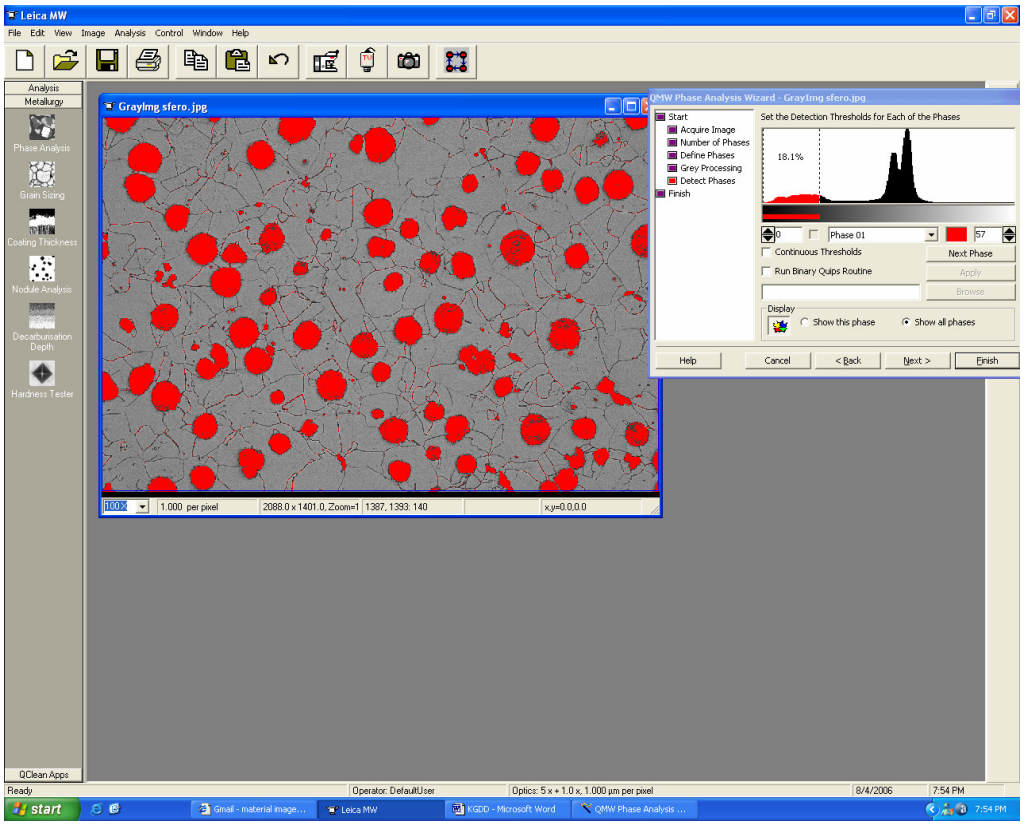
Üçüncü Parti Yazılımı ile Küresel Grafitli Dökme Demir için sonuçlar :

Oluşturulan yazılımın sonuçlarını kontrol maksadıyla, girdi olarak kullanılan gri seviye görüntü, üçüncü parti görüntü işleme yazılımında işleme tabi tutulmuştur. Uygulama yazılımı ve ticari yazılım da Küresel Grafitli Dökme Demir için elde

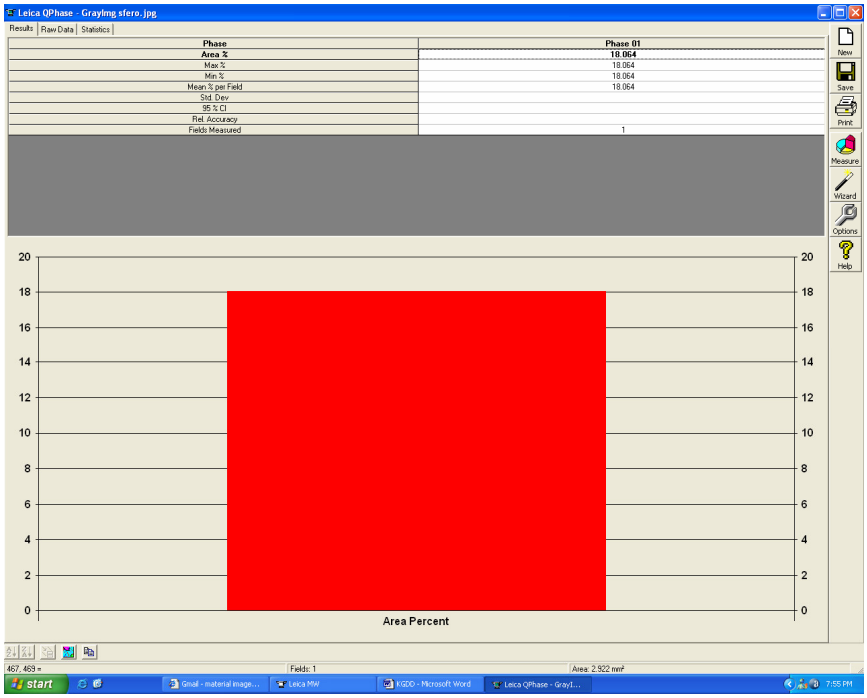
edilen sonuçlar Çizelge 7.9 da görülmektedir. Şekil 7.7 ve Şekil 7.8' de ise ticari yazılımdaki uygulama aşamaları görülmektedir.

Çizelge 7.9. Küresel Grafitli Dökme Demir için sonuçların karşılaştırması

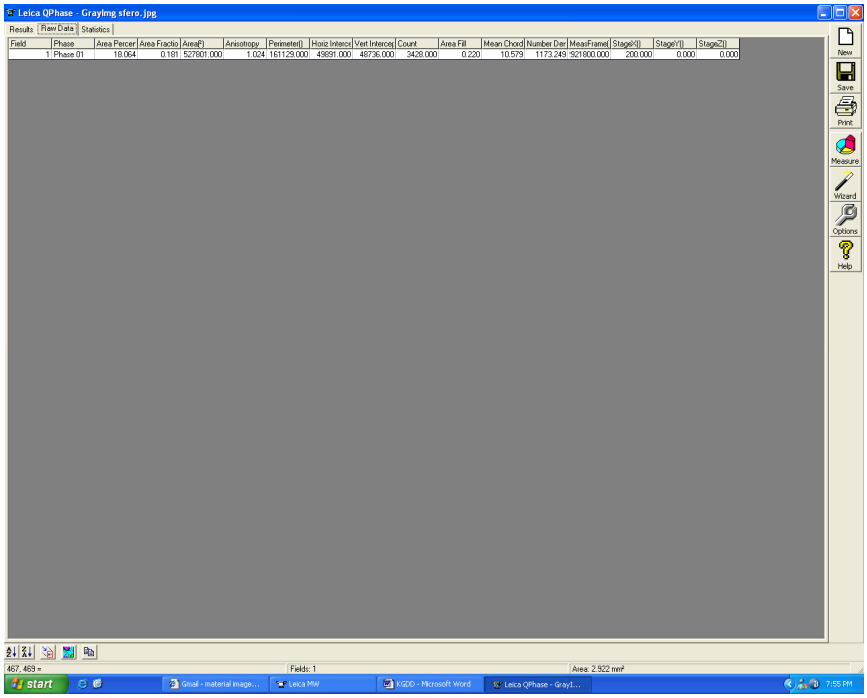
DEĞERLER	LEICA Q550 MW	Oluşturulan Uygulama yazılımı ile	FARK
Piksel	527801	525025	2776
Alan (%)	18,064	17,95	0,114



Şekil 7.7. Küresel grafitli dökme demirin diğer yazılım ile işlenmesi



(a)



(b)

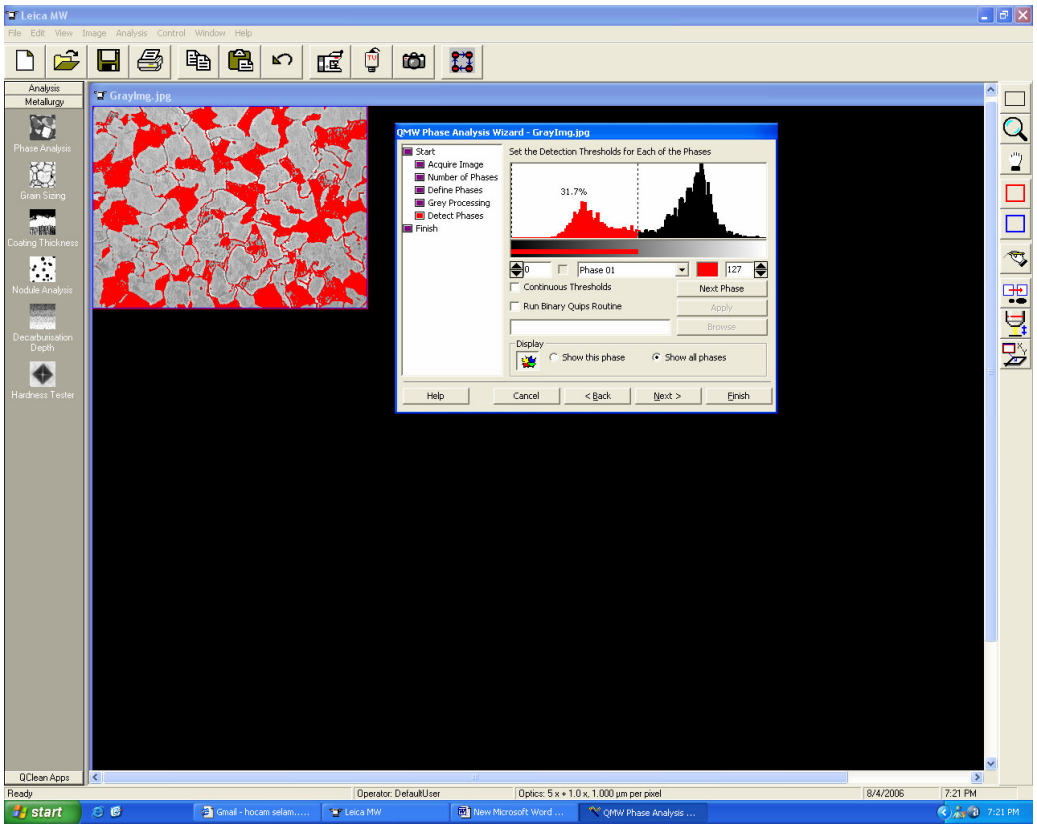
Şekil 7.8. (a) Küresel grafitli dökme demir için diğer yazılımın elde ettiği sonuçlar
(b) Küresel Grafitli Dökme demir için diğer yazılımın elde ettiği sonuçlara ait diğer aşamanın gösterimi

Üçüncü Parti Görüntü İşleme Yazılımı ile işlemsiz 1030 çeliği için sonuçlar :

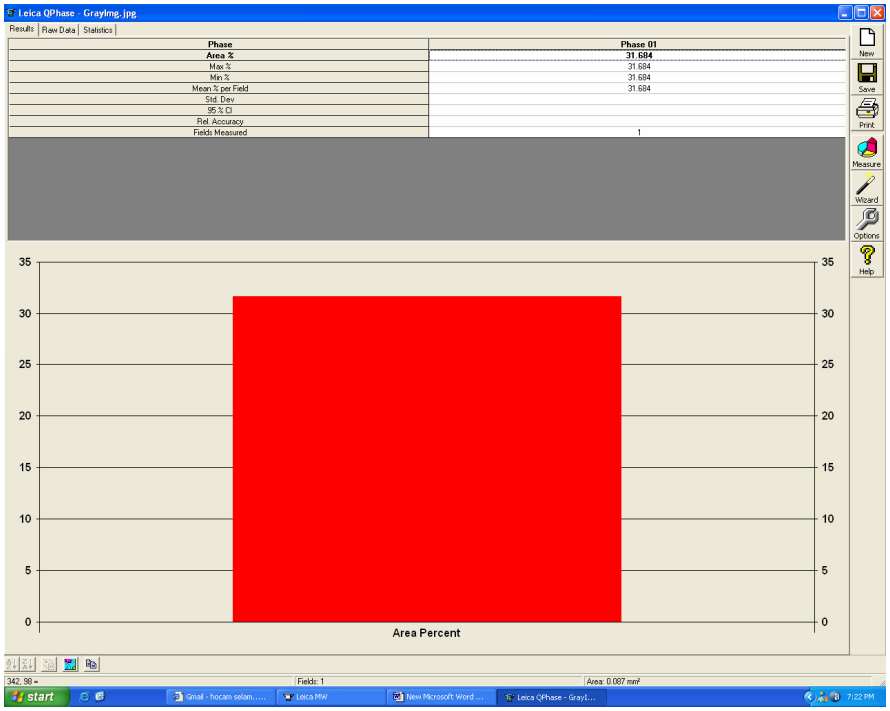
Çizelge 7.10'da işlemsiz çelik için uygulama sonucu ve üçüncü parti yazılım sonucu elde edilen değerler görülmektedir. Şekil 7.9 ve 7.10 da ise işlemsiz çeliğe uygulanan görüntü işleme aşamaları görülmektedir.

Çizelge 7.10 İşlemsiz Çelik için sonuçların karşılaştırması

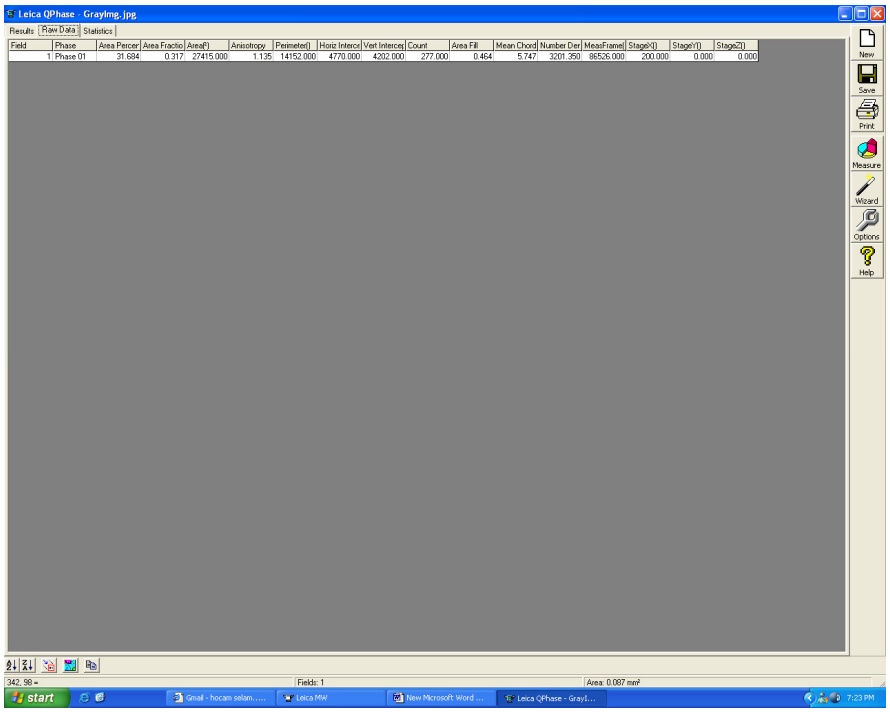
DEĞERLER	LEICA Q550 MW	Oluşturulan Uygulama yazılımı ile	FARK
piksel^2	27415	26468	947
Alan (%)	31,684	30,380	1,304



Şekil 7.9. İşlemsiz çelik görüntüsünün diğer yazılım ile işlenmesi



(a)



(b)

Şekil 7.10. (a) İşlemsiz çelik için diğer yazılımın elde ettiği sonuçlar (b) Küresel grafitli dökme demir için diğer yazılımın elde ettiği sonuçların gösterimi

8. SONUÇLAR VE ÖNERİLER

Bu çalışmada görüntü sistemi donanımları ile elde edilen görüntülerden, hazırlanan yazılım ile görüntü işleme teknikleri kullanılarak ilk bölüm uygulamada domino taşları için tanıma işlemi, sonrasında bu uygulama malzeme görüntüleri için uyarlanarak tane analizi ve resim içerisindeki yüzdesini belirleme işlemi gerçekleştirilmiştir.

