

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**DİJİTAL GÖRÜNTÜ İŞLEME TEKNİKLERİ KULLANILARAK
GÖRÜNTÜLERDEN DETAY ÇIKARIMI**

YÜKSEK LİSANS TEZİ

Güzide Miray PERİHANOĞLU

Geomatik Mühendisliği Anabilim Dalı

Geomatik Mühendisliği Programı

OCAK 2015

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**DİJİTAL GÖRÜNTÜ İŞLEME TEKNİKLERİ KULLANILARAK
GÖRÜNTÜLERDEN DETAY ÇIKARIMI**

YÜKSEK LİSANS TEZİ

**Güzide Miray PERİHANOĞLU
(501111607)**

Geomatik Mühendisliği Anabilim Dalı

Geomatik Mühendisliği Programı

Tez Danışmanı: Öğr. Gör. Dr. Mehmet Ufuk ÖZERMAN

OCAK 2015

İTÜ, Fen Bilimleri Enstitüsü'nün 501111607 numaralı Yüksek Lisans Öğrencisi **Güzide Miray PERİHANOĞLU**, ilgili yönetmeliklerin belirlediği gerekli tüm şartları yerine getirdikten sonra hazırladığı “**DİJİTAL GÖRÜNTÜ İŞLEME TEKNİKLERİ KULLANILARAK GÖRÜNTÜLERDEN DETAY ÇIKARIMI**” başlıklı tezini aşağıda imzaları olan jüri önünde başarı ile sunmuştur.

Tez Danışmanı : **Öğr. Gör. Dr. Mehmet Ufuk ÖZERMAN**
İstanbul Teknik Üniversitesi

Jüri Üyeleri : **Prof. D. Dursun Zafer ŞEKER**
İstanbul Teknik Üniversitesi

Prof. Dr. Bülent BAYRAM
Yıldız Teknik Üniversitesi

Teslim Tarihi : 15 Aralık 2014
Savunma Tarihi : 19 Ocak 2015

Çok Değerli Annem ve Babam, Nevin ve Hasan PERİHANOĞLU'na..

ÖNSÖZ

Yüksek Lisans öğrenimimde bu alanda çalışmaya yönlendirerek ufkumun genişlemesini sağlayan ve bu çalışmamın her aşamasında önerileri ile beni yönlendiren, tez çalışmam boyunca yol gösteren, teşvik eden danışman hocam Öğr. Gör. Dr. Mehmet Ufuk ÖZERMAN'a teşekkürü bir borç bilirim.

Bu çalışma boyunca yardımlarını esirgemeyen ve ihtiyaç duyduğum her an yanımda olarak bana bilimsel danışmanlık veren çok değerli hocam Prof. Dr. Dursun Zafer ŞEKER'e sonsuz teşekkürlerimi sunarım.

Ayrıca, bana çalışmam süresince destek olan, sıkıntılarımı paylaşan arkadaşlarım Şafak ŞAHİN'e, Yük. Müh. Mehmet BAK'a, Özge KAYMAN'a, Mecit KESİCİ ve Sibel LAVAŞKAN'a teşekkürü bir borç bilirim.

Son olarak, yalnız tez çalışması süresince değil tüm yaşamım boyunca bana her konuda ve her zaman güvenen ayrıca destekleyen çok değerli annem, babam Nevin, Hasan PERİHANOĞLU'na, kardeşim Hakan PERİHANOĞLU'na, ablam ve eşi Merve ve Harun KARATEKİN'e gösterdikleri sonsuz hoşgörülerinden dolayı minnettarım.

Ocak 2015

Güzide Miray PERİHANOĞLU
(Jeodezi ve Fotogrametri Mühendisi)

İÇİNDEKİLER

Sayfa

ÖNSÖZ.....	vii
KISALTMALAR	xi
ŞEKİL LİSTESİ.....	xiii
SEMBOL LİSTESİ.....	xvii
ÖZET.....	xix
SUMMARY.....	xxi
1. GİRİŞ.....	1
2. DİJİTAL GÖRÜNTÜ İŞLEME	3
2.1 Dijital Görüntü	4
2.2 Dijital Görüntü İşleme Yöntemleri	6
2.2.1 Nokta işleme teknikleri.....	6
2.2.1.1 Negatif görüntü.....	6
2.2.1.2 Görüntü histogramı	7
2.2.1.3 Histogram eşitleme	9
2.2.1.4 Parlaklık ayarlama	10
2.2.1.5 Kontrast geliştirme.....	11
2.2.1.6 Eşikleme yöntemleri	13
2.2.2 Görüntü zenginleştirme yöntemleri	17
2.2.2.1 Görüntü yumuşatma.....	18
2.2.2.2 Görüntü keskinleştirme	21
2.2.2.3 Kenar belirleme yöntemleri.....	22
2.2.3 Morfolojik işlemler.....	26
2.2.3.1 Genişleme ve Aşınma morfoloji operatörleri.....	27
2.2.3.2 Açma ve Kapama morfoloji operatörleri	29
2.2.3.3 Diğer morfoloji operatörleri	30
3. MATLAB YAZILIMINDA GÖRÜNTÜ İŞLEME.....	31
3.1 MATLAB'ın Avantajları.....	31
3.2 MATLAB Ana Ekranı.....	32
3.3 MATLAB Kod Yazma Ortamı	33
3.3 MATLAB Arayüz Geliştirme Ortamı (Guide)	33
4. UYGULAMA	35
4.1 Uygulanan Yöntemler	35
4.1.1 Ön işleme	35
4.1.2 Eşikleme ve morfolojik işlemler	43
4.1.3 Detay çıkarımı	48
5. SONUÇLAR VE ÖNERİLER.....	55
KAYNAKLAR.....	57
EKLER.....	59
ÖZGEÇMİŞ..	67

KISALTMALAR

MATLAB	: MATrix LABoratory
CCD	: Kamera (Charge Coupled Device)
RGB	: Kırmızı,yeşil,mavi renk uzayı (Red, Green, Blue)
LOG	: Kenar belirleme filtresi (Laplacian of Gauss)
GUIDE	: Matlab arayüzü (Graphical User Interface Design)

ŞEKİL LİSTESİ

Sayfa

Şekil 2.1 : Örnek bir görüntünün örnekleme ve nicemlemesi.....	3
Şekil 2.2 : Örnek bir dijital görüntü ve belli bir kısmındaki piksellerin büyütülmüş hali.	4
Şekil 2.3 : Görüntü üzerindeki pikselasyon.....	5
Şekil 2.4 : Dijital görüntü çeşitleri.....	6
Şekil 2.5 : Negatif görüntü için dönüşüm fonksiyonu.....	7
Şekil 2.6 : Gri seviyeli görüntü ve histogramı.....	8
Şekil 2.7 : Gri seviyeli görüntü ve histogramında karanlık ve parlaklık imgesi.....	8
Şekil 2.8 : Gri seviyeli görüntü ve histogramında karşıtlığı düşük ve yüksek imge....	9
Şekil 2.9 : Bir görüntüye histogram eşitleme yönteminin uygulanması ve sonucu (a) orijinal görüntü, (b) histogram eşitleme uygulanmış görüntü, (c) orijinal görüntünün histogramı (d) histogram eşitleme uygulanmış görüntünün histogramı.....	10
Şekil 2.10 : Sabit sayıya göre parlaklık ayarı.....	11
Şekil 2.11 : (a) $c < 0$ için dönüşüm fonksiyonu, (b) $c > 0$ için dönüşüm fonksiyonu....	11
Şekil 2.12 : Sabit sayıya göre kontrast ayarı.....	12
Şekil 2.13 : (a) Kontrast germe işleminden önceki görüntü (b) Kontrast germe işleminden sonraki görüntü.	13
Şekil 2.14 : Lineer germe fonksiyonu.....	13
Şekil 2.15 : Lokal eşikleme sırası.....	15
Şekil 2.16 : Tek bir eşik değeri ile bölmelenen histogram	15
Şekil 2.17 : Otsu eşik değeri algoritmasıyla elde edilen görüntü.....	17
Şekil 2.18 : Averaj filtresi.....	18
Şekil 2.19 : (a) Orijinal görüntü (b) 5x5 Averaj filtresi geçirilmiş hali.....	19
Şekil 2.20 : Medyan filtresinin işlem adımı.....	20
Şekil 2.21 : (a) Orijinal görüntü (b) 3x3 Median filtre sonucu (c) 5x5 Median filtre sonucu (d) 7x7 Median filtre sonucu	20
Şekil 2.22 : (a) 12x12 Gauss $\sigma = 1.5$ b) 24x24 Gauss $\sigma = 3$	21
Şekil 2.23 : (a) Orijinal görüntü (b) Laplace maskesi geçirilmiş görüntü c) Maske ile görüntünün toplanmış hali.....	22
Şekil 2.24 : Kenar belirleme maskeleri	22
Şekil 2.25 : Gradient metodu blok diagramı.....	23
Şekil 2.26 : Prewitt operatörü maskeleri.....	24
Şekil 2.27 : (a) Orijinal görüntü (b) Prewitt G_x (c) Prewitt G_y	24
Şekil 2.28 : (a) Orijinal görüntü (b) Sobel kenar belirleme yöntemi ile oluşan görüntü.	24
Şekil 2.29 : (a) Orijinal görüntü (b) Canny kenar belirleme yöntemi ile oluşan görüntü.	25
Şekil 2.30 : (a) Gauss fonksiyonun 3-B görünüşü (b) Gauss fonksiyonun birinci türevi (c) Gauss fonksiyonun laplace'ı.....	26
Şekil 2.31 : Morfolojik yapıtaşı elemanı örnekleri.....	27

Şekil 2.32 : (a) Binari görüntü (b) Genişleme operatörü işlemi sonucunda oluşan görüntü.....	27
Şekil 2.33 : Genişleme işlemi temel işleyişi.....	28
Şekil 2.34 : (a) Binari görüntü (b) Aşınma operatörü sonucunda oluşan görüntü	28
Şekil 2.35 : Aşınma işlemi temel işleyişi	29
Şekil 2.36 : (a) Binari görüntü (b) Açma operatörü sonucunda oluşan görüntü.....	29
Şekil 2.37 : (a) Binari görüntü (b) Kapama operatörü sonucunda oluşan görüntü.....	30
Şekil 3.1 : Matlab ana ekranı	32
Şekil 3.2 : Matlab editör penceresi	33
Şekil 3.3 : Matlab arayüz geliştirme ortamı	34
Şekil 4.1 : Çalışmada kullanılan örnek renkli görüntüler.....	35
Şekil 4.2 : (a) Gri seviyeli görüntü (b) Gri seviyeli görüntünün histogramı.....	36
Şekil 4.3 : (a) Gri seviyeli görüntü (b) Gri seviyeli görüntünün histogramı.....	36
Şekil 4.4 : (a) Global histogram eşitlemesi sonrası görüntü (b) Histogram eşitlemesi sonrası görüntünün histogramı.	37
Şekil 4.5 : (a) Global histogram eşitlemesi sonrası görüntü (b) Histogram eşitlemesi sonrası görüntünün histogramı.	37
Şekil 4.6 : (a) Çalışmada kullanılan örnek renkli görüntü (b) $\sigma = 1.5$ ve 5×5 Gauss filtresi (c) $\sigma = 3$ ve 3×3 Gauss filtresi (d) $\sigma = 10$ ve 7×7 Gauss filtresi....	38
Şekil 4.7 : (a) Görüntüye 0.01 varyansla gürültü eklenmiş hali (b) Görüntünün 5×5 Gauss filtresi geçirildikten sonraki hali.....	39
Şekil 4.8 : (a) 3×3 maske geçirilmiş averaj filtresi (b) 5×5 maske geçirilmiş averaj filtresi.....	39
Şekil 4.9 : (a) 3×3 maske geçirilmiş averaj filtresinin değerleri (b) 5×5 maske geçirilmiş averaj filtresinin değerleri.	40
Şekil 4.10 : (a) 0.02 derece ile tuz & biber gürültüsü eklenmiş görüntü (b) 3×3 maskesi ile medyan filtresi geçirildikten sonra oluşan görüntü.	41
Şekil 4.11 : (a) 0.1 derece ile tuz & biber gürültüsü eklenmiş görüntü (b) 3×3 maskesi ile medyan filtresi geçirildikten sonra oluşan görüntü.	41
Şekil 4.12 : (a) Çalışmada kullanılmış renkli görüntülerden biri (b) $\alpha = 0.2$ kullanılarak laplace filtresi geçirildikten sonra oluşan görüntü.	42
Şekil 4.13 : (a) Çalışmada kullanılmış renkli görüntülerden biri (b) $\alpha = 0.5$ kullanılarak laplace filtresi geçirildikten sonra oluşan görüntü.	42
Şekil 4.14 : (a) Çalışmada kullanılmış renkli görüntülerden biri (b) $\alpha = 0.2$ kullanılarak unsharp filtresi geçirildikten sonra oluşan görüntü.....	43
Şekil 4.15 : (a) Çalışmada kullanılmış renkli görüntülerden biri (b) $\alpha = 0.5$ kullanılarak unsharp filtresi geçirildikten sonra oluşan görüntü.....	43
Şekil 4.16 : (a) 122 eşik değeri sonucunda oluşan görüntü (b) 145 eşik değeri sonucunda oluşan görüntü.	44
Şekil 4.17 : (a) Orijinal görüntü (b) 0.6 sınır eşik değeri sonucunda oluşan görüntü.	44
Şekil 4.18 : (a) Orijinal görüntü (b) Otsu eşik metoduna göre 0.4863 sınır eşik değeri sonucunda oluşan görüntü.	45
Şekil 4.19 : Çalışmada kullanılan gri seviyeli görüntülerden biri.	46
Şekil 4.20 : Eşik değer sınırı 0.4 olan binari görüntü.....	46
Şekil 4.21 : Strel (disk ,3) değer sınırı ile erozyon işlemi sonucunda oluşan binari görüntü.....	46
Şekil 4.22 : Strel (disk,3) değer sınırı ile açma işlemi sonucunda oluşan binari görüntü	47
Şekil 4.23 : Açma kapama işlemi sonucunda oluşan görüntü.	47

Şekil 4.24 : Açma kapama sonucunda oluşan görüntü üzerinden keskinleştirme filtresi geçirildikten sonra oluşan görüntü.....	48
Şekil 4.25 : Morfolojik işlemler sonucunda çıkan detay görüntüsü.....	48
Şekil 4.26 : Çalışmada kullanılan gri seviyeli görüntülerden biri.....	49
Şekil 4.27 : Prewitt gradientinin G_x yönünde oluşan görüntü	49
Şekil 4.28 : Prewitt gradientinin G_y yönünde oluşan görüntü	49
Şekil 4.29 : Prewitt gradientinin $G_x + G_y$ yönünde oluşan görüntü..	50
Şekil 4.30 : Sobel gradientinin $G_x + G_y$ yönünde oluşan görüntü.....	50
Şekil 4.31 : Sobel kenar belirleme operatöründe 45 derece yön ve 150 eşik değer sonucu oluşan görüntü	51
Şekil 4.32 : Prewitt kenar belirleme operatöründe 0.1 eşik değer sonucu oluşan görüntü	51
Şekil 4.33 : Sobel kenar belirleme operatöründe 0.1 eşik değer sonucu oluşan görüntü	51
Şekil 4.34 : Log(Laplacian of Gauss) kenar belirleme operatörü sonucu oluşan görüntü	51
Şekil 4.35 : Canny kenar belirleme operatöründe 0.1 eşik değer sonucu oluşan görüntü	52
Şekil 4.36 : Arayüz tasarımı.....	53

SEMBOL LİSTESİ

$f(x,y)$	: Görüntü fonksiyonu (giriş)
$g(x,y)$	: Görüntü fonksiyonu (çıkış)
L	: Gri değer sayısı
t	: Eşik değer
T	: Dönüşüm fonksiyonu
$P(x)$	: Normalize histogram fonksiyonu
μ	: Görüntüdeki gri seviyeli değerlerin ortalaması
c	: Sabit sayı
σ^2	: Varyans
f_{ij}	: Filtre matrisinin katsayısı
n	: Filtre matrisinin boyutları
F	: Filtre matrisinin elemanları toplamı
∇f	: Gradient
M	: Gradient'in genliği
α	: Alfa parametresi
θ	: Kenarın yönü
G_x	: X yönündeki gradient filtresi
G_y	: Y yönündeki gradient filtresi
δ	: Genişleme operatörü
ε	: Aşındırma operatörü
σ	: Standart sapma

DİJİTAL GÖRÜNTÜ İŞLEME TEKNİKLERİ KULLANILARAK GÖRÜNTÜLERDEN DETAY ÇIKARIMI

ÖZET

Günümüzde bilgisayar teknolojisinin gelişmesi ile birlikte dijital görüntü işleme teknikleri yaygın bir şekilde kullanılmaktadır. Dijital görüntü işleme uzaktan algılamanın uygulamaları arasına girmektedir. Dijital görüntü işleme çevreyle hiçbir etkileşim olmaksızın teknolojik araçlar yardımıyla görüntüler içerisinde bulunan objelerin öznitelikleri hakkında bilgi edinmemizi sağlar. Dijital görüntü işleme teknikleri ile iyileştirilmiş veya daha farklı görüntüler elde edilmekle birlikte görüntülerden öznitelik ve anlamlı bilgi çıkarımı yapılabilmektedir. Dijital görüntüler sayesinde görüntü yorumlamaya olanaklı olan mekansal veriler, kentsel planlama, çevresel değişimin incelenmesi, coğrafi analiz verilerinden bilgisayar teknolojisi yardımıyla birçok analiz yapılmaktadır. Bu işlemler sayesinde şu ana kadar birçok yaklaşım ve yöntem sunulmuştur. Bulunan yöntemlerin görüntülere özgü yöntemler olmasından dolayı görüntü işleme konusu birçok farklı türdeki görüntüler için ayrı ayrı yöntemler olarak sunulmuştur.