Uygulamanın ilk bölümünde kurulan düzenekle, sayısal kamera kullanılarak domino görüntüleri bilgisayar ortamına aktarılmıştır. Sayısal kamera kullanımı görüntü yakalama donanımı ihtiyacını ortadan kaldırdı. Görüntüler iki farklı çözünürlükte (640x480 - 720x576, RGB 24bit) bilgisayar ortamına aktarılmıştır. Bu çözünürlükler sayısal kameranın direk olarak bu büyüklüklerde resim yakalama olanağı sunmasından dolayı tercih edilmiştir. Malzeme görüntüleri için ise optik mikroskoptan daha önce elde edilmiş hazır görüntüler kullanılmıştır. Kullanılan malzeme görüntüleri yüksek çözünürlüktedir (2088x1550 RGB 24bit).

Görüntü işleme teknikleri, görüntüler 256 gri düzeye dönüştürüldükten sonra uygulanmıştır. 256 gri düzey kullanımında ana neden domino taşları ve kullanılan malzeme görüntüleri için daha yüksek renk seviyesi kullanmanın sonuçlar üzerinde önemli bir değişiklik yapmamasıdır. Bu çalışmada renk seviyesini (miktarlama) yükseltmek yerine çözünürlüğü (örnekleme) artırmak sonuçların daha doğru olmasını sağladığı özellikle birinci bölüm uygulamada gözlenmiştir. Yapılacak uygulamaya (havacılık-uzay, tıp, malzeme, nesne tanıma) göre seçilecek renk derinliği ve çözünürlük, görüntü içerisindeki detaylara bağlı olarak farklılık göstermektedir.

Görüntü analizinde doğru sonuçların alınabilmesi için ana etken, uygun aydınlatma olarak ortaya çıktı. Bu durum uygulamanın ilk bölümünde görülmektedir. İlk bölümde aydınlatma kaynağı beyaz bir yüzeye yansıtılarak kullanılmıştır. Görüntü alınan ortamın uygun aydınlatılmaması birinci bölüm uygulamada histogram eşitleme sonrası bariz şekilde göze çarpmaktadır. Bu durum filtreleme, bölümlendirme işlemleri sonrası bazı geriplan noktalarının nesne olarak

algılanmasına, nesne noktalarının ise geriplan olarak algılanmasına yol açtı. Bu olumsuzluklar bölüm 4 de anlatılan yapısal işlemlerle düzeltilebilir. Yapısal aydınlatma özellikle görüntü işleme ile kalite kontrolü uygulamaları için önem taşımaktadır. Aydınlatma hatası sonucu oluşan gölgeler, parlamalar sonucun hassasiyetini ve doğruluğunu azaltmaktadır.

Nesne tanımada yapısal özelliklerin bölümlendirme veya görüntü analizi gibi aşamalarda kullanılması, istenen verilerin görüntüden elde edilmesini kolaylaştırmaktadır. Bu işlem için en genel örnek bölüm 5 de de belirtildiği gibi parmak izi tanımada kullanılan iz tekil noktalarıdır. Birinci bölüm uygulama için yapısal özellikler; domino taşları üzerindeki her noktanın aynı büyüklükte olması, her taştan belirli miktar olması (0 ile başlayanlardan 7 tane, 1 ile başlayanlardan 6 tane, 2 ile den 5 tane ...vb.), taşların alt veya üstünde en fazla altı nokta olabilmesi, geriplan ile taşların bariz şekilde farklı olmasıdır (bu durum histogramda iki mod olarak açığa çıkmıştır).

İkinci bölüm uygulamada malzeme görüntüleri için kullanılan marker kontrollü watershed bölümlendirmesiyle elde edilen sonuçlarda tane sınırları ile grafit sınırlarının bazı bölümlerde karışmış olduğu görüldü. Bölümlendirme metodu olarak eşikleme veya yerel en düşük gri seviyeleri kullanmak watershed bölümlendirmeden daha doğru sonuçlar verdi. Bu durumun asıl nedeni watershed bölümlendirmenin daha hassas olmasından ve her bölgeyi yerel işlemesinden kaynaklanmaktadır.

Malzeme görüntü işlemede elde edilen sonuçlar, üçüncü parti yazılımla elde edilen sonuçlarla karşılaştırıldığında hata oranın düşük (%1 civarında) olduğu görülmüştür. Hazırlanan yazılımla üçüncü parti yazılım arasındaki bu farkın ana nedeni üçüncü parti yazılımın tane sınırlarını da grafit noktaları olarak görmesinden kaynaklanmaktadır. Bunun sebebi ise grafit taneleri ile tane sınırlarının aynı gri seviye değerine sahip olmasıdır. İkinci bölüm uygulamada tane sınırları, bölümlendirme sonrası yapısal işlemler ile silinerek bu problem aşılmıştır. Bu da sonuca olumlu katkı sağlamaktadır.

İlk bölüm uygulama yazılımında görüntü elde etme prosedürleri de dahil edilerek yazılım gerçek zamanlı hale getirilebilir. Ayrıca yörünge takip algoritmaları ile hareketli nesnelerin pozisyon kestirimi de dahil edilebilir. İkinci bölüm uygulama yazılımı tane büyüklüğü belirleme, tane sayısı belirleme, yüzey pürüzlülüğü ölçme işlemleri için kullanılarak elde edilebilecek bilgilerle malzeme üzerinde uygulanacak işlemleri belirlemede kullanılabilir. Robotik uygulamaları için görüntü ve sinyal işleme algoritmaları geniş bir alanda halihazırda kullanılmaktadır. Görüntü işleme teknikleri ile nesne tanımanın yanında nesne konumlandırma, nesne takibi, esnek üretim hatlarının takibi gibi uygulamalar yapılabilir. Ayrıca tekstil, tıp, askeri alanlarda, desen tanıma, doku inceleme, hedef takibi gibi uygulamalar için görüntü işlem teknikleri kullanılabilir.

KAYNAKLAR

1. Xiang Z, Carsten K., Bernd K., "Automatic classification of defects on the product surface in grinding and polishing", ***International Journal of Machine Tools & Manufacture*** ,46: 59–69, (2006).

2. Sunil K. S., PaulW. F. , "Morphological segmentation and classification of underground pipe images", ***Machine Vision and Applications,*** 17: 21–31, (2006).

3. Li-Chen F., Cheng-Yi L., "Computer Vision Based Object Detection and recognition for Vehicle Driving", ***Proceedings of the :2001 IEEE International Conference on Robotics 8 Automation,*** Seoul-Korea , 2634-2641, (2001).

4. Mohammed B., Boualem B., "A Structural-Description-Based Vision System for Automatic Object Recognition", ***IEEE Transactıons On Systems, Man, And Cybernetıcs—Part B: Cybernetıcs,*** 27: 893-906, (1997).

5. Vujovic N., Brzakovic D., "Establishing the correspondence between control points in pairs of mammographic images", ***Image Processing, IEEE Transactions*****,** 6: 1388 – 1399, (1997).

6. Thamsanqa M., "The identification of mammalian species through the classification of hair patterns using image pattern recognition", ***ACM Special Interest Group on Computer Graphics and Interactive Techniques ACM Press*****,** **Image-based techniques,** 10:177 – 181, (2006).

7. Min Young K, "Autonomous Map-Building for Mobile Robots Mimicking A Human Visual Attention Model", Doktora Tezi, ***Korea Advanced Institute of Science and Technology,*** 1-27, (2004).

8. Christophe D., Gabriel A., Eric O., "Recognition and Localization of Solid Objects by a Monocular Vision System for Robotic Tasks", ***Intelligent Robots and Systems '94. 'Advanced Robotic Systems and the Real World', IROS '94. Proceedings of the IEEE/RSJ/GI Int.Conf.,*** 3: 2007-2014, (1994).

9. Adnan A. Y. Mustafa, "Boundary Signature Matching for Object Recognition", ***Vision Interface Annual Conference Ottawa*****,** 1-9, (2001).

10. Alessandro L., Cosimo D., Francesco B., "A shadow elimination approach in video-surveillance context", ***Pattern Recognition Letters*****,** 27: 345-355, (2006).

11. BAYRAKTAR H., "Development Of A Stereo Vısıon System For An Industrıal Robot", Yüksek Lisans Tezi, ***Middle East Technıcal Unıvercıty***, Ankara, 1-120, (2004).

12. Arıcan Z., "Vısıon-Based Robot Localızatıon Usıng Artıfıcıal And Natural Landmarks", Yüksek Lisans Tezi, ***Middle East Technıcal Unıvercıty***, Ankara, 16-47, (2004).

13. Fumi H., Masayoshi H., Shinji K., Tsugito M., "A Stereo Vision System for Position Measurement and Recognition in an Autonomous Robotic System for Carrying Food Trays", ***Systems, Man, and Cybernetics 1999. IEEE SMC '99 Conference Proceedings*** , 661-668, (1999).

14. Önder M., "Face Detection In Active Robot Vision",Yüksek Lisans Tezi, ***Middle East Technıcal Unıvercıty***, Ankara, 5-70, (2004).

15. Şeran O., "Visually Guided Robotic Assembly", Yüksek Lisans Tezi, ***Middle East Technıcal Unıvercıty***, Ankara, 8-94, (2004).

16. Ünal Y., "Seri üretim hatlarında görüntü işlemeyle kalite kontrolü", Yüksek Lisans Tezi, ***Gazi Üniversitesi Fen Bilimleri Enstitüsü Makina Mühendisliği***, Ankara, 1-100, (1998).

17. Yaman K., "Görüntü işleme yönteminin Ankara hızlı raylı sistem güzergahında sefer aralıklarının optimizasyonuna yönelik olarak incelenmesi", Yüksek Lisans Tezi, ***Gazi Üniversitesi Fen Bilimleri Enstitüsü Makina Mühendisliği***, Ankara, 30-80, (2001).

18. Gonzalez R.C., Woods R.E., "Digital Image Processing, Second Edition", ***Prentice-Hall,*** New Jersey, 25-344,519-732 , (2002).

19. Gonzalez R.C., Woods R.E., Steven L.E., "Digital İmage Processing Using MATLAB", ***Prentice-Hall,*** New Jersey, 65-107,334-474 , (2004).

20. Baxes G.A., "Digital Image Processing Principles and Applications", ***John Wiley & Sons Inc.,*** New York, 13-20, 330-397, (1994).

21. Wojnar L., "Image Analysis Application in Materials Engineering", ***CRC Press,*** New York, 8-151, (1999).

22. Morris T., "Computer Vision and Image Processing", ***Palgrave Macmillan,*** New York, 43-79, 92-136, (2004).

23. Castleman K.R., "Digital Image Processing", ***Prentice Hall,*** New Jersey, 71-86, 447-538, (1996).

24. Zhang. Z, "A flexible new technique for camera calibration", ***Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on,*** Kerkyra, 1: 666-673, (1999)

25. Bouguet J.Y., “Visual methods for three-dimensional modeling”, Doktora Tezi, ***California Institute of Technology,*** Pasadena, 20-42, 192-236, (1999)

EKLER

EK-1 Kullanılan donanımlar ve özellikleri

1.1. CCD kamera

Marka-Model	: Panasonic NV-GS75
Lens	
Görüntü algılayıcısı	: 1/6" CCD x 3 (420k x 3 pixels)
F değeri	: F1.8 (16/9 da) / F2.8 (4/3 de)
Optik yakınlaştırma	: 10x
Odak uzaklığı	: 2.45 - 24.5mm
Filtre çapı	: 37mm
Lens Tipi	: Leica Dicomar Lens
Camera	
Sabit görüntü kayıt	: 1.7 Megapixel
Sabit resim çekme	: mevcut
Minimum aydınlatma ihtiyaçı	: 1 Lux (Advanced Magic Pix)
Yüksek hız kesimi	: Teyp Modu;1/60 - 1/8,000 Kart Modu; 1/30 - 1/500
Iris	: otomatik veya elle
Saf renk işlemcisi	: gelişmiş
Eş zamanlı video ve resim kaydı	: mevcut
Sahte renkler	: siyah veya beyaz
Record ReviewYes	
Resim Dengeleyicisi	:(D-EIS)
Sayısal yakınlaştırma	:500x
Web Kamera fonksiyonu	: mevcut
Kaydetme ve yürütme	
Kayıt modu	: SP/LP
Anlık görüntü alma	: JPEG1.7Mbit 640 x 480(VGA)

EK-1 (Devam) Kullanılan donanımlar ve özellikleri

Giriş-Çıkış

S-Video girişi	: yok
S-Video çıkışı	: mevcut
Mini DV giriş/çıkış	: IEEE1394 giriş/çıkış
USB Port	: 2.0
Analog giriş/çıkış	: sadece çıkış
Microphone (stereo mini)	: mevcut

Genel

Güç beslemesi	: DC 7.2V
Güç tüketimi	: 3.6W (EVF) - 3.9W (LCD)

1.2. IEEE 1394 arayüz

1394 arabiriminin özellikleri

1394 arabiriminin özellikleri aşağıdaki gibi özetlenebilir. Her özelliğin kısa bir açıklaması da verilmiştir.

- Anında Takma/Çıkarma, PnP: 1394 aygıtları, aygıt takma/çıkarma işlemlerini tespit edebilir ve konfigürasyonunu otomatik olarak yapar.
- Seri Veri Yolu: Veri aktarımı için iki farklı sinyal çifti kullanılmaktadır.
- Eşsüreli Aktarım: Kamera gibi video uygulamalarında, sabit bit oranına yakın aktarım gereklidir. Eşsüreli aktarım, veri kesinliği yerine ayrılmış aktarım oranı sözü verir.
- Direk Aktarım: Aynı 1394 veri yolunda bulunan aygıtların birbirleri ile iletişim kurmaları için sunucu sistemi etkilememe şansları vardır.
- Kablo Gücü: Kendi elektriğini sağlayamayan aygıtlar için, kablo gücü, 1394'ün altı pinlik kablosu aracılığıyla mümkündür.

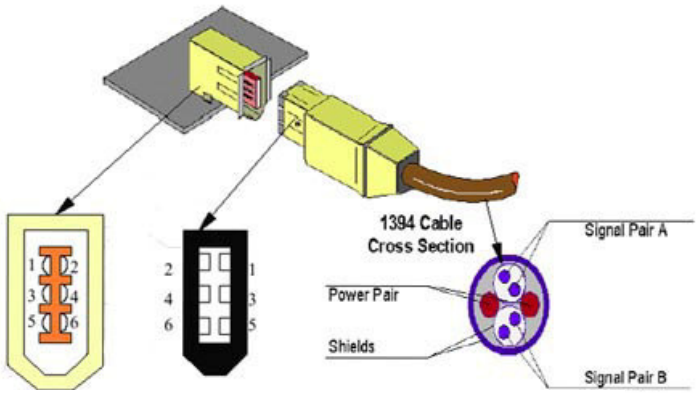
EK-1 (Devam) Kullanılan donanımlar ve özellikleri

- Ölçeklenebilir Performans: 1394 aygıtlarının aktarım hızı, 1394a'ya göre azami 100Mbps, 200Mbps veya 400Mbps olabilir.

1394 Bağlantısı

İki 1394 aygıtını bağlayan kablonun iki tipi bulunmaktadır: 6 iletken içeren ve 4 iletken içeren. Altı pinlik bağlantı ve kablolar IEEE1394-1995 özelliklerinde, dört pinlik sürüm ise 1394a ekinde tanımlanmıştır. Altı pinlik 1394 kablo veya bağlantısı Şekil 1.1'de görülmektedir. Kablo gücü için iki iletken, sinyal aktarımı için ise iki çift iletken barındırır. Pin tanımları aşağıdaki gibidir:

- Pin 1.VP: Kablo Gücü
- Pin 2.VG: Kablo Toprağı
- Pin 3 ve 4. TPB* ve TPB: Twisted Pair B. Iletimde farklı veri.
- Pin 5 ve 6. TPA* ve TPA: Twisted Pair A. Alımda farklı veri.

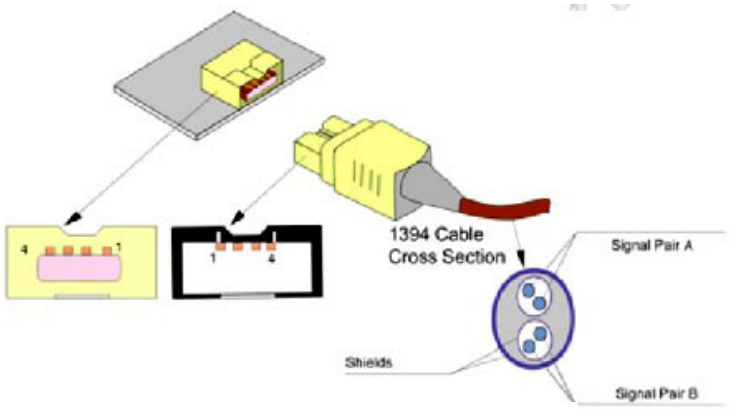


Şekil 1.1. IEEE 1394 6 pin kablo ve konnektör

Dört pinlik kablo veya bağlanti içinse, kablo gücü ve toprak kaldırılmıştır. Dört pinlik bir 1394 bağlantısı için TPB*, TPB, TPA* ve TPA'nin pin numaraları sırasıyla 1, 2, 3 ve 4 şeklindedir.

EK-1 (Devam) Kullanılan donanımlar ve özellikleri

Daha küçük dört pinlik kablo, bataryayla çalışan aygıtlar için uygundur. Şekil 1.2, dört pinlik 1394 kablosunu ve baglantılarını göstermektedir.



Bir 1394 kablosu için, bir uçta 6 pinlik çıkış, bir ucunda ise 4 pinlik çıkış olması kablo gücü ve toprak için olan bağlantılar kaldırılarak mümkündür. Dikkat edilmesi gereken nokta, özelliklerine göre, bir 1394 kablosunun azami boyu 4.5 metredir.

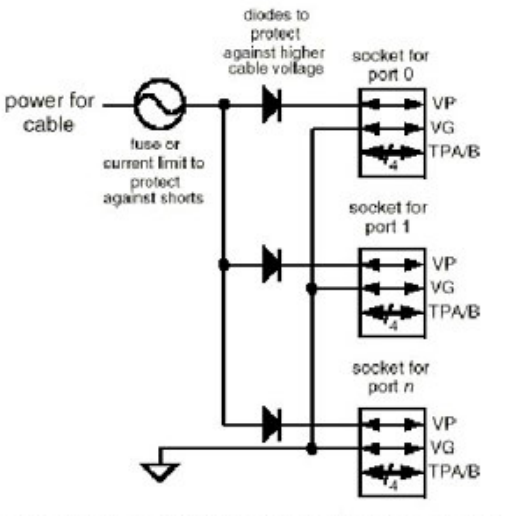
Güç Dağıtımı

Bir 1394 aygıtı, kendi gücünü sağlıyor ya da gücü kablodan alabilir. 1394'te tanımlı 8 (üç ikili rakam) güç sınıfı vardır. Bunlardan üçü 1394 sistemlerdeki ana kablo güç kaynaklarıdır. 0 haricindeki güç sınıfları, kendi kendilerine akış yönünde güç sunmazlar. Onun yerine, gelen gücü tekrar gönderirler. Güç sınıfı 0 aygıtı ise, gelen gücü yok etmez, ama bu gücü de iletmez. Genel olarak, 1394a güç sunucuları aşağıdaki gereksinimleri karşılayacak şekilde tasarlanmalıdır:

- Port başına azami çıkış akımı: 1.5A
- Azami çıkış voltajı: 30V (1394a 3.0'a göre)
- Azami çıkış dalgası: 100mV

EK-1 (Devam) Kullanılan donanımlar ve özellikleri

Güç sınıfı 1 ile 3 arasındaki aygıtlar için olan gereksinimler, diğerlerinden daha sıkıdır. Güç sınıfı 1-3 olan aygıtlar için asgari çıkış voltajı 20V iken diğerleri için 8V'tur. Güç sınıfı 1-3 olan aygıtlar için, asgari güç dağıtma kapasitesi, sırasıyla 15W, 30W veya 45W'tır. 1394 güç sağlama ortamlarında, farklı güç kaynakları için farklı çıkış voltajlarının bağlanması mümkündür. Az voltajlı güç sağlayıcılarını, yüksek voltajlıların yok etmesinden korumak için, izolasyon diyodu gereklidir. Şekil 3, 1394 düğümleri için bir koruma örneği göstermektedir.



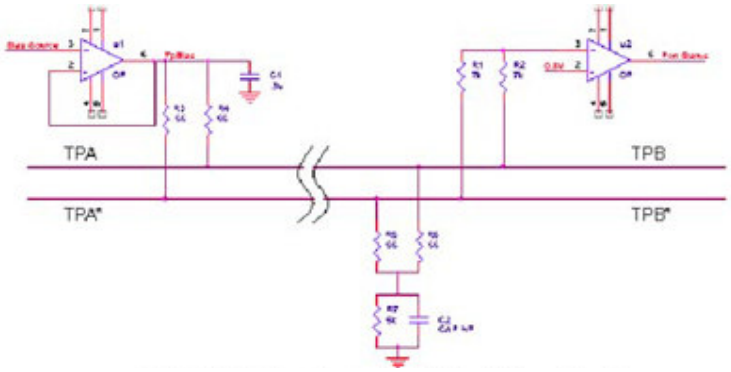
Şekil 1.3. Güç sınıfları için güç düğüm arabirim örneği

Aygıt Takma/Çıkarma Tespiti

1394, anında takma/çıkarma özelliğini desteklediğinden, aygıt takma ve çıkarmayı tespit edebilen bir mekanizma gereklidir. Şekil 1.4, tespit mekanizmasını örnekler. İlk olarak, şeklin sağ tarafına bakınız; orada TPB ve TPB* çiftine bağlı devre bulunmaktadır. Sol taraftaki devrenin bulunmaması halinde (örneğin hiçbir aygıt yokken), U2 işletimsel yükseltecin çıkışındaki "Port_Status" düşecektir; çünkü pozitif giriş terminali topraklanacaktır.

EK-1 (Devam) Kullanılan donanımlar ve özellikleri

Bir aygıt takıldığında ise (yani sol taraftaki devre uygulandığında), U2'nin pozitif girişindeki voltaj 0.8V'tan yüksek olacaktır. Bu nedenle, bir 1394 aygıtı takıldıktan sonra "Port_Status" daha yüksek bir voltaja ulaşacaktır.



Şekil 1.4. 1394 aygıtın ekle-kaldır tespit devre örneği

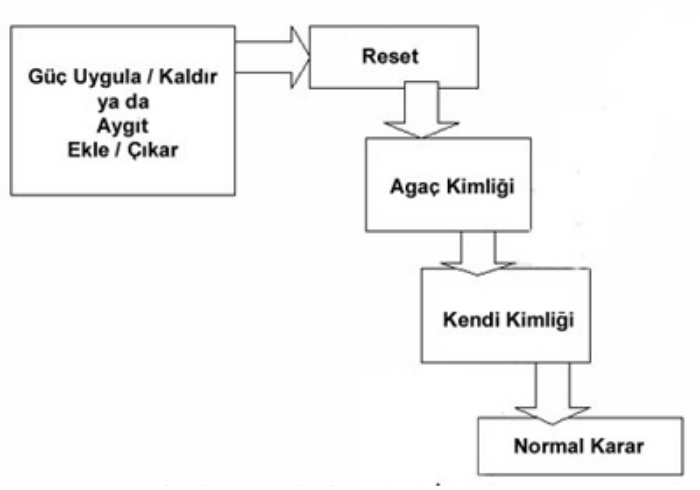
1394 Konfigürasyon İşlemi

Sistemi her yeniden başlatışınızda ya da bir aygıt takıldığında/çıkartıldığında, 1394 veri yolu tekrar ayarlanmak zorundadır. Konfigürasyon işlemi Şekil 1.5' de gösterilmiştir. Aygıtın takıldığı/çıkartıldığı düğüm tarafından bir reset sinyali gönderilir ve diğer bütün düğümler bu sinyali kalan kablo parçalarına iletir. Aygıt takma/çıkarma tespit mekanizması bir önceki konuda anlatılmıştır. Reset sinyalinin gönderilmesinden sonra, her düğümün baba düğümünü ve çocuk düğümlerini belirlemesi için bir ağaç tanımlama işlemi gerçekleştirilir; son olarak da, kök düğüm belirlenir. Yani kendi kendini tanımlama, aşağıdaki işlemleri yapmak için başlatılır.

- Her düğümün kendine ait bir Fiziksel ID'ye sahip olması.
- Komşu düğümler arasında aktarım ve güç yeteneklerinin değiş tokuş edilmesi.
- Topolojinin yayınlanması

EK-1 (Devam) Kullanılan donanımlar ve özellikleri

Kendi kendini tanımlama, 1394 konfigürasyon işlemindeki son adımdır. Kendi kendini tanımlamadan hemen sonra, veri yolu normal duruma döner ve aktarımlar için hazır bekler.



Şekil 1.5. 1394 konfigürasyon işlemi

1394 Aktarım Tipleri

1394 veri yolunda iki tip aktarım mümkündür: Asenkron aktarım ve eşsüreli aktarım. Eşsüreli aktarım, sabit veri oranları gerektirirken, asenkron aktarım, veri iletişiminde belli düzeltmeler gerektirir. Bir 1394 veri yolundaki bant genişliği kullanımı, 125µs'lik periyotlara bölünmüştür. Her dönüşte, yaklaşık %80 (100µs) eşsüreli aktarımlar için ayrılabilirken, asgari %20 (25µs) de asenkron aktarım için garanti edilebilir. Eşsüreli aktarım özellikle dijital video ve ses uygulamaları için önemlidir.

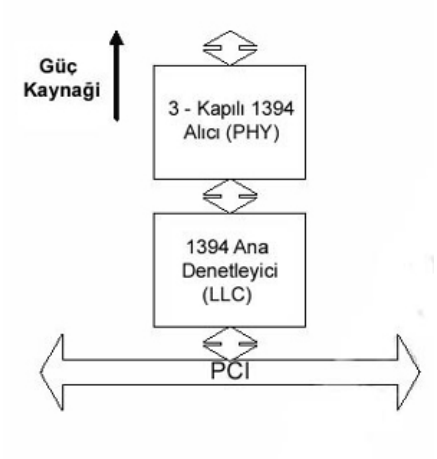
Pratik Tanımlama

Donanımsal bakış açısında, 1394 düğümünün pratik olarak tanımlanması iki çipten oluşur: PHY (Physical Layer) ve LLC (Link Layer Controller). Şekil 1.6'da da gösterildiği gibi, PHY, harici 1394 aygıtları için bağlantılar sunar ve

EK-1 (Devam) Kullanılan donanımlar ve özellikleri

LLC de, sistemde PCI-1394 köprüsü görevine sahiptir. PHY, 1394 veri yolu başlatımı, NRZI kodlanması, kod çözülmesi ve veri senkronizasyonu içindir.

LLC ise 1394 protokol yığınında bağlantı ve iletim katmanı olarak çalışır. Sonuç olarak, "okuma" ve "yazma" komutları, LLC'de alınır, paketlenir veya paketi açılır.



Şekil 1.6. 1394 aygıtın harici bağlantısı

1.3. Bilgisayar

Kullanılan bilgisayarın teknik özellikleri:

İşlemci: AMD Athlon 64 3.0Ghz
RAM : 512 MB DDR 400
Anakart : MSI Nforce Neo3
Ekran Kartı : ATI Rodeon X600 128 MB RAM DVI+TV-out
HDD : 80 GB SATA
Kasa : 350 W Aopen
Monitör : 17” Samsung DF

EK-1 (Devam) Kullanılan Donanımlar ve özellikleri

Giriş ve Çıkışlar:

USB 2.0 : 6 adet

LAN : GigabitEthernet

IEEE1394 : mevcut

İşletim sistemi ve diğer yazılımlar:

Windows XP SP2, Panasonic Motion Capture, MATLAB 7.1, IPT 5.3, DIPUM 4.3

EK-2 Kamera kalibrasyonu

2.1. Kamera Kalibrasyonu

Kamera kalibrasyonu iki boyutlu (2D) görüntüden metrik bilgileri çıkarmak için üç boyutlu (3D) görüntü işlemede zorunlu bir aşamadır. Kamera kalibrasyon teknikleri kabaca 2 grupta toplanabilir [24].

Fotogrametrik kalibrasyon: Bu teknik, geometrik bilgileri 3 boyutlu olarak büyük bir hassasiyetle bilinen bir kalibrasyon nesnesinin gözlenmesi şeklinde uygulanır. Kalibrasyon nesnesi genelde birbirlerine ortagonal iki veya üç yüzeyden oluşur. Elde edilen sonuçlar oldukça verimlidir. Fakat bu teknik için pahalı kalibrasyon aparatları ve laboratuvar düzenekleri gereklidir.

Ekstra kurulum gerektirmeyen kalibrasyon (self-calibration): Bu teknikte bir kalibrasyon nesnesi yoktur. Sadece kamera sabit bir sahneyi değişik açılardan çeker. Çekilen sahnenin sabitliği, bir kamera yer değişiminden tek, görüntü bilgisini kullanarak kamera dahili parametrelerinde genel iki kısıtlayıcı sunar. Böylece, eğer görüntüler aynı kameradan sabit dahili ve harici parametrelerle alınmışsa, bize , üç boyutlu yapıyı yeniden elde etmede kullanılan, dahili ve harici parametreleri bulmak için, en az üç görüntü arasındaki ilişki yeterli olur.

Dahili görüntü düzeltme ile iki basamaklı kamera kalibrasyon prosedürü [24] uygulanmıştır. Kamera kalibrasyon yazılımı ile kameranın dahili ve harici parametreleri elde edilmiştir. Bu uygulamada domino taşlarının metrik ve geometrik bilgileri elde edilmemiştir.