Bu çalışma kapsamında birinci bölümde dijital görüntü işlemenin tarihçesi ve uygulama alanları hakkında bilgi verilmiştir.

İkinci bölümde, görüntü elde etme ve görüntü işleme tekniklerinin temel kavramlar üzerinden örnek görüntüler verilerek anlatılmıştır. Özellikle bu tez çalışmasında, en yaygın kullanılan görüntü işleme tekniklerine değinilmiştir.

Üçüncü bölümde MATLAB programlama dilinin avantajlarına değinilmiştir. Kod yazma ortamı ve arayüz tasarlama ortamı olan 'Guide' araç fonksiyonu tanıtılmıştır.

Uygulama bölümünde, tez çalışmasında kullanılan görüntülerden ve uygulanan yöntemlerden bahsedilmiştir. Nokta işleme teknikleriyle görüntüler hakkında bilgi sağlamakla birlikte kontrast ve parlaklık ayarlarına değinilmiştir. Çeşitli filtreler geçirilerek görüntü zenginleştirme yöntemleri denenmiştir. Görüntülerin gri seviyelerindeki ani değişikliklerin olduğu bölgelerde kenar belirleme algoritmaları kullanılmıştır. Bu kenar algoritmalarından birinci türeve ve ikinci türeve dayalı kenar belirleme algoritmalarından belli eşik değeri altında çıkarılan detaylar yorumlanmıştır. Morfolojik yöntemlerin çeşitli operatörleri kullanılmıştır ve daha önceki çalışmalara bu operatörler eklenerek detay ve sınırların çıkarılması çalışılmıştır. Bu kapsamda dijital görüntü işleme tekniklerinin klasik yapısal programlama dilleri kullanılarak gerçekleştirilmesi; yoğun matematiksel işlemlerin olması, veri sayısının yüksek ve algoritma karışıklığının olması gibi nedenlerden dolayı zorlaşmaktadır. Bu sorunu aşmak için MATLAB programlama dili tercih edilmiştir. MATLAB'in dijital görüntü işleme fonksiyonları kullanılarak görüntü işlemeye yönelik uygulama arayüz tasarımı gerçekleştirilmiştir. Son olarak beşinci bölümde değerlendirmeler ve öneriler verilmiştir.

FEATURE EXTRACTION FROM IMAGES BY USING DIGITAL IMAGE PROCESSING TECHNIQUES

SUMMARY

Today, with the development of computer technology, digital image processing techniques are being widely used. Digital image processing is one of the remote sensing applications. The availability of digital image processing techniques have significantly enhanced the possibilities for photogrammetric image measurement and analysis. As in other fields, digital images not only enable new methods for the acquisition, storage, archiving and output of images, but most importantly for the automated processing of the images themselves. Digital image processing without any interaction to environment allows us to obtain information about the features of objects that contained inside of the images by means of technological devices. By digital image processing, improved or with more different images that have been obtained, it can be carried out meaningful information and feature extraction from images. Owing to the digital images, many analysis are being carried out on spatial data; which give the possibility of interpretation in images, urban planning, examination of environmental change, geographical data analysis thanks to computer technology. Owing to these processes, many approaches and methods have been put forward. Due to the found methods that related to the specific image methods, image processing subject has been presented for the many different types of images as separated methods.

Within the scope of this thesis, information about the history of digital image processing and application areas of have been given in the first section.

In the second part, image acquisition and image processing techniques are described in the sample images through the basic concepts. Especially in this master thesis, there has been focused on the most commonly used image processing techniques. Digital image processing techniques, is divided into three parts in this thesis.

First part is point operations. These are histogram, histogram equalization, contrast stretching and thresholding. Histogram provides the frequency distribution of the pixel values in the image. It displays the absolute or relative frequency of each pixel value either in tabular or graphical form. The most important parameters of a histogram are; relative and absolute frequencies, minimum and maximum pixel value of the image, contrast and mean of pixel values. While minimum and maximum pixel values define the image contrast, the mean is a measure of the average intensity of the image. Histogram equalization is a non-linear process aimed to highlight image brightness in a way particularly suited to human visual analysis. The cumulative frequency function is calculated from the histogram of the original image. Contrast stretching is linear interpolation between $g_{\min}$ and $g_{\max}$. Minimum and maximum pixel values can be derived from the histogram, or defined interactively. Thresholding is used to clearly differentiate pixel values which belong to different object classes to separate object and background. Thresholding is often a pre-processing stage prior to segmentation.

The second part of techniques is image enhancement methods. These methods are image smoothing, image sharpening and edge detection. In image processing, many filter operations are applied to an image by performing a special operation called convolution with a matrix called a kernel. Kernels are typically 3x3 square matrices, although kernels of size 2x2, 4x4, and 5x5 sometimes used. The values stored in the kernel directly relate to the results of applying the filter, and filters are characterized solely by their kernel matrix. Convolution can be described as a function that is the integral or summation of two component functions, and that measures the amount of overlap as a function is shifted over the other. Filters can be used for denoising signal and images. Many different filters can achieve this purpose and the optimal filter often depends on the particular requirements of the application. One such filter is called Gaussian, so named because the filter's kernel is a discrete approximation of the Gaussian distribution. The Gaussian filter is known as a 'smoothing' operator, as its convolution with an image averages the pixels in the image, affectively decreasing the difference in value between neighboring pixels. The σ parameter is standard deviation of the Gaussian, and be adjusted according to the desired distribution. Another smoothing filter is average filter. Average filter is reducing the amount of intensity variation between one pixel and the next. It is often used to reduce noise in images. The idea of mean filtering is simply to replace each pixel value in an image with the mean ('average') value of its neighbors, including itself. This has the effect of eliminating pixel values which are unrepresentative of their surroundings. Mean filtering is usually thought of as a convolution filter. Like other convolutions it is based around a kernel, which represents the shape and size of the neighborhood to be sampled when calculating the mean. The other filter is median filter. The non-linear median filter performs good smoothing while retaining sharp edges. The median filter is not based on convolution. Instead the median value (as opposed to the mean) output pixel value. The output image therefore consists only of pixel values which exist in the input image. This property is essential for the filtering of images which consist of attributes or special palettes instead of intensities. The median filter is a member of the group of rank-order filters. The median is the centre of a rank-ordered distribution. Laplacian operator is the one of the image sharpening filters. Laplacian operator is a template which implements second order differencing. The second order differential can be approximated by the difference between two adjacent first order differences. The other technique of the image enhancement is edge detection. Many approaches to image interpretation are based on edges, since analysis based on edge detection in insensitive to change in the overall illumination level. Edge detection highlights image contrast. Detecting contrast, which is difference in intensity, can emphasize the boundaries of features within an image, since this is where image contrast occurs. There are two differential filters in the edge detection. Taylor series analysis reveals that differencing adjacent points provides an estimate of the first order derivative at a point. An alternative to taking the maximum is simply to add the results of the two templates together to combine horizontal and vertical edges. There are of course more varieties of edges and it is often better to consider the two templates as providing components of an edge vector: the strength of the edge along the horizontal and vertical axes. These give components of a vector and can be added in a vectorial manner. The edge magnitude is the length of the vector, and the edge direction is the vector's orientation. The first order differential edge filters are Sobel and Prewitt edge filters. Canny edge detection aims to reduce the response to noise. This can be affected by optimal smoothing; Canny was the first to demonstrate that Gaussian filtering is optimal for edge detection. Moreover second order differential

filter is Laplacian of Gauss filter. It illustrates the sensitivity to noise of the Laplacian filter, hence minor intensity changes are interpreted as edges. The second derivative of the Gaussian function is regarded as an optimal edge filter which combines smoothing properties with edge extraction capabilities.

The third part of techniques is Morphology operations. Morphology was originally developed for binary images and was extended to include grey-level data. The word morphology concerns shapes: in mathematical morphology we process images according to shape, by treating both as sets of points. There are several morphology operators. These are erosion, dilation, opening, closing. Erosion, which leads to the shrinking of regions. Dilation, which yields to the extension of connected regions. Opening, achieved by an erosion followed by dilation. Small objects are removed. Closing, the reverse of opening. Dilation is followed by erosion in order to close gaps between objects.

The third part has been referred to the advantage of the MATLAB programming language. As a code-write and interface design space, 'Guide' tool function has been introduced.

The images used and the methods applied in this thesis are mentioned in the fourth section. With the point processing technique, information about the images is provided. In addition to this, contrast and brightness settings have been experienced. Image enhancement methods have been tried through various filters processes. Edge detection algorithms have been used in areas where there are sudden changes in gray level images. Features derived under a certain threshold of edge defining algorithms based on the first and second derivative of all these edge algorithms have been interpreted. Various operators of morphological methods have been used and by adding these operators to the previous studies, extraction of boundaries and features have been studied. Within this scope, due to reasons like the higher number of data and algorithmic complexity, intensive mathematical process, classic structure of language programming of digital image processing techniques is difficult. In order to cope with these challenges, MATLAB programming language has been preferred. Application interface is designed for digital image processing techniques with using MATLAB image processing toolbox. And finally, in the fifth section, evaluations and recommendations have been given.

1. GİRİŞ

Dijital görüntülerin ilk uygulaması denizaltı kablosu tarafından gönderilen gazete resimlerinin sayısallaştırılması ile başlanmıştır. 1920'li yıllarda iletişim verilerinin başlanması yanı sıra bir haftadan daha fazla sürede ulaştırılan görüntüler çözümleme yöntemi ile üç saate düşürülmüştür. Gelişen teknoloji ile birlikte ileri düzeyde yüksek kapasiteli bilgisayarlar ve uzay programlarının ortaya çıkmasıyla dijital görüntü alanının potansiyelini arttırdı. 1960'lı yıllarda güç laboratuvarlarının kurulmasıyla uzay araştırmalarında yeni bir boyuta varıldı. Uzay programlarındaki uygulamalara ek olarak dijital görüntü işleme teknikleri günümüzde de birçok alanda önem kazanmıştır. Bu alanlar arasında; dijital görüntülerden anlamlı sonuçlar çıkarma çalışmaları gibi bilgisayarda görme (computer vision) alanında, uydu görüntüleri üzerinden nüfus yoğunluğu, yerleşim yerleri, hava tahmini ve gözlem gibi uzaktan algılama uygulamalarında, medikal görüntülemelerde, astronomi uygulamaları gibi daha birçok alanda da uygulamalar yer almaktadır. Bunların içerisinde en yaygın kullanım alanlarından biri de yersel fotogrametridir. Yersel fotogrametride kullanılan görüntülere dijital görüntü işleme teknikleri uygulanarak görüntüler iyileştirilerek kaliteleri arttırılabilmektedir.

Dijital görüntü işleme uygulamalarını geliştirmek ve gerçekleştirmek için birçok programlama dili ve programlar bulunmaktadır. Dijital görüntü işleme algoritmalarına yönelik fonksiyon kütüphaneleri, bu programlama dillerine eklenmiş ve bu işlemlere yönelik özel programlama dilleri geliştirilmiştir.

MATLAB; kontrol sistemlerinden haberleşmeye, finansal analizlerden mühendislik uygulamalarına kadar birçok alana yönelik hazır fonksiyona sahip teknik bir programlama dilidir. MATLAB'in görüntü işlemeye yönelik kütüphanesi mevcuttur. Bu kütüphane sayesinde, MATLAB dijital görüntü işleme uygulamaları geliştirilmesinde tercih edilen programlama dillerinden biridir.

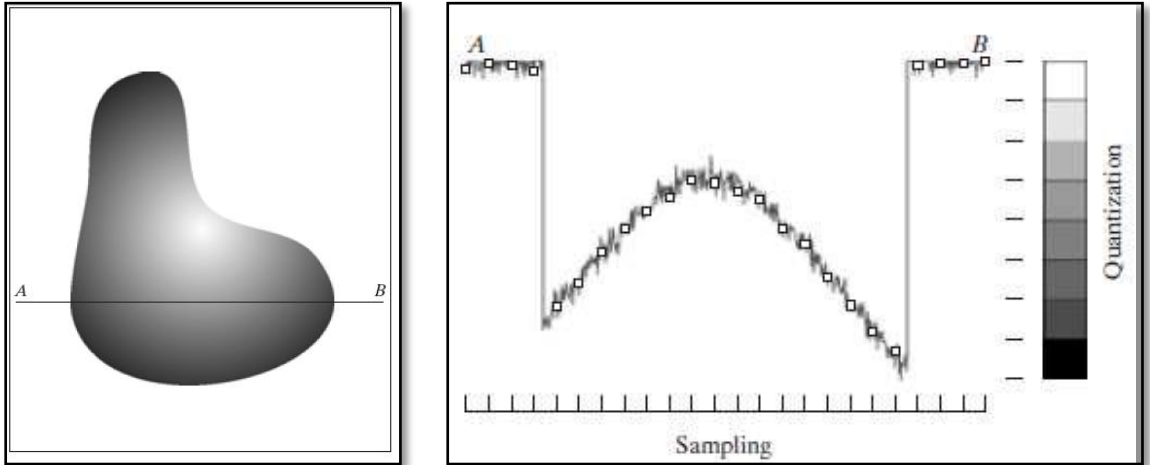
Bu çalışmada, dijital görüntü işleme teknikleri kullanılarak anlamlı ve öznetelik bilgilerinin çıkarımının yanı sıra görüntülerin homojen bir şekilde ayrıştırılması amaçlanmıştır. Bu detayların çıkarılması için birden fazla olmak üzere farklı zaman

aralıklarında ve mekanlarda çekilen görüntüler kullanılmıştır. Dijital görüntü işleme konusunda önemli bir yere sahip olan görüntü zenginleştirme tekniklerine değinilmiştir. Çeşitli nedenlerle üzerine gürültü eklenmiş, bulanıklaştırılmış ve kontrast ayarı bozulmuş olan görüntüler gürültü temizleme ve netleştirme algoritmaları ile optimum seviyeye getirilmesi sağlanmıştır. Bunlara ek olarak morfolojik işlemler sonucunda belli bir eşik değeri altında kenar belirleme yöntemleri uygulanmış ve değerlendirilmiştir.