Gerçek (3 boyutlu) koordinatları $M = [X, Y, Z]^T$ olan bir noktanın görüntü (2 boyutlu) koordinatları $m = [u, v]^T$ olsun. M ve m'in son elementlerine 1 koyarak artırırsak $\tilde{M} = [X, Y, Z, 1]^T$ ve $\tilde{m} = [u, v]^T$ elde edilir. Pinhole kamera modeli kullanarak $\tilde{m}$ ve $\tilde{M}$ arasındaki ilişki Eş.2.1 dedir.

EK-2 (Devamı) Kamera kalibrasyonu

$$\tilde{m} = A[R\,t]\tilde{M} \tag{2.1}$$

Burada A kameranın dahili parametreler matrisi, R rotasyon matrisi ve t transformasyon matrisidir. R ve t, görüntülenen nesnenin ifade edildiği mutlak koordinat sistemini, görüntüleyen merkezli olan kamera koordinat sistemine dönüştürmek için kullanılır. Bu parametreler harici parametreler olarak bilinir. Kamera dahili parametrelerini ifade eden A matrisi aşağıdaki gibidir.

$$A = \begin{bmatrix} \alpha & c & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.2}$$

Burada (u_0,v_0) resim düzleminin merkez noktasını, α ve β görüntüdeki u ve v eksenlerindeki ölçek faktörünü, c de u ve v deki çarpıklığı tanımlamaktadır. Nesne düzleminin mutlak koordinat sistemine göre Z=0 da olduğu kabul edilirse Eş.2.1 'den:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = A\begin{bmatrix} r_1 & r_2 & r_3 & t \end{bmatrix}\begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix} = A\begin{bmatrix} r_1 & r_2 & t \end{bmatrix}\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \tag{2.3}$$

Elde edilir. Burada r_i , R dönme matrisinin i'inci sütununu göstermektedir. Z genelde 0'a eşit olduğundan $\tilde{M} = [X,Y,Z,1]^T$ indirgenerek $\tilde{M} = [X,Y,1]^T$ şekline dönüşür. Bu durumda gösterim aynılığını sağlamak için görüntü noktasıyla sahne noktası arasındaki ilişki Eş. 2.4'deki gibidir. Burada $H=A[r_1\ r_2\ t]$ dir. $H=[h_1\ h_2\ h_3]$ şeklinde gösterilirse Eş. 2.5 elde edilir.

$$\tilde{m} = H\tilde{M} \tag{2.4}$$

EK-2 (Devamı) Kamera kalibrasyonu

$$\begin{bmatrix} h_1 & h_2 & h_3 \end{bmatrix} = \lambda A \begin{bmatrix} r_1 & r_2 & t \end{bmatrix} \tag{2.5}$$

Burada λ rasgele bir sabittir. r_1 ve r_2 ortanormal olduğunda aşağıdaki sınırlama oluşur.

$$r_1^T . r_2 = h_1^T . (A^{-1})^T . A^{-1} . h_2 = 0 \tag{2.6}$$

$$r_1^T . r_2 = r_2^T . r_2 \rightarrow h_1^T . (A^{-1})^T . A^{-1} . h_1 = h_2^T . (A^{-1})^T . A^{-1} . h_2 \tag{2.7}$$

Burada $r_1 . = A^{-1} . h_1^T$ ve $r_2 = A^{-1} . h_2$ dir.

$$B = (A^{-1})^T . A^{-1} \equiv \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{23} & B_{23} \\ B_{31} & B_{32} & B_{33} \end{bmatrix} \tag{2.8}$$

şeklinde yazılırsa B aşağıdaki gibidir.

$$B = \begin{bmatrix} \frac{1}{\alpha^2} & \frac{-c}{\alpha^2 \beta} & \frac{cv_0 - u_0 \beta}{\alpha^2 \beta} \\ \frac{-c}{\alpha^2 \beta} & \frac{c^2}{\alpha^2 \beta^2} + \frac{1}{\beta^2} & \frac{-c(cv_0 - u_0 \beta)}{\alpha^2 \beta^2} - \frac{v_0}{\beta^2} \\ \frac{cv_0 - u_0 \beta}{\alpha^2 \beta} & \frac{-c(cv_0 - u_0 \beta)}{\alpha^2 \beta^2} - \frac{v_0}{\beta^2} & \frac{(cv_0 - u_0 \beta)^2}{\alpha^2 \beta^2} + \frac{v_0^2}{\beta^2} + 1 \end{bmatrix} \tag{2.9}$$

B simetrik olduğundan altı boyutlu vektör olarak tanımlanabilir.

$$b = [B_{11}, B_{12}, B_{22}, B_{13}, B_{23}, B_{33}]^T \tag{2.10}$$

H'ın *i*'inci sütün vektörü h_i şeklinde gösterelim. Bu durumda $h_i = [h_{i1}, h_{i2}, h_{i3}]^T$ olur ve aşağıdaki ilişki kurulabilir.

EK-2 (Devamı) Kamera kalibrasyonu

$$h_i^T B h_j = v_{ij}^T b \tag{2.11}$$

Burada $vij = \left[h_{i1}h_{j1}, h_{i1}h_{j2} + h_{i2}h_{j1}, h_{i2}h_{j2}, h_{i3}h_{j1} + h_{i1}h_{j3}, h_{i3}h_{j2} + h_{i2}h_{j3}, h_{i3}h_{j3}\right]^T$ dir. Eş.2.6 ve Eş.2.7 'deki iki temel sınırlandırma kullanılarak aşağıdaki homojen denklem yazılabilir.

$$\begin{bmatrix} v_{12}^T \\ (v_{11} - v_{22}) \end{bmatrix} b = 0 \tag{2.12}$$

Örnek düzleminin n tane görüntüsü için *Vb=0* şekilde yazılabilir. Burada V, $2n \times 6$ matristir ve çözümü $V^T V$ nin ilgili en küçük özdeğerine göre elde edilen özvektörüdür. b elde edildikten sonra kamera öznitel matrisi *A,* Eş.2.13 ile elde edilir.

$$\begin{aligned}
u_0 &= cv_0 / \alpha - \beta_{13}\alpha^2 / \lambda \\
v_0 &= (B_{12}B_{13} - B_{11}B_{23}) / (B_{11}B_{22} - B_{12}^2) \\
\lambda &= B_{33} - \left[B_{13}^2 + v_0(B_{12}B_{13} - B_{11}B_{23})\right] / B_{11} \\
\alpha &= \sqrt{\lambda / B_{11}} \\
\beta &= \sqrt{\lambda B_{11} / (B_{11}B_{22} - B_{12}^{2)}} \\
c &= -B_{12}\alpha^2 \beta / \lambda
\end{aligned} \tag{2.13}$$

A matrisi belirlendikten sonra, herbir resim için harici parametreler Eş.2.4 kullanılarak belirlenebilir.

$$\begin{aligned}
r_1 &= \lambda A^{-1} h_1 \\
r_2 &= \lambda A^{-1} h_2 \\
r_3 &= r_1 \times r_2 \\
t &= \lambda A^{-1} h_3 \\
\lambda &= 1 / \left\| A^{-1} h_1 \right\| = 1 / \left\| A^{-1} h_2 \right\|
\end{aligned} \tag{2.15}$$

EK-2 (Devamı) Kamera kalibrasyonu

Bu çözüm maksimum olasılık kestirimi ile rafine edilebilir. *n* tane kalibrasyon örneği görüntüsü ve, örnek görüntü üzerindeki *m* tane kontrol noktası için Maksimum Olasılık Kesritimi aşağıdaki fonksiyonun minimize edilmesi ile elde edilir.

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\left\|m_{ij}-\hat{m}(A,R_i,t_i,M_j)\right\|^2 \tag{2.16}$$

Burada $\hat{m}(A,R_i,t_i,M_j)$ *Mj* noktasının *i*'inci görüntüdeki Eş.2.4 ile elde edilen çıkarımıdır. Eş.2.16 Levenberg-Marquardt algoritması ile çözülebilen nonlineer optimizasyon problemidir.

Pinhole kamera modeli, gerçek kamera modelinin bir yaklaşımı olduğundan, yüksek doğruluk gerektiğinde mercek bozulmalarının (ana mercek bozulmaları olan radyal ve tanjantsal bozulmaları) bozulma katsayıları kullanarak düzeltilmesi gerekir. (u,v)

Bozulmasız, $(\breve{u},\breve{v})$ bozulmalı resim düzlemi (CCD) koordinatları, aynı şekilde (x,y) bozulmasız, $(\breve{x},\breve{y})$ bozulmalı görüntü (resim) koordinatları olsun. Bu durumda radyal bozulma aşağıdaki şekilde yazılabilir.

$$\delta x^r = x(k_1r^2+k_2r^2+.....) \tag{2.17}$$

$$\delta y^r = y(k_1r^2+k_2r^2+.....) \tag{2.18}$$

Burada k_1 ve k_2 radyal bozulma katsayıları, $r^2=x^2+y^2$ dir. Tanjantsal bozulma ise aşağıdaki şekilde yazılabilir.

$$\delta x^t = 2p_1xy+p_2(2x^2+y^2) \tag{2.19}$$

$$\delta y^t = p_1(x^2+2y^2)+2p_2xy \tag{2.20}$$

EK-2 (Devamı) Kamera kalibrasyonu

Bu durumda $(\breve{x}, \breve{y})$ aşağıdaki biçimde elde edilir.

$$\breve{x} = x + \delta x^r + \delta x^t \tag{2.21}$$

$$\breve{y} = y + \delta y^r + \delta y^t \tag{2.22}$$

Benzer şekilde $(\breve{u}, \breve{v})$ aşağıdaki biçimde elde edilir.

$$\breve{u} = u_0 + \alpha\, \breve{x} + c\breve{y} \tag{2.23}$$

$$\breve{v} = v_0 + \beta\, \breve{y} \tag{2.24}$$

Parametrelerin komple seti, Eş.*xxx*' e benzer olarak aşağıdaki fonksiyonun Maximum olasılık kestiriminin minimize edilmesi ile elde edilebilir.

$$\sum_{i=1}^{n} \sum_{j=1}^{m} \left\| m_{ij} - \hat{m}(A, k_1, k_2, p_1, p_2, R_i, t_i, M_j) \right\|^2 \tag{2.25}$$

Burada $\hat{m}(A, k_1, k_2, p_1, p_2, R_i, t_i, M_j)$ *Mj* noktasının *i*'inci görüntüdeki Eş.2.4 ile elde edilen çıkarımıdır [11].

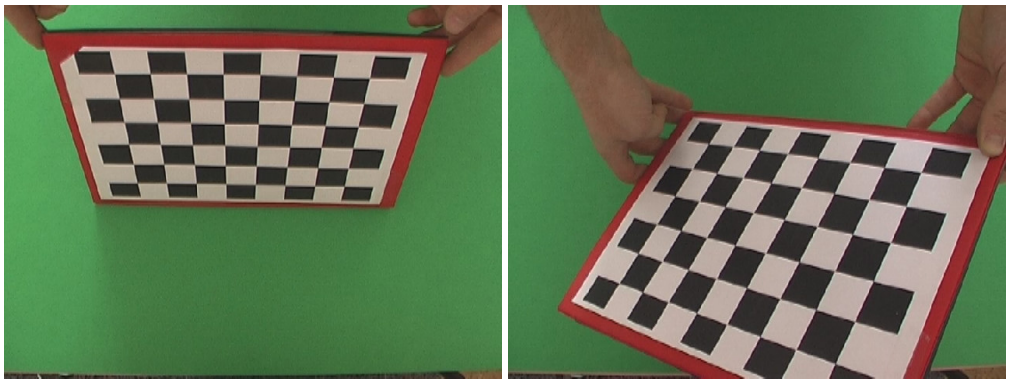
2.2. Kamera Kalibrasyon Uygulaması

Kamera dahili ve harici parametrelerini belirlemek için lazer yazıcı ile bastırılmış 3x3 cm lik karelerden oluşan satranç tahtası örneği kullanılmıştır.

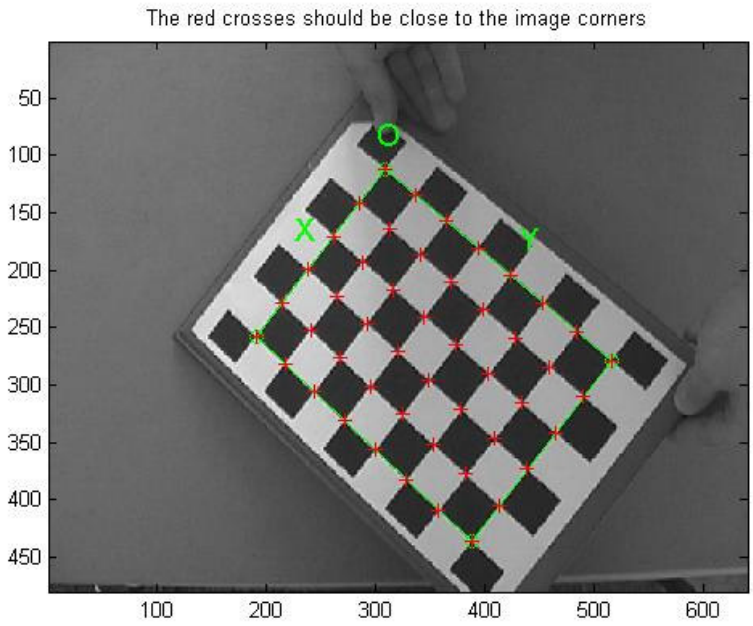
Örnek düz bir yüzeye sabitlendikten sonra değişik uzaklık ve doğrultularda kamera ile 20 farklı görüntü alınmıştır. Bu görüntülerden bazıları Resim *i*' de dir. Daha sonra dört köşe noktası seçilerek köşe çıkarımı uygulanmıştır. İlk olarak radyal, tanjantsal bozulma parametreleri dahil edilmeden kapalı formda çözümle hesaplanmış,

EK-2 (Devamı) Kamera kalibrasyonu

ardından nonlineer optimizasyon ile reprojeksiyon hataları uygulama yazılımı ile tüm kalibrasyon parametrelerinden minimize edilmiştir. Elde edilen sonuçlar aşağıtaki gibidir.