Bu çalışmanın birinci bölümde dijital görüntü işlemenin tarihçesi hakkında bilgi verilerek uygulama alanları anlatılmıştır. İkinci bölümde ise görüntü elde etme ve dijital görüntü işleme tekniklerinin temel kavramları üzerinden örnek görüntüler verilerek anlatılmıştır. Üçüncü bölümde MATLAB programlama dilinin avantajlarına anlatılarak kod yazma ortamı ve arayüz tasarlama ortamı olan ‘Guide’ araç fonksiyonu tanıtılmıştır. Uygulama bölümünde, farklı görüntülere farklı filtreler, eşikleme, morfolojik işlemler ve kenar belirleme yöntemleri uygulanarak sonuç görüntüleri değerlendirilmiştir. Sonuç kısmında, uygulama bölümündeki sonuçlar karşılaştırılarak dijital görüntü işleme tekniklerinin etkinliği/kullanılabilirliği değerlendirilmiştir.

2. DİJİTAL GÖRÜNTÜ İŞLEME

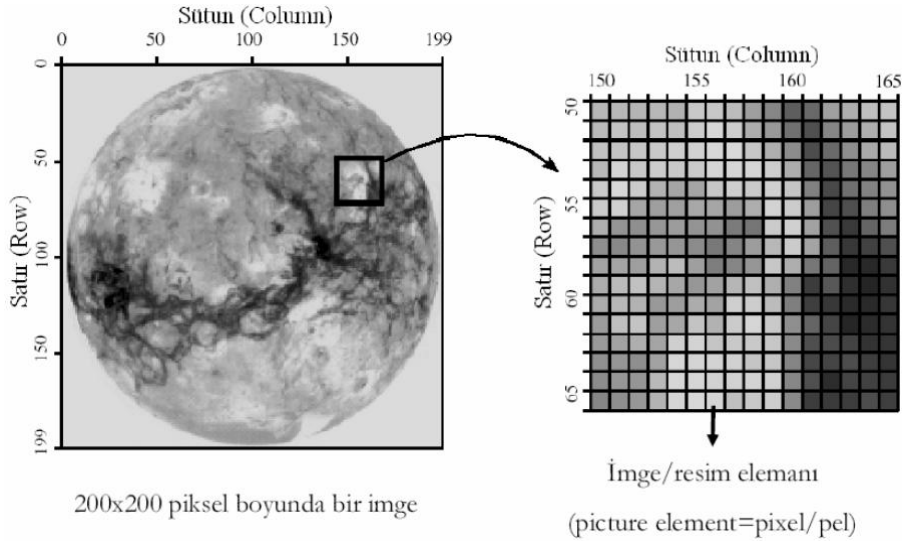
Dijital görüntü işlemenin ilk adımı görüntü elde etmektir. Dijital görüntü işleme sensörlerden gelen görüntünün bilgisayara aktarılıp üzerinde herhangi bir işlem yapılmasının ardından görüntüleyici çıkışa iletilmesi işlemidir. Görüntü elde etmede kullanılan yazılım ve donanım görüntü kaynağına bağlı olarak farklılık gösterebilir. Gerçek dünyadaki üç boyutlu nesnelerin dijital görüntülere dönüştürülmesinde; CCD kameralar, kızılötesi kameralar, ultrason cihazları, X-ray, manyetik rezonans görüntüleme araçları ve uydu gibi kaynaklardan sağlanır. Sensörün çıkışı, genliği ve uzaysal davranışı görüntülenen olguya bağlı değişen sürekli voltaj dalgasıdır. Dijital görüntü elde etme için bu sürekli veriyi dijitize etmek gerekir. Bunun için iki işlem gereklidir. Bunlar; örnekleme ve kuantulama (nicemleme). Elde edilen görüntü fonksiyonu $f(x,y)$ hem koordinatlarında hem de genlikte sürekli olabilir. Dijital görüntü için ikisini de ayrı yapmak gerekir. Eğer koordinat değerleri dijitize ediliyorsa **örnekleme** eğer $f(x,y)$ genlik değerleri dijitize ediliyorsa **kuantulama (nicemleme)** denir (Şekil 2.1) [1].



Şekil 2.1 : Örnek bir görüntünün örnekleme ve nicemlemesi.

2.1 Dijital Görüntü

Temel yapısı Şekil 2.2’de gösterilen bir dijital görüntü, satır ve sütun indisleri görüntü içerisinde herhangi bir noktayı tanımlayan elemanlardan meydana gelmiş bir matris olarak göz önüne alınabilir. Bu matrisin her bir elemanının barındırdığı dijital değer veya değerler, o noktanın rengi hakkındaki bilgiyi içerir. Bu dijital dizinin veya matrisin her bir elemanına görüntü elemanı veya piksel denir [1].



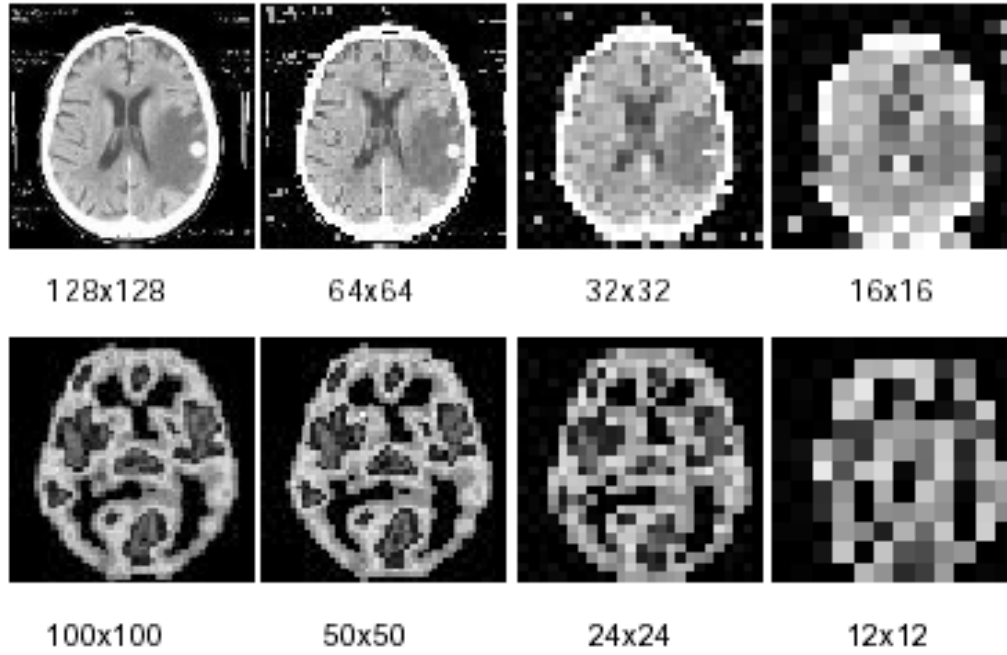
Şekil 2.2 : Örnek bir dijital görüntü ve belli bir kısımdaki piksellerin büyütülmüş hali.

Gerçek yaşamda, bir görüntü basit iki değişkenin bir fonksiyonu olarak tanımlanır. $f(x,y)$ gibi bir fonksiyonla ifade edilen bir resimde f bir şiddet birimi (örneğin parlaklık) ve x ve y değişkenleri ise görüntünün gerçek koordinatlarıdır [2]. Dijital bir görüntü, elemanları, uzaydaki x,y konumlarına karşılık gelen noktaların $f(x,y)$ parlaklık değerlerini içeren matristir. Dijital görüntü = $f(x,y)$

$$f(x, y) = \begin{bmatrix} f(0, 0) & f(0, 1) & \cdots & f(0, N - 1) \\ f(1, 0) & f(1, 1) & \cdots & f(1, N - 1) \\ \vdots & \vdots & \ddots & \vdots \\ f(M - 1, 0) & f(M - 1, 1) & \cdots & f(M - 1, N - 1) \end{bmatrix} \quad (2.1)$$

Burada; R,C = Görüntü boyutu(toplam satır sayısı x toplam sütun sayısı)

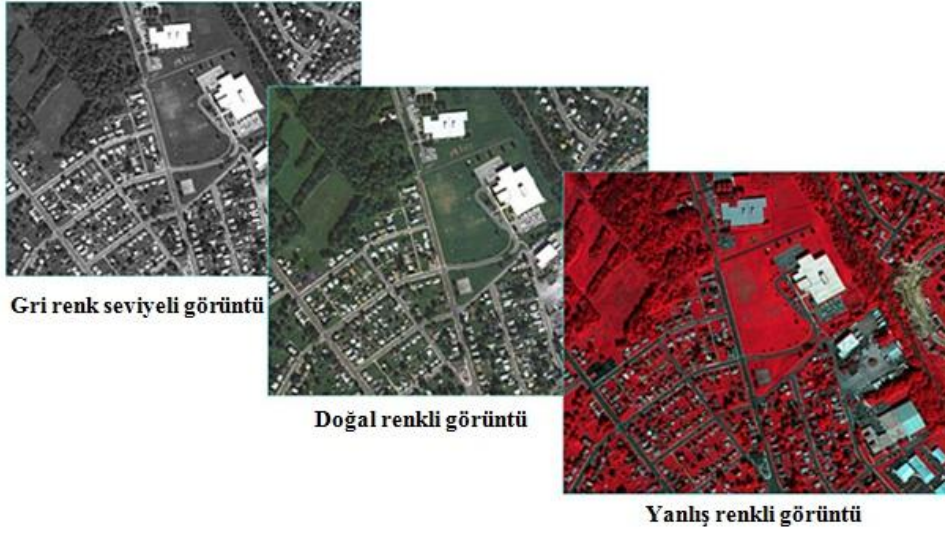
Çözünürlük, bir görüntüdeki yatay piksel sayısı ve dikey piksel sayısının çarpımı olarak ifade edilir [1]. Çözünürlük görüntünün boyutları ile ilgili bir bilgi içermez, çünkü pikseller için boyut söz konusu değildir. Ancak çözünürlük düştükçe (görüntüdeki piksel sayısı azaldıkça) görüntüdeki pikselasyon artar. Pikselasyon olayı Şekil 2.3’de gösterilmiştir.



Şekil 2.3 : Görüntü üzerindeki pikselasyon.

Genel olarak dijital görüntüler 4 farklı grupta toplanmaktadır (Şekil 2.4).

- 1) **Binari Görüntü:** 0 veya 1 olmak üzere 2 değer (siyah ve beyaz) alırlar.
- 2) **Gri Renk Seviyeli Görüntü:** Tek renkli görüntülerdir. Renk bilgisi içermez sadece parlaklık bilgisi içerirler.
- 3) **Renkli Görüntü:** bilgisayar ekranlarında 24 bitlik veri olarak görüntülenir. Görüntüleme R(Kırmızı), G(Yeşil), B(Mavi) kodlanmış aynı objeye ait üç adet gri düzeyli görüntünün üst üste ekrana iletilmesi ile oluşur.
- 4) **Çok Spektrumlu Görüntü:** Görünür spekturumun dışındaki bölgelerden alınan ve yanlış renkli görüntü olarak da adlandırılan görüntülerdir.



Şekil 2.4 : Dijital görüntü çeşitleri.

2.2. Dijital Görüntü İşleme Teknikleri

2.2.1. Nokta işleme teknikleri

Nokta işleme tekniklerinde sonuçta elde edilen görüntünün bir (x, y) noktasındaki değeri, orijinal görüntünün aynı (x, y) noktasındaki değerine bağlıdır.

$$g(x,y) = T[f(x,y)] \quad (2.2)$$

Buradaki T işlevi, doğrudan (x,y) pikselini işleyebileceği gibi, (x,y) pikselinin komşuluklarını da hesaba katabilir. Nokta işlemleri görüntü işlemlerinin temelini oluşturur. Nokta işleme algoritmaları bir pikselin değerinde değişiklikler yaparlar ve yeni görüntü alırlar (Bazen yerlerinde de değişiklikler olabilir). Diğer piksel değerleri ise değişikliğe uğramazlar. Bireysel pikselin değeri algoritma tarafından değiştirildikten sonra tekrar yerine yerleştirilir. Sonucunda ise yeni bir görüntü ortaya çıkar. Nokta işleme algoritmaları tek piksel üzerinde işlem yaparlar. Görüntü geliştirme işlemi sırasında orijinal görüntü taranır, her pikselin gri düzeyli değerine karşılık gelen yeni değer tablodan bulunur ve yeni görüntüde yerine yazılır. Nokta işleme tekniklerinde bir piksel sadece bir defa kullanıldığı için yeni piksel değerini orijinal görüntüdeki yerine yazmak mümkündür [2].

2.1.1.1 Negatif görüntü

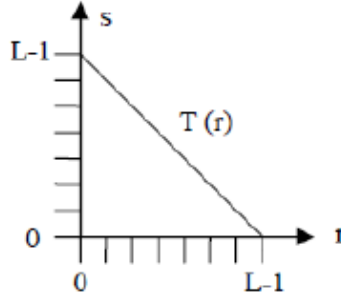
Bir görüntünün negatifi, Şekil 2.5'te görülen $s = T(r)$ dönüşüm fonksiyonu ile elde edilir. Bu fonksiyon aşağıdaki eşitlikle ifade edilebilir:

$$S = T(r) = L - 1 - r \quad (2.3)$$

Burada; Pixel.red=255-Pixel.red;

Pixel.green=255-Pixel.green;

Pixel.blue=255-Pixel.blue gibi. Burada L değeri gri değer sayısını göstermektedir.



Şekil 2.5 : Negatif görüntü için dönüşüm fonksiyonu.

2.1.1.2. Görüntü histogramı

Histogram, her bir gri ton seviyesinin $[0,255]$ görüntüdeki bulunma sıklığını (frekansını) gösterir. Görüntü histogramı, görüntünün her bir noktasındaki piksellerin tespiti ile bu piksellerin sayısının ne olduğunu gösterir. Bu sayede histogram üzerinden görüntü ile ilgili çeşitli bilgilerin çıkartılması sağlanır. Fakat görüntünün aydınlık-karanlık bölge değerlerinden görüntü hakkında genel bilgiler elde edilebilir. Uygulanmak istenen eşik değerleri tahmin edilebilir [3].

Histogramda piksel yoğunluk değerleri x eksenini doğrultusunda gösterilirken y eksenini doğrultusunda da o yoğunluğa sahip piksel sayısı verilir (Şekil 2.6). Böylece grafikte verilen diyagramdan görüntünün yoğunluk dağılımı olan parlaklık ve karanlık imgesi ile karışıklık hakkında bilgi edinilebilir (Şekil 2.7 ve Şekil 2.8). Özellikle eşikleme yöntemleri için görüntünün histogramından faydalanılır. Histogram (2.4) eşitliği ile ifade edilir.

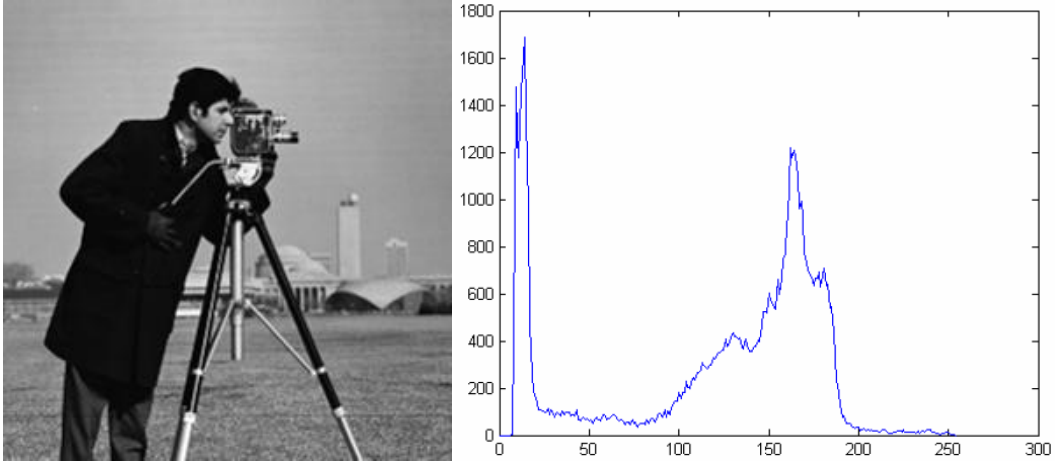
$$h(r_k) = n_k \quad (2.4)$$

$r_k = k.$ gri seviye

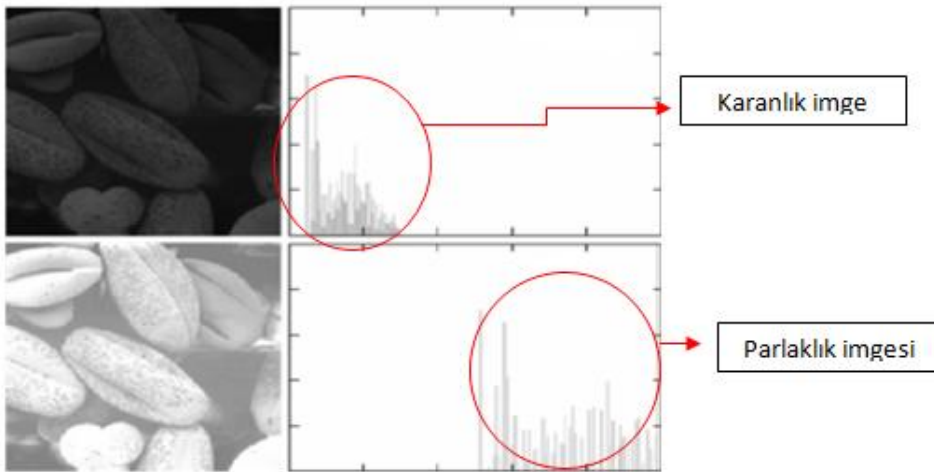
$n_k = k.$ gri seviyedeki toplam piksel sayısı ($k = 0, 1, \dots, L-1$)

Histogram normalizasyonu, (r_k) grilik seviyesindeki piksel sayısının, toplam piksel sayısına bölünmesiyle ifade edilir. Histogram normalize edildiğinde, gri seviyelerin görüntü içerisindeki bulunma olasılıkları elde edilir.

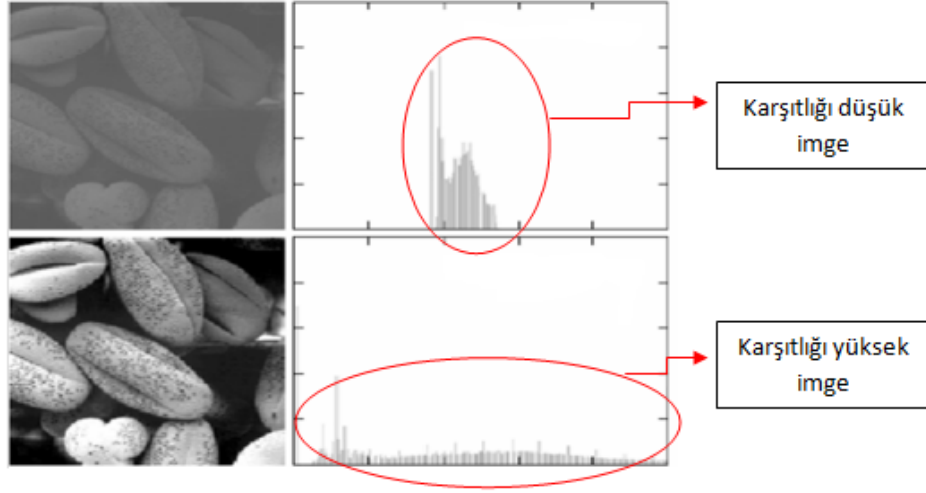
$$P(r_k) = n_k / n \quad (2.5)$$



Şekil 2.6 : Gri seviyeli görüntü ve histogramı.



Şekil 2.7: Gri seviyeli görüntü ve histogramında karanlık ve parlaklık imgesi.



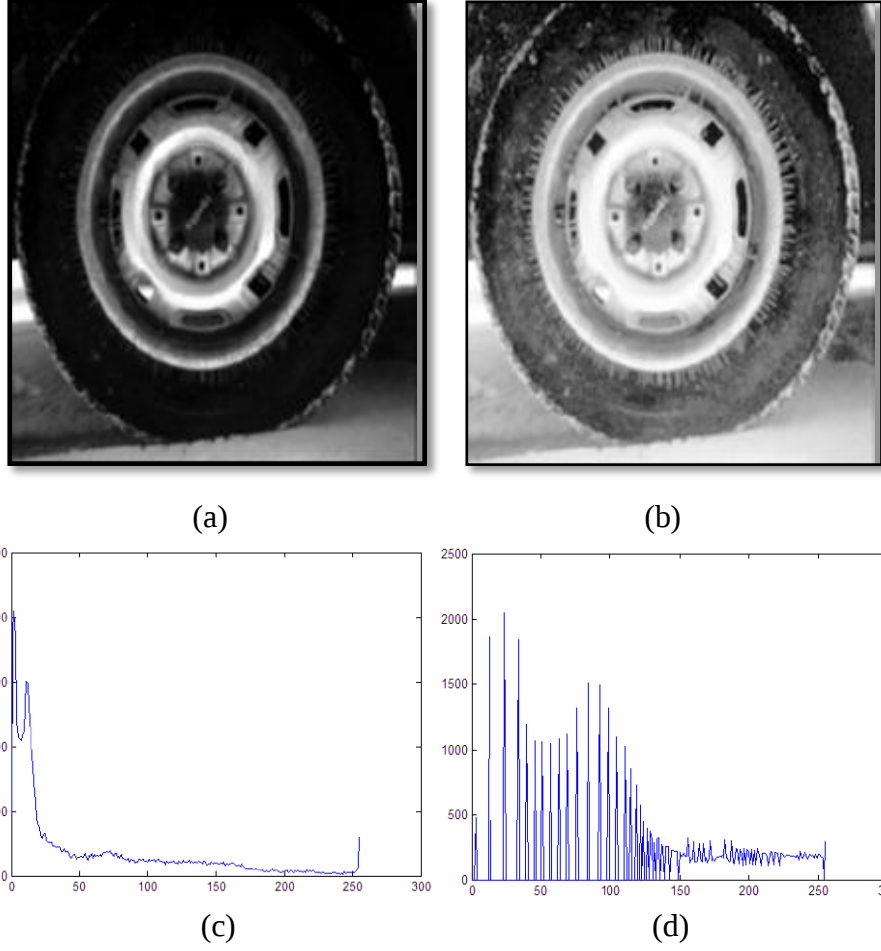
Şekil 2.8 : Gri seviyeli görüntü ve histogramında karşıtlığı düşük ve yüksek imge.

2.2.1.3 Histogram eşitleme

Histogram eşitleme kontrast zenginleştirme yöntemleri arasına girmektedir. Histogram eşitlemesi ile görüntüde parlaklık en çok toplandığı alana en büyük kontrast zenginleştirme uygulanmaktadır. Histogram eşitleme, görüntünün tümüne uygulanabileceği gibi sadece belli bir bölgesine de uygulanabilir. Tüm görüntüye uygulanırsa global histogram eşitleme, görüntünün belli bir bölgesine uygulandığında ise lokal histogram eşitleme adını alır. Bu yöntem histogramı dar olan görüntüler ya da görüntü içindeki bölgeler için daha iyi sonuç verir [4,5].

Görüntünün olasılık dağılım fonksiyon doğrusallaştırılmaktadır. Histogram eşitlemenin uygulanması aşağıda verilen adımlarla gerçekleştirilir:

1. Görüntünün histogramı bulunur.
2. Histogramdan yararlanılarak kümülatif histogram bulunur. Kümülatif histogram, histogramın her değerinin kendisinden öncekiler ve kendisinin toplamı ile elde edilen değerleri içeren grafiklerdir.
3. Kümülatif histogram değerleri normalize edilip (toplam piksel sayısına bölünerek), yeni görüntüde olmasını istediğimiz max. renk değerleri ile çarpılır, çıkan değer tam sayıya yuvarlatılır. Böylelikle gri seviye değerleri elde edilmiş olur.
4. Orijinal gri seviyeli değerleri ile üçüncü adımda elde edilen gri seviye değerleri birbirine karşılık düşürülür ve yeni histogram elde edilir [2,6].



Şekil 2.9 : Bir görüntüye histogram eşitleme yönteminin uygulanması ve sonucu (a) orijinal görüntü, (b) histogram eşitleme uygulanmış görüntü, (c) orijinal görüntünün histogramı (d) histogram eşitleme uygulanmış görüntünün histogramı.

2.2.1.4 Parlaklık ayarlama

Gri değerlerinin ağırlıklı ortalamasını ve varyansını hesaplayarak görüntünün genel görünümüyle ilgili iki adet özellik elde edebiliriz.

Gri değerlerinin ağırlıklı ortalaması görüntünün ortalama parlaklığının, varyans da kontrastın bir ölçüsüdür. Görüntünün parlaklığını ayarlamak için görüntünün piksellerinin gri değerlerine sabit c sayısı eklenebilir (Şekil 2.10).



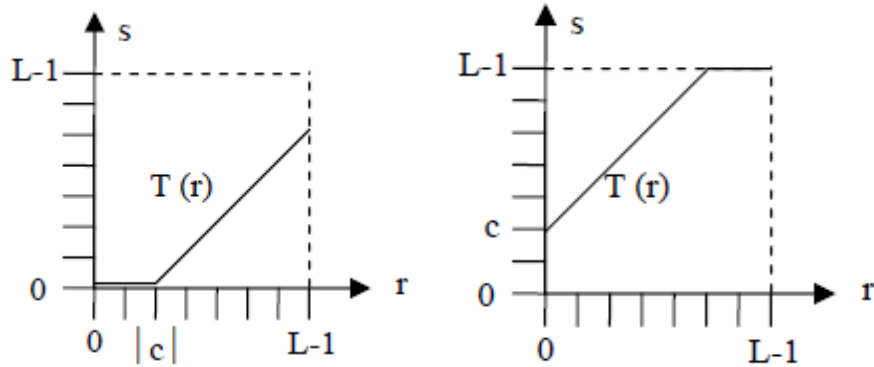
Şekil 2.10 : Sabit sayıya göre parlaklık ayarı.

Bu durumda dönüşüm fonksiyonu,

$$s = T(r) = r + c \quad (2.6)$$

şeklindedir. Gri değerleri $[0, L - 1]$ aralığındaki dijital görüntü için $c < 0$ durumunda $s < 0$ bulunuyorsa $s = 0$, $c > 0$ durumunda $s > L - 1$ bulunuyorsa $s = L - 1$ kabul edilir. Bu iki durum için dönüşüm fonksiyonlarının grafiği Şekil 2.11’de görülmektedir.

Görüntü histogramı $c < 0$ durumunda şeklini değiştirmeden sol tarafa doğru, $c > 0$ durumunda ise sağ tarafa doğru kayar. $[0, L - 1]$ sınırlarını aştığı hallerde, 0 veya $L - 1$ gri değerlerinde birikme olur [7].



Şekil 2.11 : (a) $c < 0$ için dönüşüm fonksiyonu, (b) $c > 0$ için dönüşüm fonksiyonu.

2.2.1.5 Kontrast geliştirme

Sözlük anlamı “karşıtlık” olan kontrast, görüntünün ayırt edilebilirliğini ifade etmek için kullanılır. İnsan gözünün gri değerlerini ayırt etme kapasitesi ortalama olarak otuz adet gri değeriyle sınırlıdır. Bir görüntüdeki en düşük ve en yüksek gri değerleri arasındaki farkın bu sınırdan düşük olması durumunda, bu görüntüdeki detayların insan gözü tarafından yeterince algılanamadığı söylenebilir. Bu tipteki görüntülerin

histogramlarında, gri düzeyli bir görüntünün dar bir aralığında çok sayıda pikselin yoğunlaştığı görülür. Bu görüntülere düşük kontrastlı görüntüler adı verilir [7].

Kontrast kavramı bir görüntü içindeki grilik seviyesi farklarının genliğini ifade etmektedir. İnsan incelemesine uygun olmayan düşük kontrastlı görüntüler çeşitli dönüşüm fonksiyonları yoluyla insan için daha uygun duruma getirilebilirler. Farklı türdeki düşük kontrastlı görüntüler için farklı fonksiyonlar kullanmak gerekir. Koyu tonların ağırlıklı olduğu bir görüntüyü geliştirmek için gri değerlerini açık tonları gösteren yüksek değerlere doğru kaydırmak gerekir (Şekil 2.12) [7].

$$g(x,y) = T[f(x,y)] \quad (2.7)$$

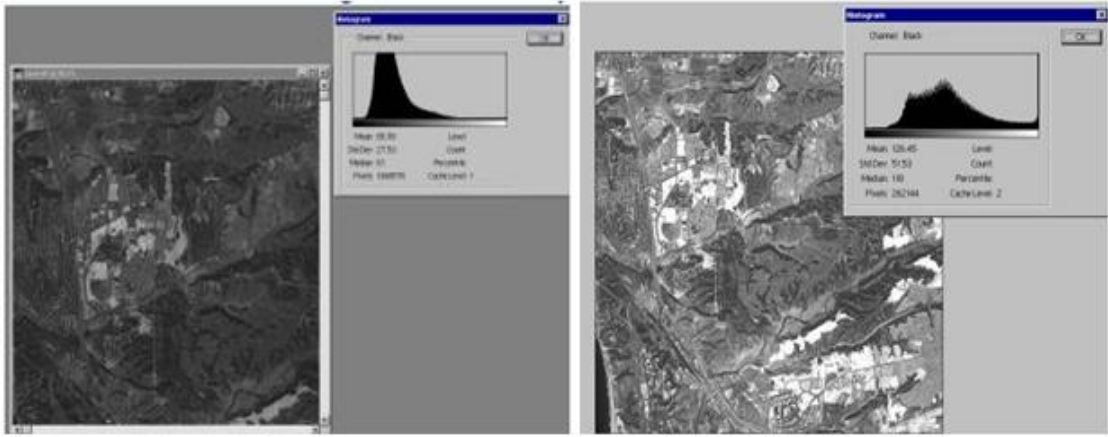
$=af(x,y)$ $s = ar$ $a > 1$ ise karşıtlık artar.

$a < 1$ ise karşıtlık azalır.



Şekil 2.12 : Sabit sayıya göre kontrast ayarı.

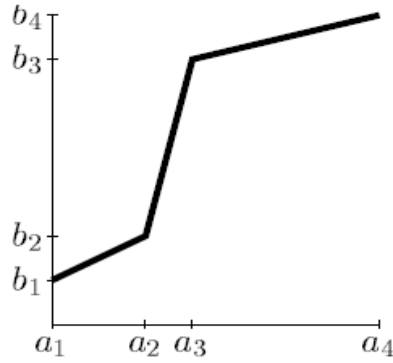
- **Kontrast germe;** kötü kontrastlı görüntülerin (histogramı dar bir alana yayılmış görüntüler), kontrastlarını iyileştirmek için yapılan genel bir yöntemdir. Amacı, histogramı geniş bir bölgeye yaymaktır (Şekil 2.13). Gri seviye değerlerini, parça parça lineer bir fonksiyon uygulayarak, orijinal aralığın dışına yayabiliriz. Şekil 2.14'te gösterilen **parça parça lineer germe fonksiyonu** a_i , b_i ve a_{i+1} ve b_{i+1} koordinatları arasında kalan piksel değerlerini bulmak için kullanılır [8].