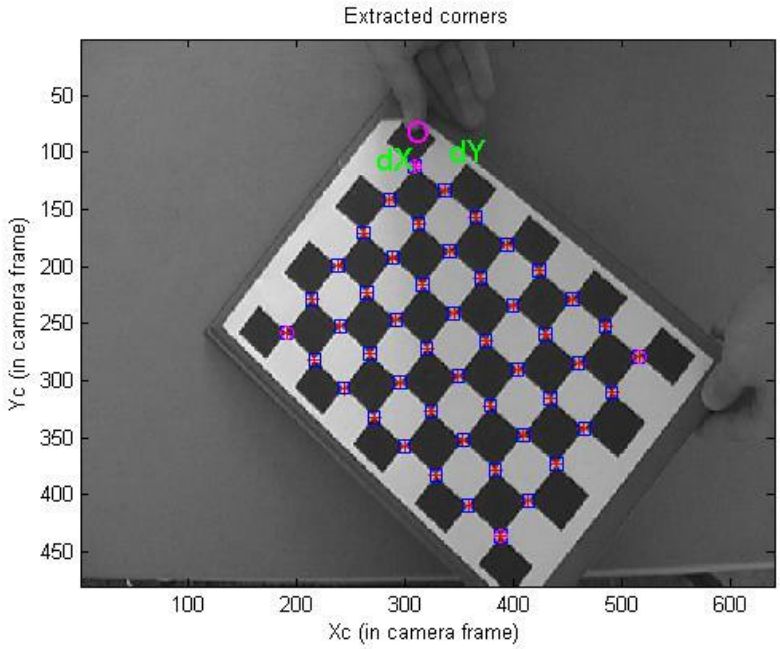


Resim 2.1 Kullanılan kalibrasyon görüntülerinden ikisi

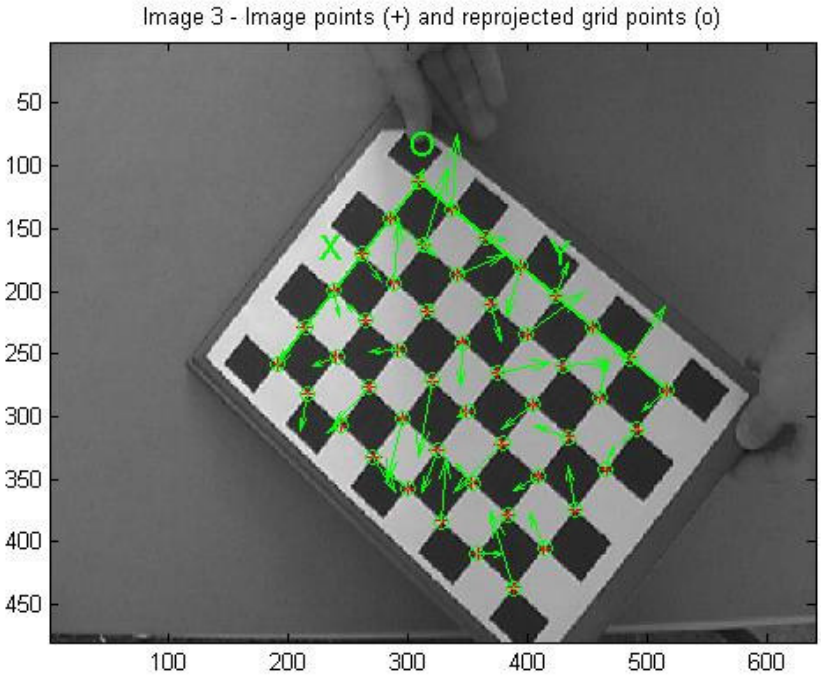


Resim 2.2. Köşe seçimi (kırmızı artılar köşeleri göstermektedir).

EK-2 (Devamı) Kamera kalibrasyonu

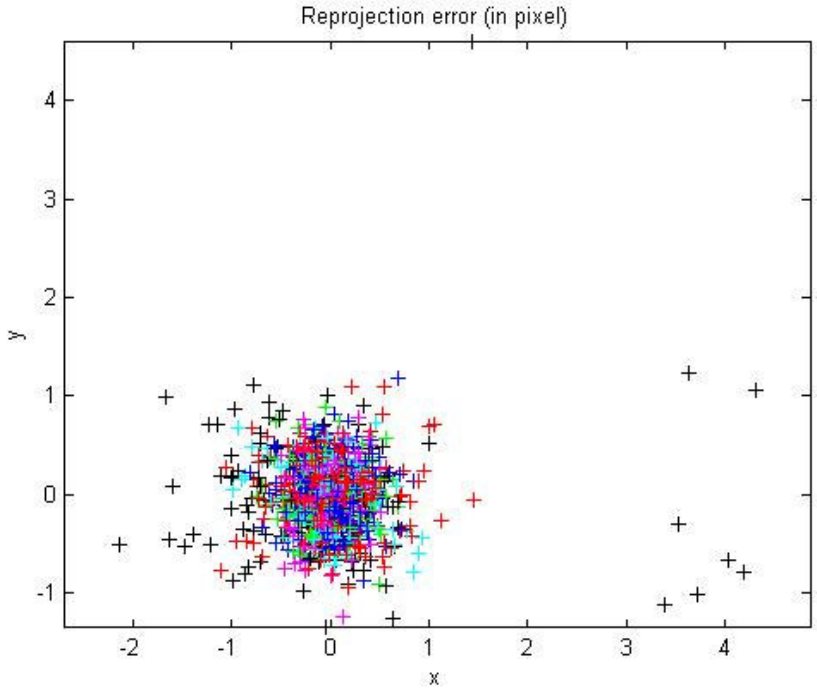


Resim 2.3. Köşe çıkarımı (mavi kareler).

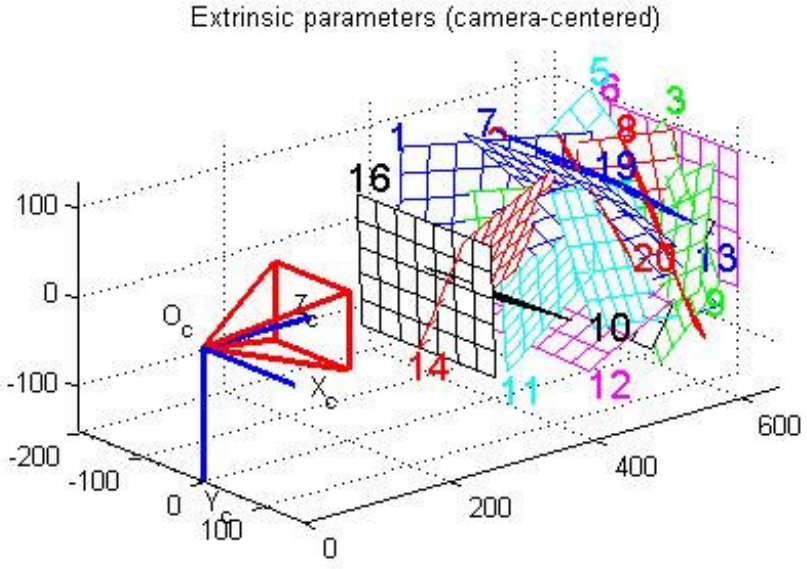


Resim 2.4. Reprojeksiyon düzeltmesi (yeşil oklar).

EK-2 (Devamı) Kamera kalibrasyonu

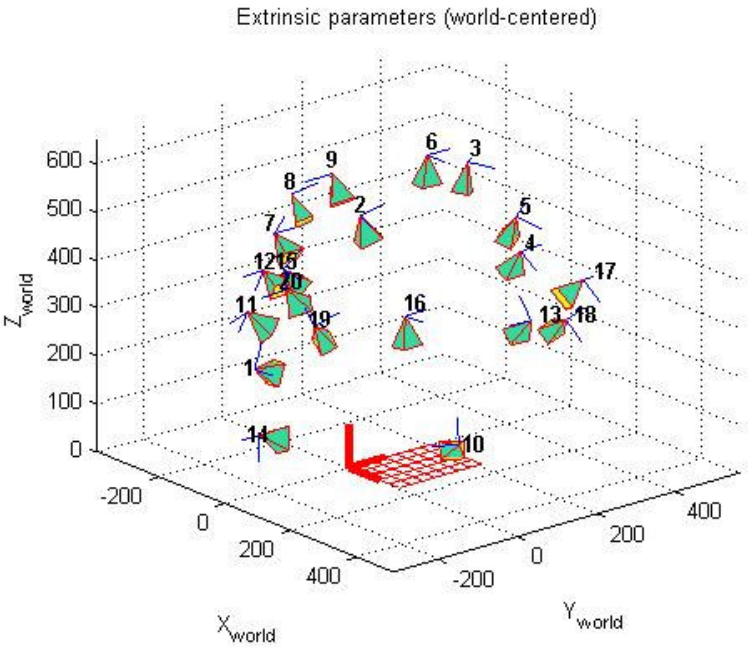


Resim 2.5. Kalibrasyon görüntülerinin reprojeksiyon hatalari

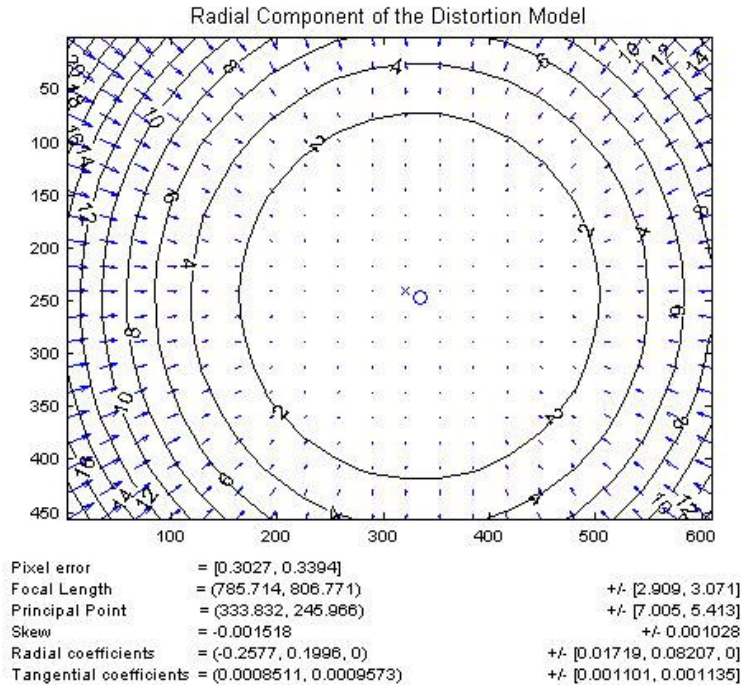


Resim 2.6. Kalibrasyon görüntülerinin kameraya göre gridleri

EK-2 (Devamı) Kamera kalibrasyonu

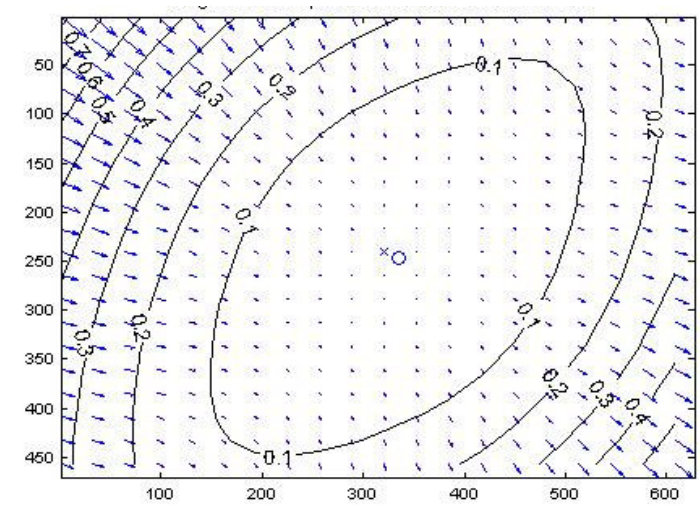


Resim 2.7. Kameranın kalibrasyon örneğine göre mutlak koordinatlarda gösterimi

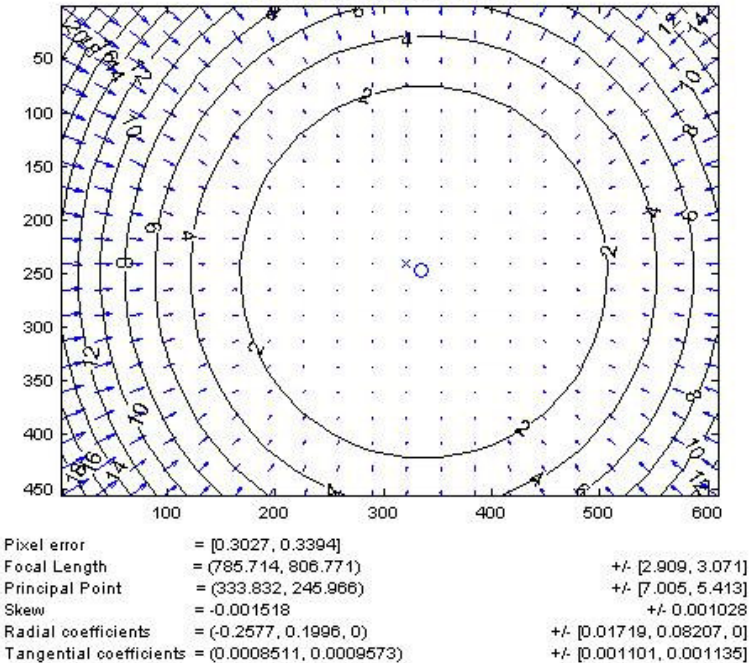


Resim 2.8. Bozulma modelinin radyal bileşeni ve elde edilen değerler.

EK-2 (Devamı) Kamera kalibrasyonu



Resim 2.9 Bozulma modelinin tanjantsal bileşeni ve elde edilen değerler.



Resim 2.10. Komple Bozulma modeli ve elde edilen değerler.

EK-3 Birinci bölüm uygulamada kullanılan kaynak kodlar

menu.m

```
function Ana_Menu,

cell_list = { };

%-------- Menu secenekleri ----------%


fig_number = 1;

title_figure = 'Tez - Hikmet BAL';

cell_list{1,1} = {'Dosya aç','dosya_ac;'};
cell_list{2,1} = {'Histogram işleme','intensity_trans;'};
cell_list{3,1} = {'Filtreleme','spatial_filtreleme;'};
cell_list{4,1} = {'Segmentasyon','segmentasyon;'};
cell_list{5,1} = {'Nesne tanıma','nesne_tani;'};
cell_list{6,1} = {'Sonuclar','sonuc;'};
cell_list{7,1} = {'Çıkış',['disp("Bye. "hiko" yazarak yediden çalıştırabilirsiniz.");
close(' num2str(fig_number) ');']}; %{'Exit','Tez_gui;'};

%-----menuyu goruntuleme fonksiyonunu cagirmak: -----
show_window(cell_list,fig_number,title_figure,150,20,5,'clean',12);


%-------- menunun bitimi----------%



%-----------Butonlarin olusturulmasi----------%

function
show_window(cell_list,fig_number,title_figure,x_size,y_size,gap_x,font_name,font_
size)


if ~exist('cell_list'),
   error('No description of the functions');
end;
if ~exist('fig_number'),
   fig_number = 1;
end;
```

EK-3 (Devam) Birinci bölüm uygulamada kullanılan kaynak kodlar

```
if ~exist('title_figure'),
   title_figure = 'Hikmet BAL';
end;
if ~exist('x_size'),
   x_size = 85;
end;

if ~exist('y_size'),
   y_size = 14;
end;
if ~exist('gap_x'),
   gap_x = 0;
end;
if ~exist('font_name'),
   font_name = 'clean';
end;
if ~exist('font_size'),
   font_size = 8;
end;

figure(fig_number); clf;
pos = get(fig_number,'Position');

[n_row,n_col] = size(cell_list);

fig_size_x = x_size*n_col+(n_col+1)*gap_x;
fig_size_y = y_size*n_row+(n_row+1)*gap_x;

set(fig_number,'Units','points', ...
   'BackingStore','off', ...
   'Color',[0.2 0.2 0.2], ...
   'MenuBar','none', ...
   'Resize','off', ...
   'Name',title_figure, ...
'Position',[pos(1) pos(2) fig_size_x fig_size_y], ...
'NumberTitle','off');    %,'WindowButtonMotionFcn',['figure('    num2str(fig_number)
');']);

h_mat = zeros(n_row,n_col);
posx = zeros(n_row,n_col);
posy = zeros(n_row,n_col);
```