(a)

(b)

Şekil 2.13 : (a) Kontrast germe işleminden önceki görüntü (b) Kontrast germe işleminden sonraki görüntü.



$$y = \frac{b_{i+1}-b_i}{a_{i+1}-a_i}(x - a_i) + b_i \quad (2.8)$$

Şekil 2.14 : Lineer germe fonksiyonu.

2.2.1.6 Eşikleme yöntemleri

Eşikleme, görüntü bölütleme amacı için kullanılan en önemli yaklaşımlardan birisidir. Görüntü bölütleme (segmentasyon), bir görüntüyü her biri içerisinde farklı özelliklerin tutulduğu anlamlı bölgelere ayırmaktır. Eşikleme işleminde amaç, görüntü içerisindeki nesneleri görüntü arka planından ayırmaktır [9].

Eşikleme bölütlemenin çok basit bir uygulama biçimidir. Gri seviyeli eşikleme teknikleri, dijital bir görüntüyü ortak, özel ve geniş bölümlere ayırmak için kolay hesaplanabilir metotlardır. Eşikleme işlemi, görüntünün birkaç anlamlı parçaya

ayrılmasına dayanan en iyi eşik değerlerinin bir kümesinin belirlenmesi işlemini içermektedir [10].

Eşikleme için, görüntüdeki gri seviye dağılımlarını gösteren görüntü histogramından faydalanılır. Örneğin, koyu bir arka plan üzerinde açık renkli nesnelerden oluşan bir görüntüye ilişkin histograma göre, nesnelere ve arka plana ait pikseller olmak üzere, görüntüyü iki ana grupta değerlendirmek mümkündür. Bu durumda nesneleri arka plandan ayırmak için en kolay yol, histogramdan göreceli olarak belirlenen bir T eşik değeri ile görüntüdeki piksel değerlerini karşılaştırmak olacaktır. Eğer piksel eşik değerinden büyükse nesne, aksi takdirde arka plan olarak belirlenir [11,12].

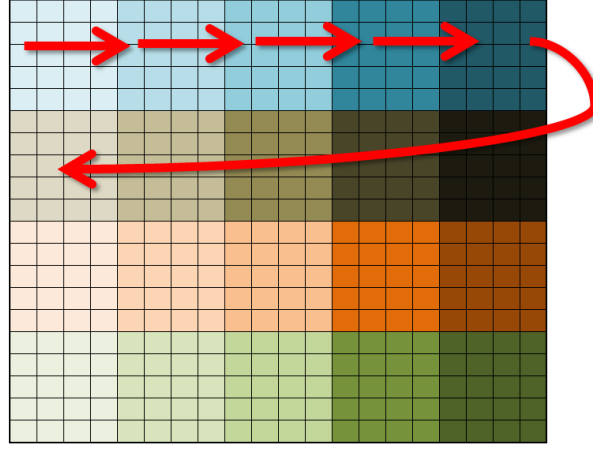
$$T = T [x, y, f(x,y), p(x,y)] \quad (2.9)$$

$$g(x,y) = \begin{cases} 1; & \text{eğer } f(x,y) > T \\ 0; & \text{eğer } f(x,y) < T \end{cases} \quad (2.10)$$

➤ Lokal eşik belirleme

Görüntünün değişik bölgelerinde farklı eşik değerler veya farklı eşik değer belirleme yöntemleri kullanılarak eşik değer belirleme işlemi yapılmasına, dinamik veya “lokal” eşik değer belirleme denilir. Eşik değer belirleme işlemindeki en önemli sorun eşik değer in amaca en uygun şekilde belirlenmesidir. Lokal eşik değer belirlemede tüm yöntemler, görüntü üzerinde bir piksel grubu (matris) gezdirilerek eşik değer belirlemenin yapılması prensibine dayanmaktadır. Eşik değer in belirlenmesindeki en yaygın yöntem, gezdirilen matris içerisindeki gri değerlerin ortalamasını almak ve belirlenen ortalamayı eşik değer olarak kullanmaktır. İşlem aynı şekilde tüm görüntüye uygulanır. Gürültü olmayan, tekdüze bir arka fona sahip bir görüntüde bu yöntem başarılı olabilir.

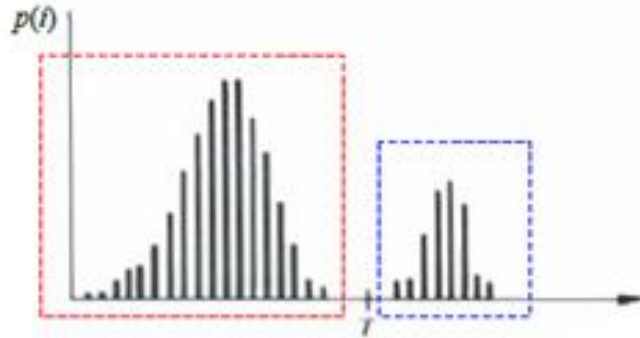
Eğer T hem $f(x,y)$ hem de $p(x,y)$ 'ye bağlı ise bu durumda (2.9) eşitliği olan T denklemindeki eşik değeri bölgesel (local) eşik olarak adlandırılır [13].



Şekil 2.15 : Lokal eşikleme sırası.

➤ Bütünsel (Global) eşikleme

Tüm eşikleme tekniklerinin en basiti, Şekil 2.16’da gösterildiği gibi tek bir eşik değeri kullanarak görüntü histogramını göstererek görüntüyü bölütleyen eşikleme yöntemidir.



Şekil 2.16 : Tek bir eşik değeri ile bölmelenen histogram.

Bu tip bölütleme, (2.10) eşitliğini dikkate alarak, görüntüdeki piksellerin nesne veya arka plan olarak etiketlenilmesi ile sonuçlanır. Bu bölütleme yönteminin başarısı, görüntü histogramının iyi bir biçimde bölmelenmesine bağlıdır. Bütünsel eşikleme bazı tür görüntülerde olumlu sonuç verse de bu yöntemlerin en belirgin eksikliği sadece tüm görüntüdeki piksellerin gri seviye değerlerini göz önüne alıyor olmalarıdır. Bütünsel eşikleme görüntünün her bir bölgesinde farklı bir eşik değeri kullanma gereksinimini, piksellerin yerleşimini ve aralarındaki yerel ilişkilerini hesaba katmazlar. Bundan dolayı sonraki aşamalarda birtakım hatalarla karşılaşmak mümkün hale gelmektedir [14].

- **Otsu eşik belirleme algoritması**

Toplam L gri seviyeden oluşan bir görüntünün her bir gri seviye değeri x ile temsil edilecek olursa bu gri seviye değerinin görüntü içerisindeki sayısının toplam piksel sayısına oranı p(x) ile ifade edilir ki p(x) fonksiyonu aynı zamanda görüntüye ilişkin normalize histograma karşı düşer.

Bu çerçevede, t eşik değeri olarak seçilecek olursa, arka plan olarak sınıflandırılacak olan piksel grubunun görüntüde bulunma olasılığı

$$\theta(t) = \sum_{x=1}^t p(x) \quad (2.11)$$

ile verilir. Diğer taraftan, nesneye ilişkin pikseller olarak sınıflandırılacak olan piksel grubunun görüntüde bulunma olasılığı

$$1 - \theta(t) = \sum_{x=t+1}^L p(x) \quad (2.12)$$

olacaktır. Arka plan ve nesneye ilişkin piksellerin gri seviyelerinin ortalama değerleri ise, sırasıyla aşağıdaki gibi hesaplanabilir:

$$\mu_b = \frac{\sum_{x=1}^t xp(x)}{\sum_{x=1}^t p(x)} = \frac{\mu(t)}{\theta(t)} \quad (2.13)$$

$$\mu_o = \frac{\sum_{x=t+1}^L xp(x)}{\sum_{x=t+1}^L p(x)} = \frac{\sum_{x=1}^L xp(x) - \sum_{x=1}^t xp(x)}{1 - \theta(t)} = \frac{\mu - \mu(t)}{1 - \theta(t)} \quad (2.14)$$

(2.14) eşitliğindeki μ , bütün bir görüntüdeki gri seviyelerin ortalamasını belirtir. (2.11) ve (2.12)'den yararlanarak, arka plan ve nesneye ait piksellerin gri seviyeleri ile ilgili varyanslar

$$\sigma_b^2 = \frac{\sum_{x=1}^t (x - \mu_b)^2 p(x)}{\sum_{x=1}^t p(x)} = \frac{1}{\theta(t)} \sum_{x=1}^t (x - \mu_b)^2 p(x) \quad (2.15)$$

$$\sigma_o^2 = \frac{\sum_{x=t+1}^L (x - \mu_o)^2 p(x)}{\sum_{x=t+1}^L p(x)} = \frac{1}{1 - \theta(t)} \sum_{x=t+1}^L (x - \mu_o)^2 p(x) \quad (2.16)$$

eşitlikleri ile hesaplanır. Görüntüdeki piksellerin dağılımının toplam varyansı ise

$$\sum_{x=1}^L (x - \mu)^2 p(x) \quad (2.17)$$

ile bulunur.

$$\sigma_T^2 = \underbrace{\theta(t)\sigma_b^2 + [1 - \theta(t)]\sigma_o^2}_{\sigma_W^2(t)} + \underbrace{(\mu_b - \mu)^2\theta(t) + (\mu_o - \mu)^2[1 - \theta(t)]}_{\sigma_B^2(t)} \quad (2.18)$$

şeklinde ifade edilir. Burada,

- $\sigma_w^2(t) \rightarrow$ Nesne ve arka plan ile ilgili her bir pikseller grubu içerisindeki varyans (Within-class variance)
- $\sigma_B^2(t) \rightarrow$ Nesne ve arka plan ile ilgili her bir pikseller grubu arasındaki varyans (Between-class variance) ifade etmektedir. (2.17)'den, nesne ve arka planı birbirinden en iyi şekilde ayırabilecek olan t eşik değeri, nesneleri temsil eden pikseller grubu ile arka planı temsil eden pikseller grubu arasındaki varyansı $\sigma_B^2(t)$ 'yi en büyük ve dolayısıyla bu pikseller grubu içerisindeki varyansı $\sigma_w^2(t)$ 'yi en küçük şekilde belirlenmelidir (Şekil 2.17) [15].



Şekil 2.17 : Otsu eşik değeri algoritmasıyla elde edilen görüntü.

2.2.2 Görüntü zenginleştirme yöntemleri

Filtreler görüntü zenginleştirme amacı ile görüntü içerisindeki bazı ayrıntıları ortaya çıkartmak veya görüntü içerisindeki istenmeyen gürültülerin yok edilmesini sağlamak için kullanılmaktadır. Özellikle; görüntüleri dijital olarak kaydeden cihazların görüntüyü hatalı bir şekilde elde etmeleri ve aydınlatma gibi çevre koşullarının yetersizliğinden kaynaklanan birçok kayıp ya da pürüz, görüntü işleme filtreleri kullanılarak en aza indirilebilmektedir. Farklı amaçlar için farklı filtreleme operatörleri vardır. Bunlara:

- Görüntü yumuşatma,
- Kenar keskinleştirme,
- Kenar yakalama gibi daha birçok amaçla kullanılan filtreler örnek verilebilir [6].

Filtreler, 3x3, 5x5 ya da 7x7 piksel boyutlu sayı matrisleri (kernel) olabilirler ve uygulanacak filtreleme yöntemine göre bu sayı matrisleri farklı değerlerden oluşmaktadır. Filtreleme işleminde bu filtre matrisleri görüntü üzerindeki bütün piksellerde dolaştırılarak uygulanır. Her piksel için filtre elemanları ve piksel

değerleri ile yeni gri ton değerleri hesaplanır. Matristeki elemanlar, belirli pikseller yönünde bu ortalamayı ağırlıklandırmak için kullanılmaktadır. Yüksek frekansları vurgulayan ve düşük frekansları bastıran filtrelere yüksek geçirgenli filtreler (high-pass filters) denir. Yüksek geçirgenli filtreler açık ve koyu tonlu detaylar arasındaki farkı daha da belirginleştirir. Alçak geçirgenli filtreler (low-pass filters) açık ve koyu detaylar arasındaki ton farkını azaltırlar. Alçak ve yüksek geçirgenli filtreleri uygulamanın en basit yolu uzaysal komşuluk ortalaması ile yapılır [16].

2.2.2.1 Görüntü yumuşatma

Görüntü yumuşatma operatörleri;

Görüntüdeki gürültünün (bozucu etkinin) yok edilmesi veya indirgenmesinde kullanılır. Eğer görüntüde mevcut gürültü yüksek frekansta ise lowpass-alçak geçirgenli filtre ile indirgenir.

Çözünürlüğün indirgenmesi işleminde kullanılır. Eğer görüntünün çözünürlüğü çok yüksek ise ön görüntüleme işlemleri için tüm çözünürlüğe gereksinim duyulmaz [17].

➤ Averaj (Mean) operatörü

$$\frac{1}{9} * \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

Şekil 2.18 : Averaj filtresi.

Ortalama filtre olarak adlandırılır. Bu filtre alçak geçirgen filtrelerdendir (Şekil 2.18). Bu filtre görüntü üzerinde piksellerin gri ton değerlerinin değişmediği bir bölgeye yerleştirildiğinde merkezdeki gri ton değeri değişmeden kalır. Aynı filtre, gri ton değerlerinde büyük değişiklikler olan bir bölgeye yerleştirildiğinde, merkez pikseline çevresindeki piksellerin gri ton değerlerinin ortalaması atanacağından merkez ile çevre pikseller arasındaki fark azalacaktır. Bu yüzden bu tür filtreler görüntüdeki gürültüleri ortadan kaldırmak için kullanılırlar ancak yan etki olarak görüntüyü bulanıklaştırırlar [20].



(a)



(b)

Şekil 2.19 : (a) Orijinal görüntü (b) 5x5 Averaaj filtresi geçirilmiş hali.

➤ **Katlama Operatörü**

Katlama filtreleme bir görüntünün mekansal frekans karakteristiğinin değiştirilmesinde kullanılır. Filtre çekirdeği kullanılarak filtreleme işlemi gerçekleştirilir [6].

$$g' = \frac{\sum_{i=1}^n (\sum_{j=1}^n f_{ij} g_{ij})}{F} \quad (2.19)$$

Burada

f_{ij} ; katlama çekirdeğinin(filtre matrisinin) katsayısıdır,

g_{ij} ; piksel gri değeridir,

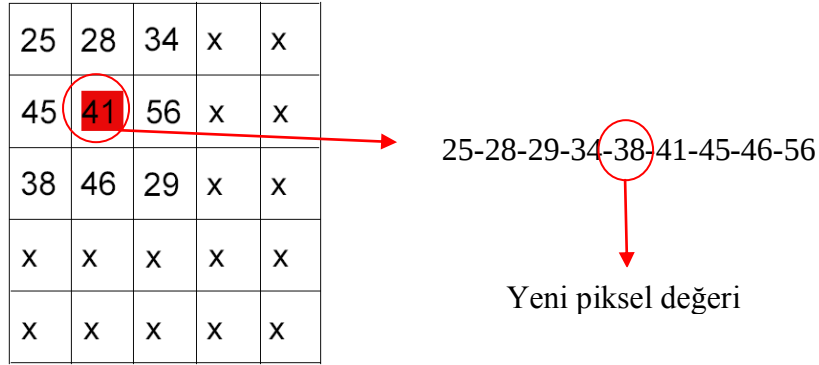
n ; filtreleme matrisinin boyutlarıdır (örn 3x3,5x5,7x7),

F ; filtreleme matrisinin elemanları toplamıdır. Eğer toplam sıfırsa 1 dir.

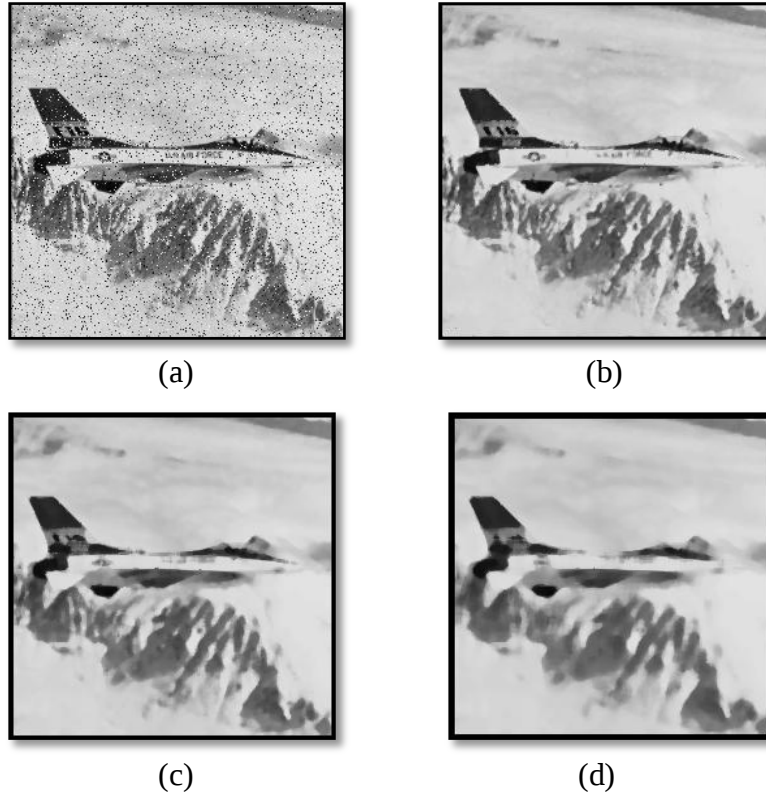
➤ **Medyan Filtresi**

Medyan filtresi doğrusal olmayan bir filtrelemedir. Tuz, biber gürültülerinin giderilmesinde etkilidir fakat görüntünün kenar bölgelerini bozmaktadır. Şekil 2.20’de görüldüğü üzere medyan filtresi, bir pikselin komşuluğundaki piksellerin gri ton değerlerinin küçükten büyüğe sıralanmasıyla oluşan listenin ortasındaki elemanı, yani medyanı çıktı değeri olarak alınır. 3x3 büyüklüğünde bir komşuluk için medyan değeri listenin beşinci elemanıdır. Sıralama işlemini temel alan filtrelerde bir pikselin yeni gri ton değerini belirlemek için bu pikselin bir komşuluğu kullanılır ancak bu bölge üzerine bir maske yerleştirilmez. Bunun yerine, önce bu komşuluktaki

piksellerin gri ton değeri küçükten büyüğe sıralanır ve sonra, bu listenin içinden bir değeri seçilerek yeni gri ton değeri belirlenir [3].



Şekil 2.20 : Medyan filtresinin işlem adımı.



Şekil 2.21 : (a) Orijinal görüntü (b) 3x3 Median filtre sonucu (c) 5x5 Median filtre sonucu (d) 7x7 Median filtre sonucu.

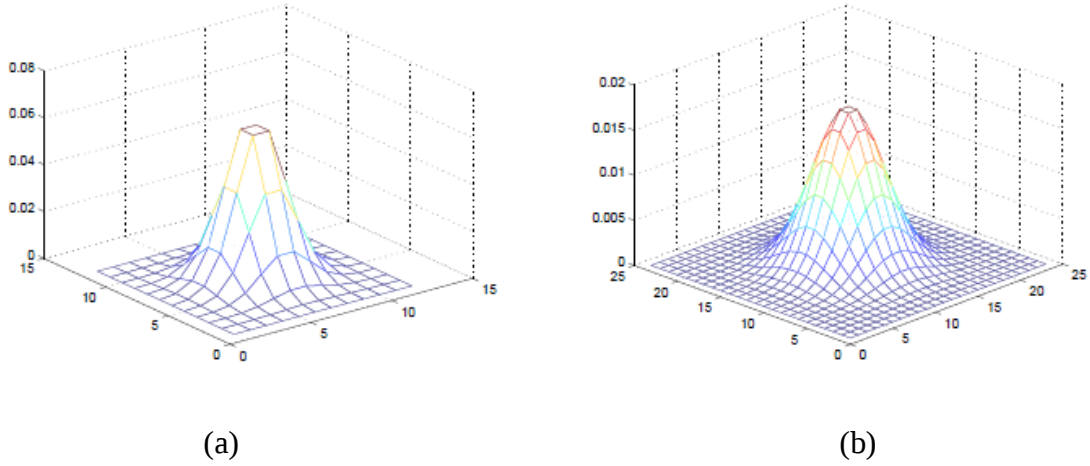
➤ Gauss filtresi

Ortalama filtrenin Gauss dağılımını kullanarak biraz daha değiştirilmiş hali Gauss filtre olarak bilinir. Fourier dönüşümüdür. Gauss filtre ile sonsuz bir transfer fonksiyonuna karşılık mekansal alanda sonlu bir pencerede (tarama penceresi) filtreleme yapılabilir. Bu da filtrelemenin temel problemini daha kolay

özöllebilir hale getirir. (2.20) eřitlięinde Gauss filtresinin matematiksel ifadesi yer almaktadır [18,19].

$$e^{\frac{-(x^2+y^2)}{2\sigma^2}} = e^{\frac{-x^2}{2\sigma^2}} * e^{\frac{-y^2}{2\sigma^2}} \quad (2.20)$$

Gauss Yumuřatmasının Avantajları: Filtreleme nce yatay ardından ıkan sonula dřey ekseninde gerekleřtirilebilir. řekil 2.22’de Gauss filtre boyutu ile sigma deęerlerinin  boyutlu grafiksel řekli gsterilmiřtir.



řekil 2.22 : (a) 12x12 Gauss $\sigma = 1.5$ b) 24x24 Gauss $\sigma = 3$.

2.2.2.2 Grnt keskinleřtirme

Keskinleřtirme iřlemi grntdeki kk detayların (yksek frekanslı bileřenler), křelerin daha belirgin hale getirilebilmesi iřleminde kullanılır.

► Laplace Operatr

Farklı ynlerdeki kenarlara aynı tepkiyi veren ikinci trev operatr olan Laplace grnt iřlemede sınır belirleme ve netleřtirme amalı olarak kullanılabilmektedir (řekil 2.23). $f(x,y)$ fonksiyonu řeklindeki bir grntye Laplace uygulandıęında;

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (2.21)$$

ifadesi, x ve y ynleri iin ayrı ayrı hesaplandıęında, ařaęıdaki eřitlikler elde edilir.

$$\frac{\partial^2 f}{\partial x^2} = f(x+1,y) + f(x-1,y) - 2f(x,y) \quad (2.22)$$

$$\frac{\partial^2 f}{\partial y^2} = f(x,y+1) + f(x,y-1) - 2f(x,y) \quad (2.23)$$

İlk eřitlikte gsterilen ifadenin aık halini tekrar ele etmek zere yukarıdaki eřitlikler toplanarak elde edilen eřitlik

$$\nabla^2 f = [f(x+1,y) + f(x-1,y) + f(x,y+1) + f(x,y-1)] - 4f(x,y) \quad (2.24)$$

Laplace filtresinin matrisleri;

$$L = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad L = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \text{ kullanılır.}$$



Şekil 2.23 : (a)Orijinal görüntü (b)Laplace maskesi geçirilmiş görüntü c) Maske ile görüntünün toplanmış hali.

2.2.2.4 Kenar Belirleme Yöntemleri

Kenar belirleme, görüntü işlemede temel öneme sahip konulardan birisidir. Bir görüntüdeki kenar, aydınlatma veya yüzey yansımaları gibi bir görüntünün fiziksel görünüşünde oluşan önemli bir değişime karşı düşer. Bir görüntünün içeriğini oluşturan nesneler, bu görüntüde arka plandan farklı gri ton değerleriyle belirlenirler. Bu farklılık arka planın sabit bir gri ton değerini taşıdığı durumlarda, bu gri ton değerinde bir süreksizlik şeklinde belli olur. Bu anlamda, bir görüntünün gri seviyelerinde ani değişikliklerin olduğu bölgelere kenar adı verilir. Görüntüde bu süreksizlikleri belirlemede en çok kullanılan yöntem, görüntü üzerinden bir maske geçirmektir. Süreksizlikleri ortaya çıkaran maskeler, çizgiler veya kenarlar çevrelerindeki piksellerden farklı gri ton değerlerine sahip olduklarından, yüksek geçiren filtre etkisi gösterirler. Kenar belirleme yöntemlerinin temel mantığı bölgesel türeve dayanır [20,21]. Kenar belirleme maskeleri şekil 2.24’te gösterilmiştir:

-1	-1	-1
2	2	2
-1	-1	-1
Yatay		

-1	-1	2
-1	2	-1
2	-1	-1
+45°		

-1	2	-1
-1	2	-1
-1	2	-1
Dikey		

2	-1	-1
-1	2	-1
-1	-1	2
-45°		

Şekil 2.24 : Kenar belirleme maskeleri.

➤ I. Türeve dayalı kenar belirleme yöntemleri

Görüntü gradienti; bir vektör olarak tanımlanır ve vektörün büyüklüğü vektörün doğrultusu boyunca her bir birimdeki değişikliğin büyüklüğünü ifade eder ve ayrıca yöne bağımlıdır. Bir $f(x,y)$ fonksiyonu için gradient bağıntısı;

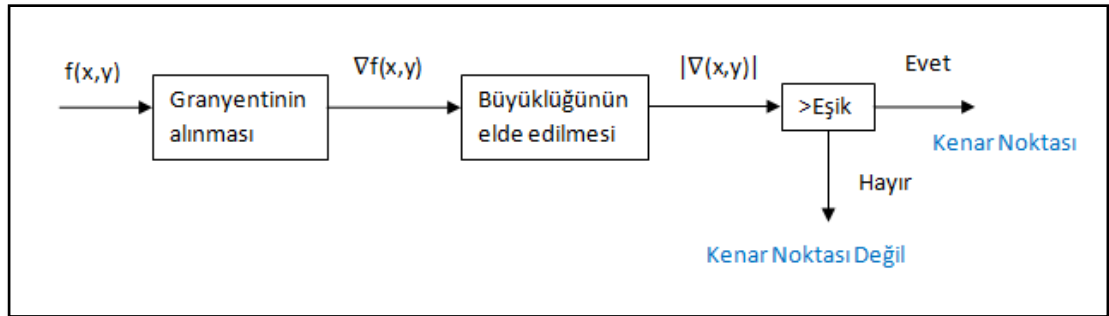
$$\text{Gradient ; } \nabla f = \text{grad}(f) = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (2.25)$$

$$\text{Gradient'in genliği ; } M(x,y) = \text{mag}(\Delta f) = \sqrt{g_x^2 + g_y^2} \quad (2.26)$$

$$\text{Gradient'in yönü ; } \alpha(x,y) = \tan^{-1} \left[\frac{g_x}{g_y} \right] \quad (2.27)$$

$$\text{Kenarın yönü ; } \theta = \alpha - 90^\circ \quad (2.28)$$

$f'(x,y)$ aynı zamanda kenarın yönünü ve büyüklüğünü kestirmede kullanılabilir. Eğer $|f'(x,y)|$ çok büyük ise, $f(x,y)$ çok hızlı değişir ve bu durum parlaklıkta hızlı bir değişime karşı düşer. Eğer $f'(x,y)$ pozitif ise, $f(x)$ artan bir fonksiyondur. Buradan hareketle birçok kenar belirleme algoritmasının temeli, Şekil 2.25'te görüldüğü üzere gradientin genliğinin bulunması ve onu bir eşik değeriyle karşılaştırmaya dayanır.



Şekil 2.25 : Gradient metodu blok diagramı.

- **Prewitt kenar belirleme operatörü;**

Prewitt kenar yakalama yöntemi birinci derece türevleri olan G_x ve G_y i kullanarak Şekil 2.26'da gösterilen maskelerle görüntüyü filtreler. Prewitt filtresi görüntü üzerinde filtre gezdirilmesi bakımından daha az zaman harcar. Sobel'e göre daha gürültülü sonuçlar verme eğilimindedir [21]. Şekil 2.27'de Prewitt kenar filtresinin G_x ve G_y maskeleri geçirildikten sonraki görüntüler gösterilmiştir.

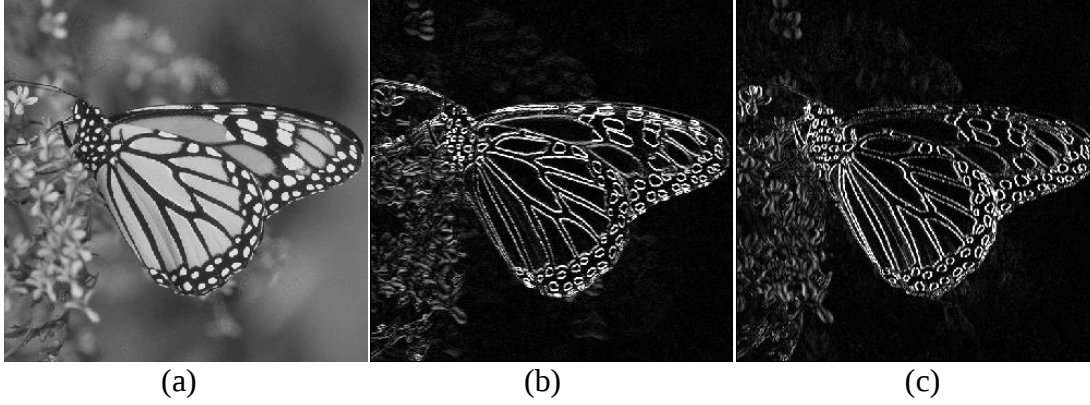
-1	-1	-1
0	0	0
1	1	1

G_x Maskesi

-1	0	1
-1	0	1
-1	0	1

G_y Maskesi

Şekil 2.26 : Prewitt operatörü maskeleri.



Şekil 2.27 : (a) Orijinal görüntü (b) Prewitt G_x (c) Prewitt G_y .

- **Sobel kenar belirleme operatörü**

Gürültü etkisini gidermek için kullanılan operatördür. Yatay ve düşey yönde keskinlikleri yakalar. 3x3 pencere alanına uygulanır [21].



Şekil 2.28 : (a) Orijinal görüntü (b) Sobel kenar belirleme yöntemi ile oluşan görüntü.

- **Canny kenar belirleme operatörü**

Canny operatörü çok adımlı bir işlemdir. İlk önce görüntü bir Gauss filtre ile yumuşatılır (smooth). Daha sonra yumuşatılmış görüntüye bölgelerini belirlemek üzere iki boyutlu basit bir türev operatörü (örneğin Robert Cross) uygulanır. Böylece

eğim yoğunluklu görüntüde kenarlar belirir (Şekil 2.29). Daha sonra bu tepe noktalar takip edilerek tepe olmayan bütün noktalar 0 yapılır. Böylece çıkışta ince bir çizgi verilmiş olur. Bu işleme non-maximal suppression adı verilir [22].



Şekil 2.29 : (a) Orijinal görüntü (b) Canny kenar belirleme yöntemi ile oluşan görüntü.

II. Türeve dayalı kenar belirleme yöntemi

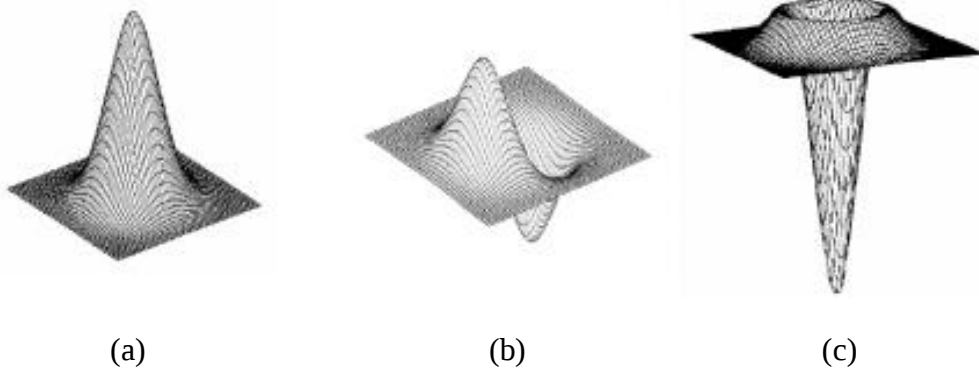
Herhangi bir kenar ilişkin ani geçiş noktasının belirlenmesi için başvurulacak yöntemlerden diğeri, fonksiyona ilişkin ikinci türevi kullanır (Şekil 2.30). Kenara ilişkin birinci türevin en büyük veya en küçük olduğu noktada işaretin ikinci türevi sıfıra eşittir. Buna dayanarak, bir görüntü fonksiyonuna ilişkin ikinci türev alınıp sıfır geçiş noktalarının tespit edilmesiyle görüntüye ilişkin kenar görüntüsüne ulaşılır.