EK-3 (Devam) Birinci bölüm uygulamada kullanılan kaynak kodlar

```
for i=n_row:-1:1,
  for j = n_col:-1:1,
    posx(i,j) = gap_x+(j-1)*(x_size+gap_x);
    posy(i,j) = fig_size_y - i*(gap_x+y_size);
  end;
end;

for i=n_row:-1:1,
   for j = n_col:-1:1,
      if ~isempty(cell_list{i,j}),
         if ~isempty(cell_list{i,j}{1}) & ~isempty(cell_list{i,j}{2}),
            h_mat(i,j) = uicontrol('Parent',fig_number, ...
               'Units','points', ...
               'Callback',cell_list{i,j}{2}, ...
               'ListboxTop',0, ...
               'Position',[posx(i,j)  posy(i,j)  x_size   y_size], ...
               'String',cell_list{i,j}{1}, ...
               'fontsize',font_size,...
               'fontname',font_name,...
               'Tag','Pushbutton1');
         end;
      end;
   end;
end;

%--------Butonlarin olusturulmasi----------%
```

EK-3 (Devam) Birinci bölüm uygulamada kullanılan kaynak kodlar

dosya_ac.m

```
clear;
clc;
[file, pathname] = uigetfile({ ...
       '*.jpg', 'JPEG image (*.jpg)'; ...
       '*.png', 'Portable Network Graphics(*.png)'; ...
       '*.bmp', 'Windows Bitmap (*.bmp)'; ...
       '*.gif', 'GIF image (*.gif)'; ...
       '*.pcx', 'Windows Paintbrush (*.pcx)'; ...
       '*.tif', 'Tagged Image File (*.tif)'; ...
       '*.*', 'All Files (*.*)'}, ...
       'Dosya seçiniz');
if isequal(file, 0) | isequal(pathname, 0)
   disp('dosya seçilmedi.');
   srcImg = [];
else
   srcImg = imread(file);
end

figure('Name','orjinal Resim','Numbertitle','off'),imshow(srcImg),title('Orjinal
resim');
dosya_ozellik=imfinfo(file);

if (dosya_ozellik.BitDepth > 8) & (dosya_ozellik.ColorType== 'truecolor')
   gryTmp=rgb2gray(srcImg);
else
   gryTmp=srcImg;
end

gryImg=Im2double(gryTmp);
figure('Name','Gri Düzey Resim','Numbertitle','off'),imshow(gryImg),title('gri düzey
resim');
clear gryTmp file dosya_ozellik;
clc;
```

EK-3 (Devam) Birinci bölüm uygulamada kullanılan kaynak kodlar

intensity_trans.m (histogram işleme)

```
disp('------------------------------------------------------------')
disp('Histogram isleme icin bir metot seciniz. Bu metotlar: ')
disp('Negatif Transformasyon icin: ´neg´ ')
disp('Logaritmik transformasyon icin: ´log´,´C´ ')
disp('Gamma transformasyonu icin: ´gamma´,´gamma_degeri´ ')
disp('Histogram gerdirme için: ´ger´ ..... girmelisiniz.')
disp('Histogram esitleme için: ´esitle´ ..... girmelisiniz.')
disp('Adaptive Histogram esitleme için: ´adaptive´ ...  girmelisiniz.')
disp('Non-Uniform Aydinlatmayi düzelmek icin: ´non´ ... girmelisiniz.')
disp('------------------------------------------------------------')
secim=input('Seciminiz:  ','s');

[a, value]=strread(secim,'%q%f','delimiter',',');
method=lower(char(a));


if strcmp(class(gryImg),'double') & ~strcmp(method,'log')
   gryImg1=mat2gray(rgb2gray(srcImg));
else
   gryImg1=gryImg;
end

switch method
   case 'neg'
      gryHistProcessed=imcomplement(gryImg1);
   case 'log'
      if isempty(value)
         c=1;
      else
         c=value;
      end
      gryHistProcessed=c*(log(1+double(gryImg1)));
   case 'gamma'
      if isempty(value)
         error('yetersiz arguman giris yaptiniz...');
      end
      gam=value;
      gryHistProcessed=imadjust(gryImg1,[],[],gam);
```

EK-3 (Devam) Birinci bölüm uygulamada kullanılan kaynak kodlar

```
    case 'ger'
        m=mean2(gryImg1);
        E=4.0;
        gryHistProcessed=1./(1+(m./(gryImg1+eps)).^E);
    case 'esitle'
        gryHistProcessed=histeq(gryImg1);
    case 'adaptive'
        gryHistProcessed=adapthisteq(gryImg1);
    case 'non'
        se=strel('disk',3);

gryHistProcessed=imsubtract(imadd(gryImg1,imtophat(gryImg1,se)),imbothat(gryI
mg1,se));
    otherwise
        error('yanlis metot girisi yaptiniz...');
end
Hist_ham=imhist(im2uint8(gryImg1));
Hist_islenmis=imhist(im2uint8(gryHistProcessed));
figure('Name','Resim','Numbertitle','off');
imshow(gryHistProcessed, 'InitialMagnification','fit'),title('islenmis resim');

figure('Name','Histogram','Numbertitle','off');
subplot(2,1,1);
imhist(im2uint8(gryImg1));
title('islenmemis Histogram');

subplot(2,1,2);
imhist(im2uint8(gryHistProcessed));
title('islenmis Histogram ');

gryHistProcessed=(im2double(gryHistProcessed));
clear a gryImg1 method secim value se;
clc;
```

EK-3 (Devam) Birinci bölüm uygulamada kullanılan kaynak kodlar

spatial_filtreleme.m

```
clc;
disp('------------------------------------------------------------')
disp('Resindeki gürültü seviyesini düsürmek icin uygulamak istediginiz')
disp('asagidaki filtreleme ´METOT´larinda birini, filtre maskesinin ')
disp('´m X n´ boyutunu; Ayrica Gaussian ve laplacian of Gaussian filtreler')
disp(' icin ´sigma´ degerini Giriniz. Varsayilan (m X n) 3X3 dür.')
disp('------------------------------------------------------------')
disp('Average filtre icin: ´ortalama´ ..... girmelisiniz.')
disp('Geometric Average filtre icin: ´gortalama´ .....girmelisiniz. ')
disp('Gaussian filtre icin: ´gauss´,´sigma´ ..... girmelisiniz.')
disp('Laplacian filtre icin: ´laplace´ .... girmelisiniz.')
disp('Laplacion of Gaussian için: ´log´,´sigma´.... girmelisiniz.')
disp('Median filtre için: ´median´ ..... girmelisiniz.')
disp('Adaptive Median icin : ´amedian´ ...  girmelisiniz.')
disp('Max filtereleme icin: ´max´ ... girmelisiniz.')
disp('Min filtereleme icin: ´min´ ... girmelisiniz.')
disp('Duzeltilmis ortalama filtre icin:  ´alphatrim´ . girmelisiniz.')
disp('------------------------------------------------------------')
secim=input('Seciminiz:  ','s');

[a, value1, value2, value3]=strread(secim,'%q%f%f%f','delimiter',',');
type=lower(char(a));



if (~isempty(value1)) & (~isempty(value2)) & (~isempty(value3))
   m=value1; n=value2; zigma = value3;
elseif isempty(value1) & isempty(value2) & isempty(value3)
   m=3; n=3; zigma= 0.5; q=1.5; d=2;
else
   error('dirgiginiz arguman sayisi yanlistir..');
end

switch type
   case 'ortalama'
      w=fspecial('average', [m n]);
      filteredGray=imfilter(gryHistProcessed, w, 'replicate');
   case 'gortalama'
      warning off;
      filteredGray=exp(imfilter(log(gryHistProcessed),
ones(m,n),'replicate')).^(1/m/n);
      warning on;
```

EK-3 (Devam) Birinci bölüm uygulamada kullanılan kaynak kodlar

```
    case 'gauss'
        w=fspecial('gaussian', [m n],zigma);
        filteredGray=imfilter(gryHistProcessed, w, 'replicate');
    case 'laplace'
        w=fspecial('laplacian',0);
        lap_tmp=imfilter(gryHistProcessed,w,'replicate');
        filteredGray=gryHistProcessed-lap_tmp;
    case 'log'
        w=fspecial('log', [m n],zigma);
        filteredGray=imfilter(gryHistProcessed, w, 'replicate');
    case 'median'
        filteredGray=medfilt2(gryHistProcessed, [m n], 'symmetric');
    case 'max'
        filteredGray=ordfilt2(gryHistProcessed, m*n, ones(m,n), 'symmetric');
    case 'min'
        filteredGray=ordfilt2(gryHistProcessed, 1, ones(m,n), 'symmetric');
    case 'amedian'
        filteredGray=adpmedian(gryHistProcessed,7);
    case 'alphatrim'
        filteredGray=imfilter(gryHistProcessed,ones(m, n),'symmetric');
        for k=1:d/2
            filteredGray=imsubtract(filteredGray,ordfilt2(gryHistProcessed, k, ones(m,n),
'symmetric'));
        end

        for k=(m*n-(d/2)+1):m*n
            filteredGray=imsubtract(filteredGray,ordfilt2(gryHistProcessed, k, ones(m,n),
'symmetric'));
        end
        filteredGray=filteredGray/(m*n-d);

    otherwise
        error('yanlis filtreleme tipi girdiniz ..')
end

figure('Name','Filtrelenmis','Numbertitle','off');
imshow(filteredGray),title([type,'  filtrelenmis resim']);

clear a d lap_tmp m n q secim sigma zigma value1 value2 value3 se k type;
clc;
```

EK-3 (Devam) Birinci bölüm uygulamada kullanılan kaynak kodlar

segmentasyon.m

```
d={'Sobel','Prewitt','Roberts','Laplacian of Gaussian','Canny'};
mesaj=('metot seciniz');
[s,v] = listdlg('PromptString','Select a file:',...
                'SelectionMode','single','Name','Kenar dedektörleri',...
                'PromptString',mesaj,'ListString',d);

 if(s==1)
    edgeGray = edge(filteredGray,'sobel');
    figure('Name','Nesne Kenarları','Numbertitle','off');
    imshow(edgeGray),title('Sobel Dedektoru ile kenarlar');
 elseif(s==2)
    edgeGray = edge(filteredGray,'prewitt');
    figure('Name','Nesne Kenarları','Numbertitle','off');
    imshow(edgeGray),title('Prewitt Dedektoru ile kenarlar');
 elseif(s==3)
    edgeGray = edge(filteredGray,'roberts');
    figure('Name','Nesne Kenarları','Numbertitle','off');
    imshow(edgeGray),title('Roberts Dedektoru ile kenarlar');
 elseif(s==4)
    edgeGray = edge(filteredGray,'log', 0.015);
    figure('Name','Nesne Kenarları','Numbertitle','off');
    imshow(edgeGray),title('Laplacian of Gaussian Dedektoru ile kenarlar');
 elseif(s==5)
    edgeGray = edge(filteredGray, 'canny',[0.1 0.3],.6);
    figure('Name','Nesne Kenarları','Numbertitle','off');
    imshow(edgeGray),title('Canny Dedektoru ile kenarlar');
 end


if v==1
    binaryImg=im2bw(filteredGray);
    [B,L,N] = bwboundaries(binaryImg);
    figure('Name','Nesneler','Numbertitle','off');
    imshow(gryImg); title('Nesne sinirlari kirmizi, ic detaylar yesil'); hold on;
    for k=1:length(B),
       boundary = B{k};
       if(k > N)
          plot(boundary(:,2), boundary(:,1), 'g','LineWidth',1);
       else
          plot(boundary(:,2), boundary(:,1), 'r','LineWidth',1);
       end
    end
```

EK-3 (Devam) Birinci bölüm uygulamada kullanılan kaynak kodlar

```
else
   error('Dedektör secmediniz...');
end

clear B L N k boundary s mesaj d v;
clc;
```

EK-3 (Devam) Birinci bölüm uygulamada kullanılan kaynak kodlar

nesne_tani.m

```
binaryImg=im2bw(filteredGray);
warning off;
filledBinary=imfill(binaryImg, 'holes');
warning off;

[Boundry, LabeledBin, ObjNum]=bwboundaries(filledBinary);
boyen=size(gryImg);
if boyen==[480 640]
   alan=30;
else
   alan=40;
end
for K=1:ObjNum
     bounds=Boundry{K};
     r=Boundry{K,1}(1:length(Boundry{K}),2);
     c=Boundry{K,1}(1:length(Boundry{K}),1);
     taslar{K} = padarray(bound2im(bounds),[5 5],'both');
     crop_taslar{K} = padarray(imcrop(binaryImg, [min(r) min(c) max(r)-min(r)
max(c)-min(c)]),[5 5],'both');
     [H,theta, rho]=hough(taslar{K},.5);
     [r, c]=houghpeaks(H,5);
     lines=houghlines(taslar{K},theta,rho,r,c);
     tmpAcilar= [lines.theta];


     acilar{K} = tmpAcilar(1);
     if (acilar{K} < 0)
       duzeltmeAcisi{K} = -90 -acilar{K};
     else
       duzeltmeAcisi{K} = +90 -acilar{K};
     end
     duz_crop_tas{K} = imrotate(crop_taslar{K}, duzeltmeAcisi{K}, 'bilinear',
'loose');
     invers_duz_crop_taslar{K} = imcomplement(duz_crop_tas{K} );
     labeled_inv_duz_crop_taslar{K} = bwlabel(invers_duz_crop_taslar{K});
     status=regionprops(labeled_inv_duz_crop_taslar{K},'Area', 'Centroid');
     center=regionprops(LabeledBin, 'Centroid');
     koordinat{K,1}=[fix(center(K,1).Centroid(1)),fix(center(K,1).Centroid(2))];
     noktaAdediUst{K}=0;
     noktaAdediAlt{K}=0;
```

EK-3 (Devam) Birinci bölüm uygulamada kullanılan kaynak kodlar

```
    for I=1:length(status)
      if (status(I).Area < alan) &
(status(I).Centroid(2)<length(invers_duz_crop_taslar{K}) /2 );
          noktaAdediUst{K}=noktaAdediUst{K}+1 ;
      elseif (status(I).Area < alan) &
(status(I).Centroid(2)>length(invers_duz_crop_taslar{K}) /2 );
          noktaAdediAlt{K}=noktaAdediAlt{K}+1 ;
      end
    end
end

%clear H I c r rho boyen alan tmpAcilar, center;
clc;
```

EK-3 (Devam) Birinci bölüm uygulamada kullanılan kaynak kodlar

sonuc.m

```
figure('Name','Sonuclar','Numbertitle','off'),
imshow(gryImg),title('Etiketlenmis resim');
hold on;
for Ka=1:ObjNum
   sol_xkoor=min(Boundry{Ka,1}(1:length(Boundry{Ka}),2))-5;
   sag_xkoor=max(Boundry{Ka,1}(1:length(Boundry{Ka}),2))-5;
   ust_ykoor=min(Boundry{Ka,1}(1:length(Boundry{Ka}),1));
   alt_ykoor=max(Boundry{Ka,1}(1:length(Boundry{Ka}),1))+10;
   text(sol_xkoor,  ust_ykoor,  [noktaAdediUst(Ka),'--',noktaAdediAlt(Ka)],'fontsize',
10,'Color','r');
   text(sol_xkoor, alt_ykoor, ['Tas No:',num2str(Ka)], 'fontsize', 10,'Color','g');
%    text(sag_xkoor, ust_ykoor, [koordinat(Ka,1)],'fontsize', 10,'Color','y');
%    text(sag_xkoor, ust_ykoor, [duzeltmeAcisi(1,Ka)],'fontsize', 10,'Color','y');
end

clear H I K Ka xkoor alt_ykoor ust_ykoor c r rho status theta tmpAcilar;
save('Sonuclar');
```