• Laplacian of Gauss (LOG) kenar belirleme

Laplace operatörünün ikinci türevine göre çalışır ve yönden bağımsızdır. Laplace işlemine dayalı yöntemler gürültüler için hassas olduğu için kenar belirleme işlemi gerçekleşmeden önce filtre ile temizlenmelidir. Bu amaçla Gauss filtresi kullanılırsa LoG (Laplacian of Gauss) kenar belirleme operatörü elde edilir. Bu operatör kullanılarak,

Gauss filtresi $g(x,y)$ ile görüntü $f(x,y)$ katlanarak yumuşatılır. Böylece gürültülü noktalar ve küçük yapıcıklar izole edilir. Ancak yumuşatma ile birlikte kenarlarda yayılma gerçekleşir.

Görüntüyü iyileştirmek (enhancement) için gauss filtresi, ikinci türev elde etme bölümünde de kullanılır. Böylece gürültülü noktalar ve küçük yapıcıklar izole edilir [23].

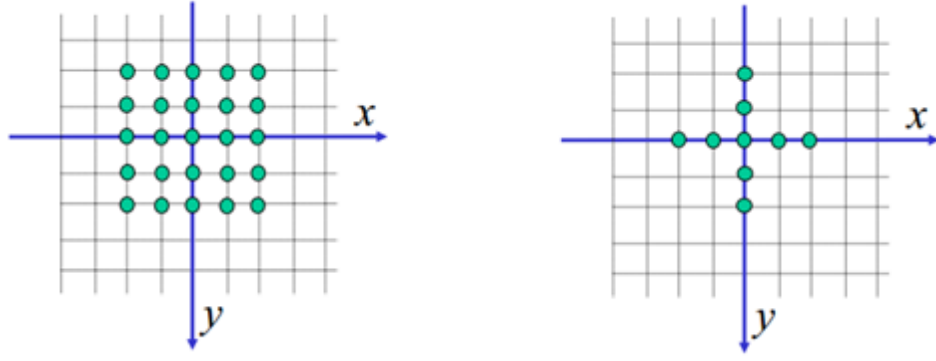


Şekil 2.30 : (a) Gauss fonksiyonun 3-B görünüşü (b) Gauss fonksiyonun birinci türevi (c) Gauss fonksiyonun laplace'ı.

2.2.3 Morfolojik İşlemler

Matematiksel morfoloji, küme teorisi, topoloji ve rastgele fonksiyonlara dayalı bir analiz ve işleme yöntemidir. Geometrik yapılar ile uğraşmaktadır. Devamlılık ve boşluk gibi topolojik ve geometrik kavramlar, şekil, ayrıklık, birleşiklik, dışbükeylik, içbükeylik ve uzaklık gibi özelliklerle birlikte matematiksel morfolojiyi karakterize eder. Matematiksel morfoloji görüntü içerisinde ayırt etmek istediğimiz nesneleri, görüntü içerisindeki diğer bölgelerden veya nesnelerden ayırt etmek için kullanılmaktadır [24].

Yapıtaşı elemanı (Structuring Element), görüntü işlemede kullanılan ve ana görüntü üzerinde dolaştırılan matrisin morfoloji için özelleştirilmiş halidir (Şekil 2.31). Yapıtaşı elemanı görüntü üzerinde dolaştırılır ve uygulanan morfolojik işlemin tanımına göre, görüntüdeki piksel değerleri ile yapıtaşı elemanının değerleri karşılaştırılarak, sonuç görüntüsünde kullanılacak değer elde edilir. Farklı şekillerde ve büyüklüklerde olabilmekle birlikte bu yapıtaşların bir merkez noktası bulunmakta olup, işlenecek görüntünün her bir pikseli bu noktaya oturtularak işlem yapılmaktadır [25].

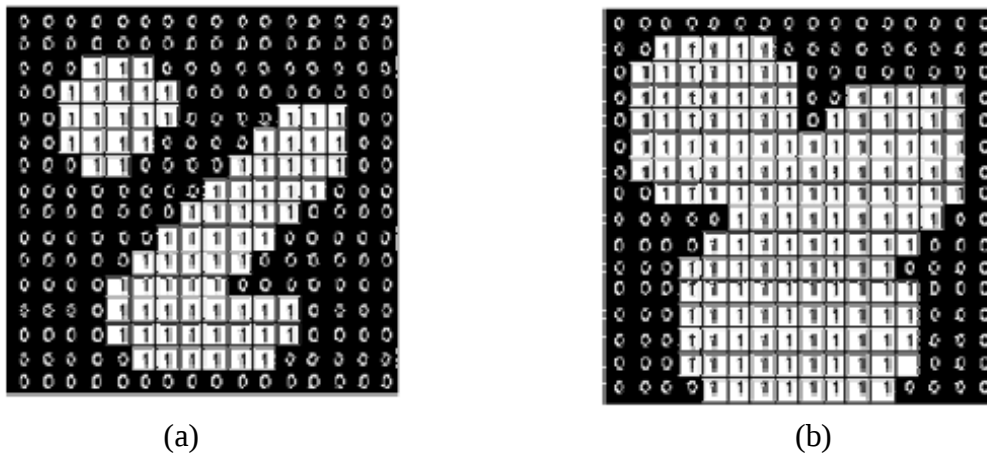


Şekil 2.31 : Morfolojik yapıtaşı elemanı örnekleri.

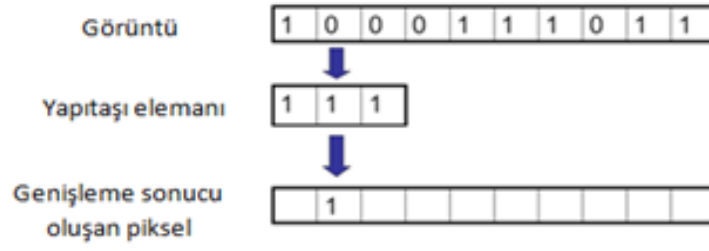
2.2.3.1 Genişleme ve aşınma morfoloji operatörleri

Genişleme işleminde, yapıtaşı elemanı görüntü üzerinde dolaşırken, yapıtaşı elemanın merkezi ile obje çakıştığı anda yapıtaşı elemanı kadar genişleme gerçekleşir (Şekil 2.32). Böylece her bir piksel yapıtaşı elemanı kadar büyür. Sonuç olarak arka plan küçülmüş ve obje büyümüştür. Eğer yapıtaşı elemanın orijini görüntü üzerinde "0" değerli bir piksel ile karşılaşırsa herhangi bir değişiklik meydana gelmez. Eğer değeri "1" olan bir piksel ile karşılaşırsa yapıtaşı elemanla yapıtaşı elemanın altında kalan pikseller "mantıksal or" işlemine tabi tutulurlar. Yani herhangi "1" değeriyle sonuç "1" e çevrilir (Şekil 2.33) [26]. A görüntüsünün B yapıtaşı elemanı ile gerçekleştirilen yayma işlemi (δ) (2.29) eşitliği ile ifade edilir:

$$\delta(A) = \bigcap_{b \in B} A + b = A \oplus B \quad (2.29)$$



Şekil 2.32 : (a) Binari görüntü (b) Genişleme operatörü işlemi sonucunda oluşan görüntü.

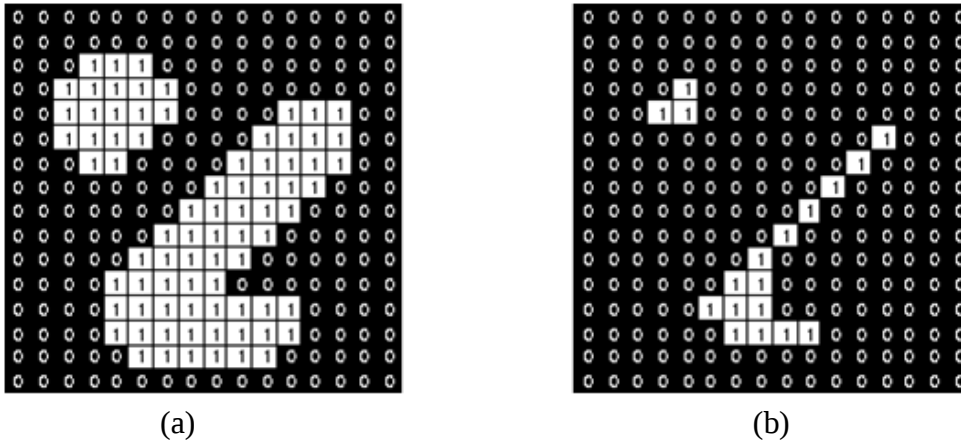


Şekil 2.33 : Genişleme işlemi temel işleyişi.

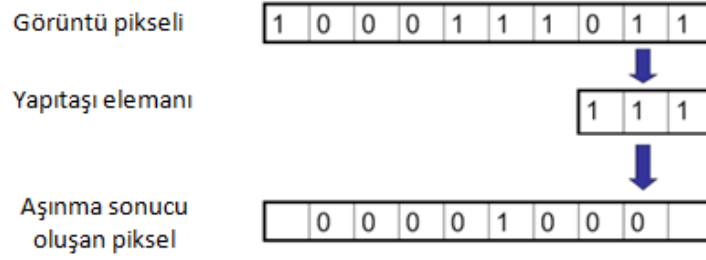
Aşınma işleminde, yapıtaşı elemanının merkez noktası dışındaki yerler arka plan halini alarak nesne de bir aşınma meydana gelir (Şekil 2.34). Aşındırma, ikili bir görüntüdeki nesneyi küçültmeye ya da inceltmeye yarayan morfolojik işlemidir. Burada yine aynı şekilde yapıtaşı elemanı görüntü üzerinde piksel-piksel dolaştırılır fakat bu defa yapıtaşı elemanın merkez pikseli "1" değeri ile karşılaşırsa yapısal eleman içerisindeki piksellerin durumuna bakılmakla birlikte eğer yapıtaşı elemanı içerisindeki "1" olan piksellerden herhangi biri altında görüntüye ait "0" değeri varsa yapıtaşı elemanın diğer "1"lerinin altındakilerle beraber bu piksel "0" a dönüştürülür (Şekil 2.35) [26].

A görüntüsünün B yapısal elemanı ile gerçekleştirilen aşındırma (ε) işlemi (2.30) eşitliği ile ifade edilir:

$$\varepsilon(A) = \cap_{b \in B} A - b = A \theta B \quad (2.30)$$



Şekil 2.34 : (a) Binari görüntü (b) Aşınma operatörü sonucunda oluşan görüntü.



Şekil 2.35 : Aşınma işlemi temel işleyişi.

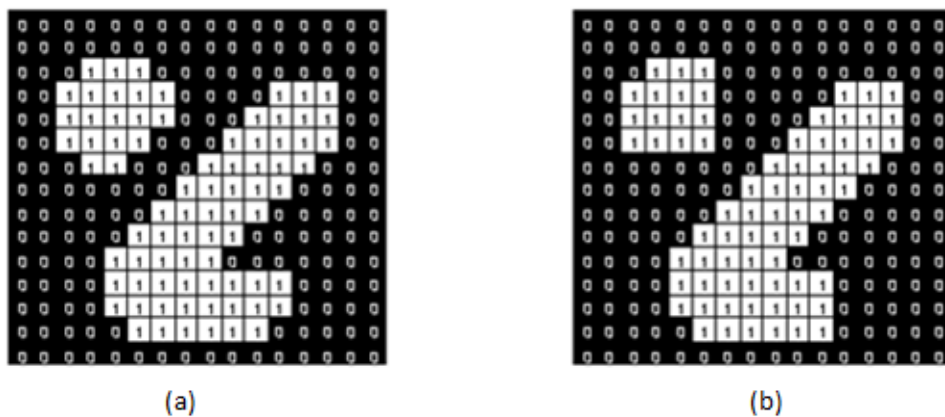
2.2.3.2 Açma ve kapama morfoloji operatörleri

Açılış operatörü, aşınma ve genişleme operatörlerinin ardından kullanılması ile oluşturulmuş bir matematiksel morfoloji operatörüdür. Operatördeki prensip, yapıtaşı elemanından daha az piksele sahip grupların elenmesi diğer yerlerin ise aynı şekilde kalmasıdır.

Açılım işlemi sonucunda nesnelerdeki ince çizgiler ortadan kalkar, ufak benekler yok olur, nesnelerin keskin ayrıntıları yumuşar (Şekil 2.36). Bu işlem sonucunda yapıtaşı elemanından daha ince bağlarla birbirine bağlanmış bu nesneler, birden çok bölgeye ayrılabilir. Açılım işlemi geometrik olarak, bir yapılandırma elemanını bir bölgenin içinde gezdirmeye ve bu elemanın bölge içinde ulaştığı yerlerin sınırlarını çizme işlemine benzetilebilir [27].

Bir A kümesinin B yapıtaşı elemanı açılımı (2.31) eşitliği ile tanımlanır.

$$A \circ B = (A \ominus B) \oplus B \quad (2.31)$$

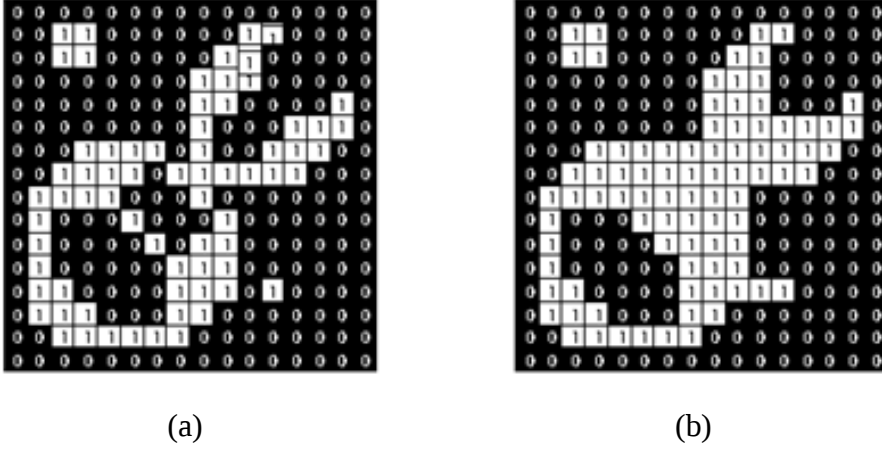


Şekil 2.36 : (a) Binarı görüntü (b) Açma operatörü sonucunda oluşan görüntü.

Kapanış operatörü, açılış operatörünün tam tersidir. Genişleme ve aşınma operatörlerinin ardından kullanılır. Kapanım operatöründeki asıl amacı boşlukları kapatmaktır (Şekil 2.37). Ancak bu matematiksel morfoloji operatörünün başarılı olduğu nokta, yalnızca obje dâhilinde bulunan boşlukları giderebilmesidir.