EK-4 İkinci bölüm uygulamada kullanılan kaynak kodlar

AnaMenu.m

```
function Ana_Menu,

cell_list = { };

%-------- Menu secenekleri ----------%
fig_number = 1;

title_figure = 'Tez II.Bolum - Hikmet BAL';

cell_list{1,1} = {'Dosya aç','dosya_ac;'};
cell_list{2,1} = {'Yeniden Yapılandır','reconstruct;'};
cell_list{3,1} = {'Bölümlendirme','segment;'};
cell_list{4,1} = {'Sonuclar','sonuc;'};
cell_list{5,1} = {'Çıkış',['disp("Bye. "AnaMenu" yazarak yediden
çalıştırabilirsiniz."); close(' num2str(fig_number) ');']}; %{'Exit','Tez_gui;'};

%-----menuyu goruntuleme fonksiyonunu cagirmak: -----
show_window(cell_list,fig_number,title_figure,150,20,5,'clean',12);
%-------- menunun bitimi----------%
%------------------------Butonlarin olusturulmasi-----------------%

function
show_window(cell_list,fig_number,title_figure,x_size,y_size,gap_x,font_name,font_
size)


if ~exist('cell_list'),
   error('No description of the functions');
end;

if ~exist('fig_number'),
   fig_number = 1;
end;
if ~exist('title_figure'),
   title_figure = 'Hikmet BAL';
end;
if ~exist('x_size'),
   x_size = 85;
end;
if ~exist('y_size'),
   y_size = 14;
end;
if ~exist('gap_x'),
```

EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

```
    gap_x = 0;
end;
if ~exist('font_name'),
    font_name = 'clean';
end;
if ~exist('font_size'),
    font_size = 8;
end;

figure(fig_number); clf;
pos = get(fig_number,'Position');

[n_row,n_col] = size(cell_list);

fig_size_x = x_size*n_col+(n_col+1)*gap_x;
fig_size_y = y_size*n_row+(n_row+1)*gap_x;

set(fig_number,'Units','points', ...
    'BackingStore','off', ...
    'Color',[0.2 0.2 0.2], ...
    'MenuBar','none', ...
    'Resize','off', ...
    'Name',title_figure, ...
'Position',[pos(1) pos(2) fig_size_x fig_size_y], ...
'NumberTitle','off'); %,'WindowButtonMotionFcn',['figure(' num2str(fig_number)
');']);

h_mat = zeros(n_row,n_col);

posx = zeros(n_row,n_col);
posy = zeros(n_row,n_col);

for i=n_row:-1:1,
  for j = n_col:-1:1,
     posx(i,j) = gap_x+(j-1)*(x_size+gap_x);
     posy(i,j) = fig_size_y - i*(gap_x+y_size);
  end;
end;

for i=n_row:-1:1,
    for j = n_col:-1:1,
       if ~isempty(cell_list{i,j}),
          if ~isempty(cell_list{i,j}{1}) & ~isempty(cell_list{i,j}{2}),
             h_mat(i,j) = uicontrol('Parent',fig_number, ...
                'Units','points', ...
```

EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

```
                'Callback',cell_list{i,j}{2}, ...
                'ListboxTop',0, ...
                'Position',[posx(i,j)  posy(i,j)  x_size   y_size], ...
                'String',cell_list{i,j}{1}, ...
                'fontsize',font_size,...
                'fontname',font_name,...
                'Tag','Pushbutton1');
            end;
        end;
    end;
end;
```

EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

dosya_ac.m

```
warning off;
clear;
clc;
[file, pathname] = uigetfile({ ...
       '*.jpg', 'JPEG image (*.jpg)'; ...
       '*.png', 'Portable Network Graphics(*.png)'; ...
       '*.bmp', 'Windows Bitmap (*.bmp)'; ...
       '*.gif', 'GIF image (*.gif)'; ...
       '*.pcx', 'Windows Paintbrush (*.pcx)'; ...
       '*.tif', 'Tagged Image File (*.tif)'; ...
       '*.*', 'All Files (*.*)'}, ...
       'Dosya seçiniz');
if isequal(file, 0) | isequal(pathname, 0)
   disp('dosya seçilmedi.');
   srcImg = [];
else
   srcImg = imread(file);
end

dosya_ozellik=imfinfo(file);

if (dosya_ozellik.BitDepth > 8) & (dosya_ozellik.ColorType== 'truecolor')
   gryTmp=rgb2gray(srcImg);
else
   gryTmp=srcImg;
end

clc;
disp('-----------------------------------------------------------')
disp('Çalışma Alanını Seçiniz')
disp('-----------------------------------------------------------')
beep,beep;
pause;
figure('Name','orjinal Resim','Numbertitle','off'),imshow(srcImg),title('Orjinal
resim');
[B,xi,yi]=roipoly(srcImg);
gryImg=imcrop(gryTmp, [fix(min(xi)) fix(min(yi)) (fix(max(xi)))-(fix(min(xi)))
(fix(max(yi)))-(fix(min(yi)))]);
crop_koor=[fix(min(xi)) fix(min(yi)) fix(max(xi)) fix(max(yi))];
figure('Name','Gri Düzey Resim','Numbertitle','off'),imshow(gryImg),title('Secilen
gri düzey resim');
clear gryTmp file dosya_ozellik B pathname xi yi;
imwrite(gryImg,'GrayImg.jpg','quality',100); clc;
```

EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

reconstruct.m

```
clc;
disp('------------------------------------------------------------')
disp(['Yapısal Eleman Biçimleri:'])
disp(['    -disk'])
disp(['    -diamond'])
disp(['    -square'])
disp(['           '])
disp(['Yeniden yapılandırma için Yapısal Elemanın biçim ve boyutunu giriniz.'])
disp(['girilen n değeri için n*n boyutlarında yapısal eleman oluşturulacaktır.'])
disp(['Ör: disk,10 '])
disp('------------------------------------------------------------')
secim=input('Seciminiz:  ','s');

[name value]=strread(secim,'%s %d','delimiter',',');

if (isempty(name))|(isempty(value));
   name='disk';
   value=10;
end

method=lower(char(name));
switch method
   case 'disk'
      se = strel('disk', value);
   case 'diamond'
      se = strel('diamond', value);
   case 'square'
      se = strel('square', value);
   otherwise
      error('yanlis metod girisi yaptiniz...');
end

erodedGray = imerode(gryImg, se);
Grayobr = imreconstruct(erodedGray, gryImg);
dilatedGrayobr = imdilate(Grayobr, se);
Gray_obrcbr = imreconstruct(imcomplement(dilatedGrayobr),
imcomplement(Grayobr));
Gray_obrcbr = imcomplement(Gray_obrcbr);
figure('Name','Reconstructed Image','Numbertitle','off'), imshow(Gray_obrcbr),
title('Gray_obrcbr'),title('Yeniden Yapilandirilmis Resim');
clear erodedGray Grayobr dilatedGrayobr name value method secim
% imwrite(Gray_obrcbr,'Reconstructed_GrayImg.jpg','quality',100);
```

EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

segment.m

```
function Segment,

cell_list = { };

%-------- Menu secenekleri ----------%

fig_number = 30;

title_figure = 'Bolumlendirme';

cell_list{1,1} = {'WaterShed','water_shed;'};
cell_list{2,1} = {'Esikle','threshy;'};
cell_list{3,1} = {'Geri Dön',['close(' num2str(fig_number) ');']}; %{'Exit','Tez_gui;'};

%-----menuyu goruntuleme fonksiyonunu cagirmak: -----
show_window(cell_list,fig_number,title_figure,100,20,5,'clean',12);
%-------- menunun bitimi----------%

%------------------------Butonlarin olusturulmasi----------%

function
show_window(cell_list,fig_number,title_figure,x_size,y_size,gap_x,font_name,font_
size)


if ~exist('cell_list'),
   error('No description of the functions');
end;

if ~exist('fig_number'),
   fig_number = 30;
end;
if ~exist('title_figure'),
   title_figure = 'Bolumlendirme';
end;
if ~exist('x_size'),
   x_size = 85;
end;
if ~exist('y_size'),
   y_size = 14;
end;
if ~exist('gap_x'),
   gap_x = 0;
```

EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

```
end;
if ~exist('font_name'),
   font_name = 'clean';
end;
if ~exist('font_size'),
   font_size = 8;

end;

figure(fig_number); clf;
pos = get(fig_number,'Position');

[n_row,n_col] = size(cell_list);

fig_size_x = x_size*n_col+(n_col+1)*gap_x;
fig_size_y = y_size*n_row+(n_row+1)*gap_x;

set(fig_number,'Units','points', ...
   'BackingStore','off', ...
   'Color',[0.2 0.2 0.2], ...
   'MenuBar','none', ...
   'Resize','off', ...
   'Name',title_figure, ...
'Position',[380 250 fig_size_x fig_size_y], ...
'NumberTitle','off'); %,'WindowButtonMotionFcn',['figure(' num2str(fig_number)
');']);

h_mat = zeros(n_row,n_col);

posx = zeros(n_row,n_col);
posy = zeros(n_row,n_col);

for i=n_row:-1:1,
  for j = n_col:-1:1,
     posx(i,j) = gap_x+(j-1)*(x_size+gap_x);
     posy(i,j) = fig_size_y - i*(gap_x+y_size);
  end;
end;

for i=n_row:-1:1,
   for j = n_col:-1:1,
      if ~isempty(cell_list{i,j}),
         if ~isempty(cell_list{i,j}{1}) & ~isempty(cell_list{i,j}{2}),
            h_mat(i,j) = uicontrol('Parent',fig_number, ...
```

EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

```
                'Units','points', ...
                'Callback',cell_list{i,j}{2}, ...
                'ListboxTop',0, ...
                'Position',[posx(i,j)  posy(i,j)  x_size   y_size], ...
                'String',cell_list{i,j}{1}, ...
                'fontsize',font_size,...
                'fontname',font_name,...
                'Tag','Pushbutton1');
          end;
       end;
    end;
end;

%------------------------Butonlarin olusturulmasi-----------------%
```

EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

threshy.m

```
warning off
se2 = strel(ones(5,5));
done=false;
while  ~done
   clc;
   Th=graythresh(Gray_obrcbr);
   disp('-----------------------------------------------------')
   disp(['Eşik değeri seçiniz, (otomatik eşik değeri:',num2str(Th),')'])
   disp('-----------------------------------------------------')
   secim=input('Seciminiz:  ','s');

   [esik]=strread(secim,'%f','delimiter',',');

   if isempty(esik);
      threshy=Th*0.848;
   elseif esik >1
      error('yanlis metod girisi yaptiniz, eşik [0 1] aralığında olmalıdır...');
   else
      threshy=esik;
   end
   bw = im2bw(Gray_obrcbr, threshy);
   figure('Name','Raw Binary Image','Numbertitle','off'), imshow(bw),
title([num2str(threshy), '  degeri ile Esiklenmis görüntü']);
   bw2 = ~(imopen(~bw, se2));
   figure('Name','cleaned Binary Image','Numbertitle','off'), imshow(bw2),
title('Düzeltilmiş İkili Görüntü');
   BW = bwareaopen((~bw2), 400);
   BWnobord = imclearborder(BW, 4);
   figure('Name','Border Cleaned Binary Image','Numbertitle','off'),
imshow(BWnobord), title('Sinirlardaki yarim tanelerin temizlendigi görüntü');
   % imwrite(BW,'Binary.jpg','quality',100);

   seD = strel('diamond',1);
   BWfinal = imerode(BWnobord,seD);
   % figure, imshow(BWfinal), title('segmented image');


   BWoutline = bwperim(BWfinal);
   BWoutline_filled=imfill((imdilate(BWoutline == 1, (strel('disk',3)))),'holes');
   BWoutline_last=bwperim(BWoutline_filled);
   imwrite(BWoutline_last,'Outlines.jpg','quality',100);

   Segout = gryImg;
   Segout(imdilate(BWoutline_last == 1, (strel('disk',1)))) = 250;
```

EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

```
    figure('Name','Grain outlined original image','Numbertitle','off'), imshow(Segout),
title('işaretlenmiş tane sınırları');
    imwrite(Segout,'Result.jpg','quality',100);
    clear Th secim esik threshy bw2
    clc;
    disp('------------------------------------------------------------')
    disp('Eşikleme sonuçu yeterli mi? (E/H)')
    disp('------------------------------------------------------------')
    cevap=input('Seciminiz:  ','s');

    [cvp] = strread(cevap, '%s');
    if (isempty(cvp))|(isequal(char(cvp),'e'))|(isequal(char(cvp),'E'));
       done = 10>9;
    elseif (isequal(char(cvp),'h'))|(isequal(char(cvp),'H'));
       done = 10<9;
    else
       error('hatalı harf dirdiniz')
       done=false;
    end
end

BWfilled=imfill(BWoutline_last,'holes');
BWfil=BW|BWfilled;
[Boundry, LabeledBW, grftNum]=bwboundaries(BWfil);
[Boundry4G, LabeledBW4G, grftNum4G]=bwboundaries(BWfilled);
status=regionprops(LabeledBW,'Area');
status4G=regionprops(LabeledBW4G,'Area');
center=regionprops(LabeledBW, 'Centroid');
figure('Name','Labeled image','Numbertitle','off'), imshow(Segout),title('Etiketlenmis
resim'); hold on;

for K=1:grftNum;
%    koordinat(K,1)=[fix(center(K,1).Centroid(1)),fix(center(K,1).Centroid(2))];
    alan{K}=[(status(K).Area)];
    sol_xkoor=(min(Boundry{K,1}(1:length(Boundry{K}),2))+10);
    sag_xkoor=max(Boundry{K,1}(1:length(Boundry{K}),2));
    ust_ykoor=min(Boundry{K,1}(1:length(Boundry{K}),1));
    alt_ykoor=max(Boundry{K,1}(1:length(Boundry{K}),1));
    mrkz_xkoor=fix(center(K,1).Centroid(1));
    mrkz_ykoor=fix(center(K,1).Centroid(2));
%    text(sol_xkoor, alt_ykoor, ['#',num2str(K)], 'fontsize', 10,'Color','y');
    text((sol_xkoor-10), mrkz_ykoor, [alan(K)],'fontsize', 10,'Color','g');
end
warning on
```

EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

water_shed.m

```
I=gryImg;
hy = fspecial('sobel');
hx = hy';
Iy = imfilter(double(I), hy, 'replicate');
Ix = imfilter(double(I), hx, 'replicate');
gradmag = sqrt(Ix.^2 + Iy.^2);
figure('Name','Gradient Magnitude of the image','Numbertitle','off'),
imshow(gradmag,[]), title('Gradient Genliği görüntüsü')
imwrite(uint8(gradmag),'Gradient_of_Gray.jpg','quality',100);


bw = im2bw(Gray_obrcbr, (graythresh(Gray_obrcbr))*0.83);
se2 = strel(ones(5,5));
bw2 = ~(imopen(~bw, se2));
BW = bwareaopen((~bw2), 200);

fgm=BW;
figure('Name','Foreground Markers image','Numbertitle','off'), imshow(fgm),
title('Önplan görüntüsü')

se2 = strel(ones(5,5));

 fgm2 = imclose(fgm, se2);


 D = bwdist(fgm2);
% figure, imshow(D), title('distance trans');
 DL = watershed(D);
% figure, imshow(DL), title('watershed of distance trans. im.');
 bgm = DL == 0;
% figure, imshow(bgm), title('bgm')

 gradmag2 = imimposemin((imclose(imopen(gradmag,ones(5,5)),ones(5,5))), bgm |
fgm);
% figure, imshow(gradmag2), title('gradmag2')

 L = watershed(gradmag2);
 G=L==0;
 figure('Name','Watershed of the composite image','Numbertitle','off'),
imshow(imdilate(G, (strel('disk',1)))), title('Watershed bölümlendirme sonucu')
 imwrite(G,'WS_lines.jpg','quality',100);

 seD = strel('diamond',1);
```

EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

```
 BWfinal = imerode(fgm2,seD);
 BWoutline = bwperim(BWfinal);
 BWoutline_filled=imfill((imdilate(BWoutline == 1, (strel('disk',3)))),'holes');
```

EK-3 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

```
 BWoutline_last=bwperim(BWoutline_filled);
% imwrite(BWoutline_last,'Outlines.jpg','quality',100);

 Segout = I;
 Segout(imdilate(BWoutline_last == 1, (strel('disk',1)))) = 250;
 figure('Name','Outlined Grains','Numbertitle','off'), imshow(Segout), title('Toplam
alan icin etiketlenmis resim');
% imwrite(Segout,'Result.jpg','quality',100);

 BWfilled=imfill(BWoutline_last,'holes');
 BWfill=fgm2|BWfilled;



 % --------------- alanları etiketleme --------------------
G=L==0;
BWfill4G=(imclearborder((imfill(G,'holes')),4));
%[Boundry, LabeledBW, grftNum]=bwboundaries(BWfilled);
[Boundry, LabeledBW, grftNum]=bwboundaries(BWfill);
[Boundry4G, LabeledBW4G, grftNum4G]=bwboundaries(BWfill4G);
status=regionprops(LabeledBW,'Area');
status4G=regionprops(LabeledBW4G,'Area');
center=regionprops(LabeledBW, 'Centroid');
center4G=regionprops(LabeledBW4G, 'Centroid');
figure, imshow(Segout),title('Toplam Alan için Etiketlenmis resim');
hold on;

for K=1:grftNum;
%    koordinat(K,1)=[fix(center(K,1).Centroid(1)),fix(center(K,1).Centroid(2))];
   alan{K}=[(status(K).Area)];
   sol_xkoor=(min(Boundry{K,1}(1:length(Boundry{K}),2))+10);
   sag_xkoor=max(Boundry{K,1}(1:length(Boundry{K}),2));
   ust_ykoor=min(Boundry{K,1}(1:length(Boundry{K}),1));
   alt_ykoor=max(Boundry{K,1}(1:length(Boundry{K}),1));
   mrkz_xkoor=fix(center(K,1).Centroid(1));
   mrkz_ykoor=fix(center(K,1).Centroid(2));
%    text(sol_xkoor, alt_ykoor, ['#',num2str(K)], 'fontsize', 10,'Color','y');
   text((sol_xkoor-10), mrkz_ykoor, [alan(K)],'fontsize', 10,'Color','g');
```

EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

```
end

hold off;

G1=gryImg;
G1(imdilate(G == 1, (strel('disk',1)))) = 250;
imwrite(G1,'Results.jpg','quality',100);
figure('Name','Outlined Grains II','Numbertitle','off'), imshow(G1),title('Ortalama
Tane büyüklüğü için etiketlenmis resim');
hold on;

for K=1:grftNum4G;
%    koordinat(K,1)=[fix(center(K,1).Centroid(1)),fix(center(K,1).Centroid(2))];
   alan4G{K}=[(status4G(K).Area)];
   sol_xkoor4G=(min(Boundry4G{K,1}(1:length(Boundry4G{K}),2))+10);
   sag_xkoor4G=max(Boundry4G{K,1}(1:length(Boundry4G{K}),2));
   ust_ykoor4G=min(Boundry4G{K,1}(1:length(Boundry4G{K}),1));
   alt_ykoor4G=max(Boundry4G{K,1}(1:length(Boundry4G{K}),1));
   mrkz_xkoor4G=fix(center4G(K,1).Centroid(1));
   mrkz_ykoor4G=fix(center4G(K,1).Centroid(2));
%    text(sol_xkoor, alt_ykoor, ['#',num2str(K)], 'fontsize', 10,'Color','y');
   text((sol_xkoor4G-10), mrkz_ykoor4G, [alan4G(K)],'fontsize', 10,'Color','g');
end
hold off;
clear I hy hx Ix Iy bw bw2 BW se2 BWoutlined_filled G1
```

EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

sonuc.m

```
%---------- Toplam Alan ----------

Total_area=0;
for K=1:grftNum;
   alan{K}=[(status(K).Area)];
   Total_area=Total_area+alan{K};
end
clc
disp('------------------------------------------------------')
disp('ortalamaya dahil etmek ISTEMEDİĞİNİZ ..')
disp('minimum ve maximum grafit alanını giriniz ..')
disp('------------------------------------------------------')
except_area=input('Seciminiz: ','s');

[min_alan, max_alan] = strread(except_area, '%d %d','delimiter',',');

grain_size=0;
sayac=0;
for K=1:grftNum4G;
   alan4G{K}=[(status4G(K).Area)];
   if (alan4G{K} <= min_alan) | (alan4G{K} >= max_alan);
      grain_size=grain_size;
   else
      grain_size=grain_size+alan4G{K};
      sayac=sayac+1;
   end
end
Avrg_grain=(grain_size/sayac);

boy_en=size(gryImg);
im_area=boy_en(1)*boy_en(2);
yuzde=Total_area/im_area;
clc
disp(['Ortalama Grafit Tane büyüklüğü (piksel) :',num2str(Avrg_grain)]);
disp(['Toplam Grafit Alanı (piksel) :',num2str(Total_area)]);
disp(['Grafit Alanı Yüzdesi (%) :',num2str(100*yuzde)]);
warning on
```

EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

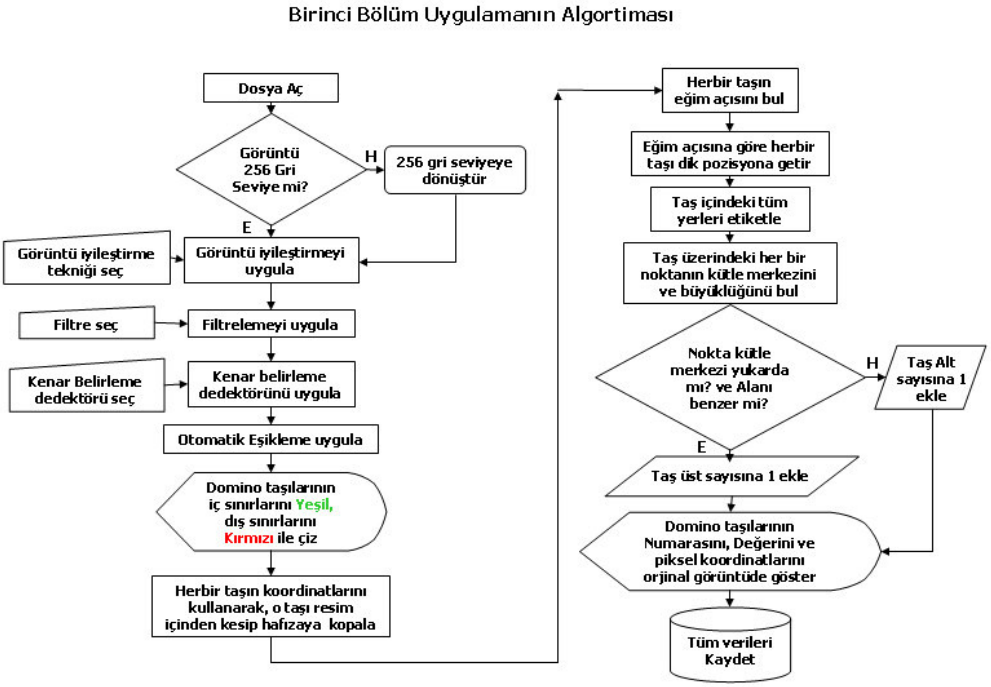
grain.m

```
f_c=gryImg;
f_crop=imcomplement(f_c);
figure,imshow(f_c),title('orjinal')
imwrite(f_c,'GrayImg.jpg','quality',100);

Im=f_crop;

se=strel('disk',5);
Im_o = imopen(Im, se);
%figure, imshow(Io), title('Io')

Im_e = imerode(Im, se);
Im_obr = imreconstruct(Im_e, Im);
%figure, imshow(Iobr), title('Iobr')

Im_oc = imclose(Im_o, se);
%figure, imshow(Ioc), title('Ioc')

Im_obrd = imdilate(Im_obr, se);
Im_obrcbr = imreconstruct(imcomplement(Im_obrd), imcomplement(Im_obr));
Im_obrcbr = imcomplement(Im_obrcbr);
figure, imshow(Im_obrcbr), title('Im_obrcbr')

Th=256*(graythresh(Im_obrcbr));


disp('-----------------------------------------------------------')
disp('istenen faz için minimum ve maximun  griseviye değeri giriniz')
disp(['(otomatik gri seviye değeri:',num2str(0.58*Th),'  (+-5))'])
disp('-----------------------------------------------------------')
maxlevel=input('Seciminiz:  ','s');

[level1, level2] = strread(maxlevel, '%d %d','delimiter',',');


fgm_max2 = imextendedmax(Im_obrcbr,level2);
fgm_max1 = imextendedmax(Im_obrcbr,level1);
fgm=(fgm_max1|fgm_max2);

figure, imshow(fgm), title('Binary görüntü (bölgesel bölümleme)')
imwrite(fgm,'Binary.jpg','quality',100);
```

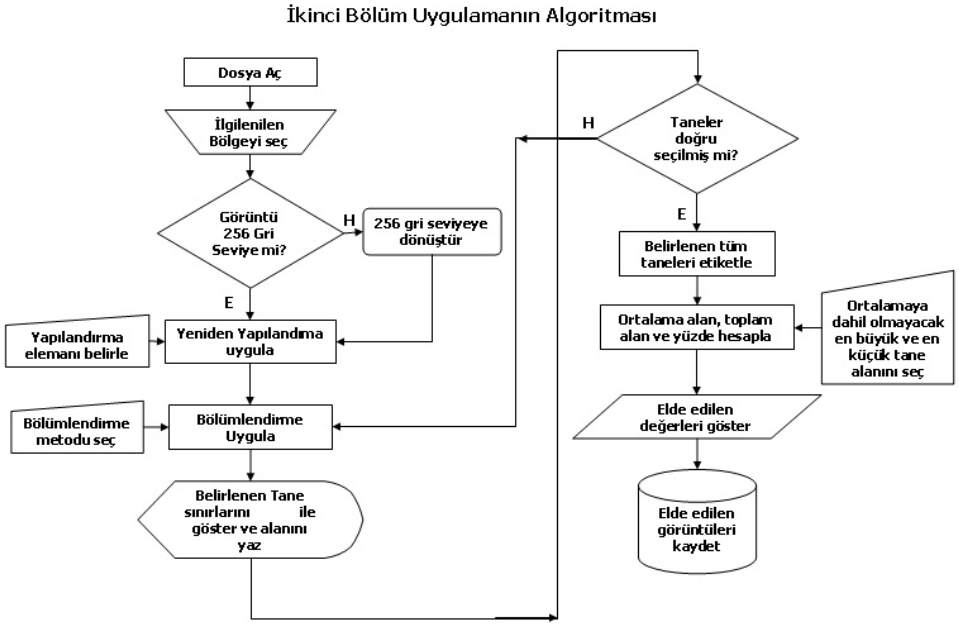
EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar

```
Im2 = Im;
Im2(fgm) = 120;
%figure, imshow(I2), title('fgm superimposed on original image')

fgm2 = imclose(fgm, ones(3,3));
fgm4 = bwmorph(fgm2,'thin',1);

Im4 = f_c;
Im4(fgm4) = 0;
figure, imshow(Im4)
title('original resim ve elde edilen tane görüntüsünün birlestirilmesi')
imwrite(Im4,'Result.jpg','quality',100);

bw_last = im2bw(Im4, graythresh(Im4));
se2 = strel(ones(3,3));
bwlast = ~(imopen(~bw_last, (strel('diamond',1))));
%figure, imshow(bw2), title('cleaned Binary image');
BW = imcomplement(bwlast);

% ---------- delikleri etiketleme ----------

[Boundry, LabeledBW, grftNum]=bwboundaries(BW);
status=regionprops(LabeledBW,'Area');
%center=regionprops(LabeledBW, 'Centroid');

Total_area=0;
for K=1:grftNum;
   alan{K}=[(status(K).Area)];
   Total_area=Total_area+alan{K};
end

figure('Name','Boundaries','Numbertitle','off');
imshow(f_c); title('Tane sinirlari kirmizi'); hold on;
for k=1:length(Boundry);
   boundary = Boundry{k};
   plot(boundary(:,2), boundary(:,1), 'r','LineWidth',2);
end

boy_en=size(gryImg);
im_area=boy_en(1)*boy_en(2);
yuzde=Total_area/im_area;
clc

disp(['Toplam Grafit Alanı (piksel) :',num2str(Total_area)]);
disp(['Faz Alanı Yüzdesi (%) :',num2str(100*yuzde)]);
```

EK-4 (Devam) İkinci bölüm uygulamada kullanılan kaynak kodlar



(a)



(b)

Şekil 3.1 (a) Birinci bölüm uygulamaya ait algoritma (b) İkinci bölüm uygulamaya ait algoritma

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : BAL, Hikmet

Uyruğu : T.C.

Doğum tarihi ve yeri : 06.10.1974 Ankara

Medeni hali : Evli

Telefon : 0 (532) 260 48 26

Faks : (-) -

e-mail : hikmet.bal@gmail.com.

Eğitim

Derece	Eğitim Birimi	Mezuniyet tarihi
Yüksek lisans	Gazi Ünv./ Makine Müh Bölümü	2006
Lisans	Gazi Ünv./ Makine Müh. Bölümü	2002
Ön Lisans	Anadolu Ünv./Açık Öğretim-Iktisat.	1996
Lise/Sınıf Ok.	Teknik Astsb. Hzlm.Ok. /MEBSOK	1992

İş Deneyimi

Yıl	Yer	Görev
2004--	MEBSOK K.lığı	Bilg. Teknisyeni Astsb.
2001-2004	2.Or. Hv.Alayı	Bilg. Teknisyeni Astsb.
1993-2001	GATA K.lığı	Bilg. Teknisyeni Astsb.
1992-1993	K.K.K.lığı	Bilg. Teknisyeni Astsb.

Yabancı Dil

İngilizce

Yayınlar

Hobiler

Satranç, Bilgisayar teknolojileri, Basketbol,Sinema,Tarih