Kapanım işlemi sonucunda nesnelerdeki ufak girintiler ve ince uzun 'körfezler' ortadan kalkar, küçük delikler yok olur. Kapanım işlemi geometrik olarak, bir yapıtaşı elemanını bir bölgenin dışında gezdirmeye benzetilebilir. Bu durumda bölgenin dışına doğru taşan detaylar özelliklerini biraz kaybetmelerine rağmen göreceli olarak sağlam kalırlar ancak, bölgenin sınırlarındaki girintiler yok olur [27]. Bir A kümesinin B yapıtaşı elemanı kapanımı (2.32) eşitliği ile tanımlanır.

$$A \bullet B = (A \oplus B) \ominus B \quad (2.32)$$



Şekil 2.37 : (a) Binari görüntü (b) Kapama operatörü sonucunda oluşan görüntü.

2.2.3.3 Diğer Morfoloji Operatörleri

Temel matematiksel operatörler olan genişleme, aşınma, açma, kapama operatörlerinin dışında birçok farklı uygulamalarda kullanılan morfolojik operatörler vardır [28]. Bu operatörler;

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

3. MATLAB YAZILIMINDA GÖRÜNTÜ İŞLEME

MATLAB Mathworks firması tarafından geliştirilen bir uygulama geliştirme ortamıdır. MATLAB MATrix LABoratory kelimelerinin kısaltılmasıdır. MATLAB, dijital ve sembolik hesaplamalar, veri çözümlemesi, gerçek ortamda test ve ölçüm yapabilme, çok gelişmiş çizim işlemleri, algoritma geliştirme, ileri seviye programlama, C/C++ ile beraber çalışabilmesi, dijital görüntü işleme, mühendislik ve bilimsel uygulamaları ile tüm dünyada birçok alanda yaygın olarak kullanılan yazılımdır [29].

3.1 MATLAB'ın Avantajları

MATLAB (matrix laboratory) dijital hesaplama ve dördüncü nesil programlama dilidir. MATLAB, matris işlenmesine, fonksiyonlar ve veri çizilmesine, algoritmalar uygulanmasına, arayüz oluşturulmasına ve diğer dillerle yazılmış programlar ile etkileşim oluşturmaya izin verir. MATLAB'ın diğer uygulama geliştirme platformlarına göre olan avantajı zengin matematiksel işlem yeteneğidir. Başka programlama dilleri ile satırlarca kod yazılarak gerçekleştirilen işlemler MATLAB'ın hazır matematiksel algoritmaları ile tek bir fonksiyon ile gerçekleştirilebilir. MATLAB birçok uygulama alanına göre hazırlanmış fonksiyonlara, algoritmalara sahiptir. MATLAB görüntü işleme gereçlerinin yanı sıra kontrol sistemleri, haberleşme, yapay sinir ağları, istatistik gibi birçok alanda uygulama geliştirebilecek imkân sağlamaktadır.

MATLAB ile kod yazılarak işlemler gerçekleştirilebildiği gibi simülasyonlar hazırlanarak birçok uygulama test edilebilir. Ayrıca simülasyonların sistemlerle gerekli donanım aracılığı ile bağlantısı sağlanarak gerçek zamanlı uygulamalar gerçekleştirilebilir.

MATLAB işlem yapılan değişkenleri matris olarak tutmaktadır. Bu matrislerin sahip olduğu değerleri istenilen anda görüntüleyip, dış ortamlara aktarma, dış ortamdan veri alma imkânları sağlamaktadır. MATLAB'in önemli özelliklerinden biride üstün

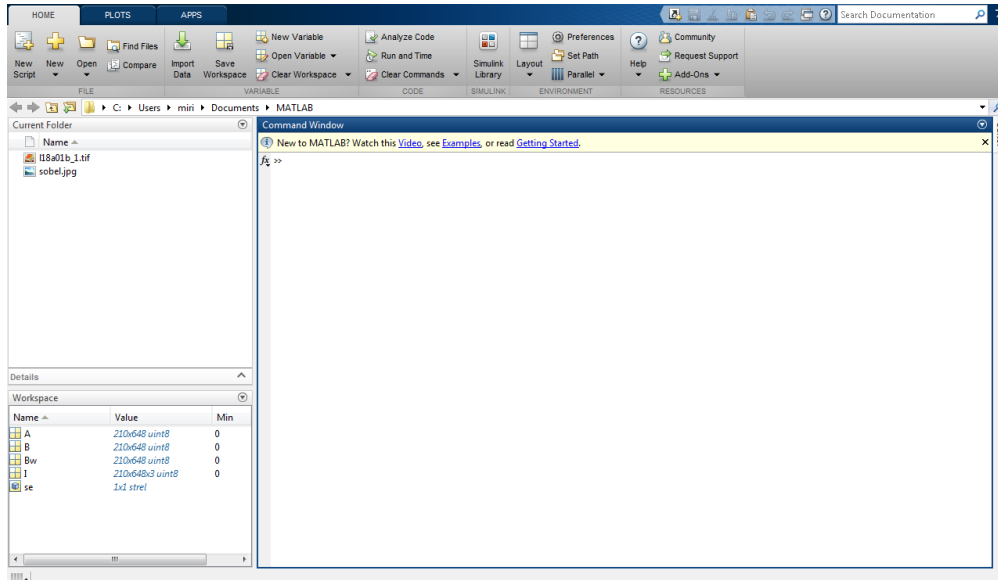
grafik yeteneğidir. İşlem sonuçları 2 ve 3 boyutlu grafik türleri ile görüntülenebilir [30].

3.2 MATLAB'in Ana Ekranı

MATLAB ana ekranında bulunan kısımların en önemlisi “Command Window” denilen komut ekranıdır. Bu ekrana tek satırlı fonksiyonlar ve komutlar yazılarak işlemler yapılabilir. Ayrıca döngü ve karar yapıları ile çok sayıda fonksiyon ve komut bir arada kullanılabilir.

“Workspace” alanında ise o anda işlem yapılan, yüklenmiş olan tüm değişkenler ve matrisler yer almaktadır. Bunlardan istenilenler “Array Editor” denilen arabirim ile görüntülenir.

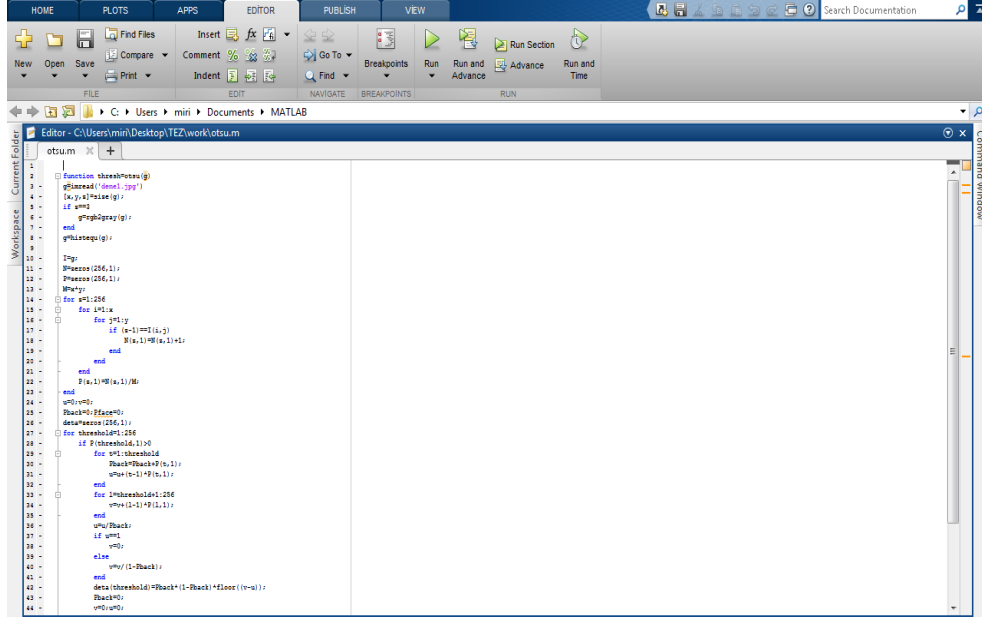
“Command History” denilen bölüm ise MATLAB'in son kullanımlarından gerçekleştirilen işlemler ve komutlar tarihleri ile birlikte tutulmaktadır. MATLAB'in menüleri kullanılarak dosya, düzenleme, grafik, program işlemleri, pencere ve yardım işlemleri yapılabilir. Ayrıca sol altta yer alan “Start” düğmesine basılarak MATLAB araç kutularına, simülasyon ve görsel arayüz tasarlama araçlarına erişilebilir [31]. Şekil 3.1'de MATLAB ana ekranı gösterilmektedir.



Şekil 3.1 : MATLAB ana ekranı.

3.3 MATLAB Kod Yazma Ekranı

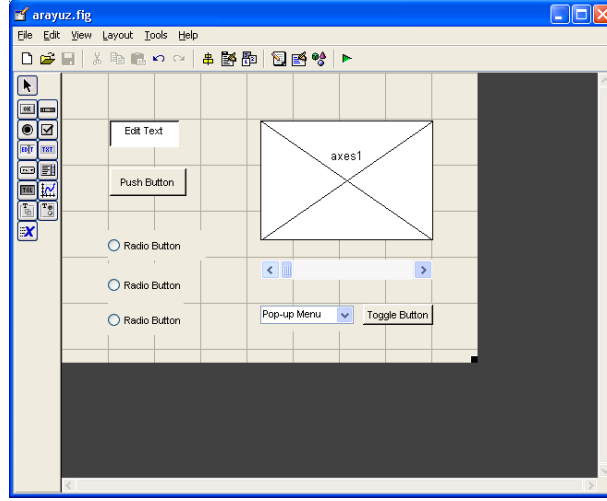
MATLAB ortamında kod ile uygulama gerçekleştirme .m dosyası denilen metin dosyaları yazılarak gerçekleştirilir. Bu uygulamalar “M File Editor” denilen araç kullanılarak yazılır. M file editör ile MATLAB’ in standart fonksiyonları ve ilgili gerecin fonksiyonları kullanılarak programlar yazılabilir. Editör gelişmiş birçok programlama editörü gibi kod düzenleme, derleme, test etme araçlarına sahiptir.



Şekil 3.2 : MATLAB editör penceresi.

3.4 MATLAB Arayüz Geliştirme Ortamı

MATLAB klasik programlama tekniği olan kod yazarak uygulama geliştirmenin yanı sıra görsel ara yüz geliştirme aracına sahiptir. Guide denilen araç (Graphical User Interface Design) MATLAB ortamında görsel tasarım yapılabilir (Şekil 3.3). Düğmeler, metin kutuları, radyo düğmeleri, onay kutuları, kaydırma çubukları kullanılarak görsel tasarım oluşturulur. Bu elemanların arka planına yazılan .m dosyaları ile uygulama tamamlanır [31].



Şekil 3.3 : MATLAB arayüz geliştirme ortamı.

4. UYGULAMA

4.1 Uygulanan Yöntemler

Bu çalışmada kullanılan MATLAB programı ve önceki bölümlerde anlatıldığı üzere dijital görüntü işleme tekniklerinin ön işleme aşamaları, eşikleme yöntemleri, kenar belirleme teknikleri, morfolojik işlemlerin nasıl kullanılabileceği üzerinde durulmuştur. Bu bölümde çeşitli görüntüler kullanılarak bu yöntemlerin kullanımı irdelenmiştir.

MATLAB; JPEG, BMP, HDF4, PCX, TIFF, PNG, GIF, XWD, ICO gibi uzantılı görüntüleri desteklemektedir. Bu çalışmada, kullanılan dosya formatı JPEG uzantılıdır.

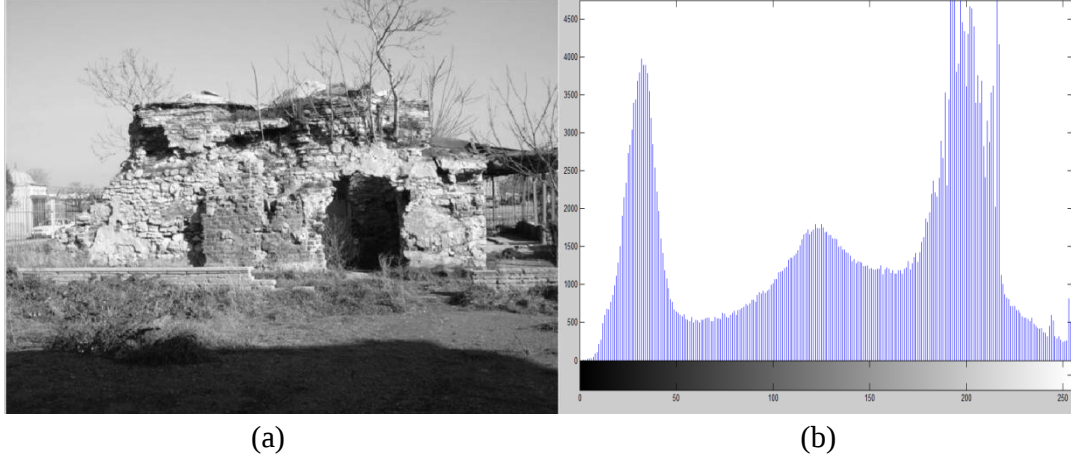
4.1.1 Ön işleme

İlk aşamada MATLAB Image Processing Toolbox'ta görüntüleri okutmak için 'imread' komutunu kullanılır. $I = \text{imread}(\text{'tez.jpg'})$; olarak girildiğinde MATLAB workspace penceresinde görüntünün (533 x 710 x 3 unit8) olduğunu yani 533 x 710 görüntü pikselini x 3 görüntünün renkli olduğunu, unit8 ise görüntünün 0-255 arasında değişen 8 bitlik bir görüntü olduğunu ifade eder (Şekil 4.1). Görüntüde işlem yapmak için önce görüntüyü RGB (red gren blue)'den gri seviyeye çevrilmiştir.

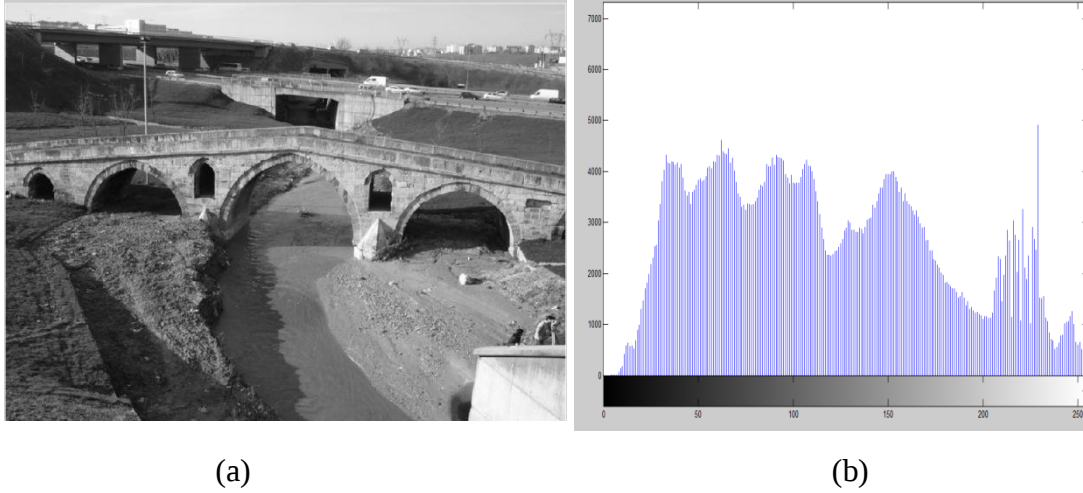


Şekil 4.1 : Çalışmada kullanılan örnek renkli görüntüler.

Daha sonra görüntünün histogramını görmek için ‘imhist’ komutu çalıştırılmıştır. Görüntünün histogramı Şekil 4.2 ve 4.3’de görüldüğü üzere oluşturularak görüntü hakkında bilgi edinmek amaçlanmıştır. Histogramda piksel yoğunluk değerleri ve o yoğunluğa ait piksel sayısı verilmesinin sayesinde görüntünün her bir noktasındaki piksellerin tespiti ile piksellerin sayısının ne olduğu hakkında bilgi edinilmiştir.

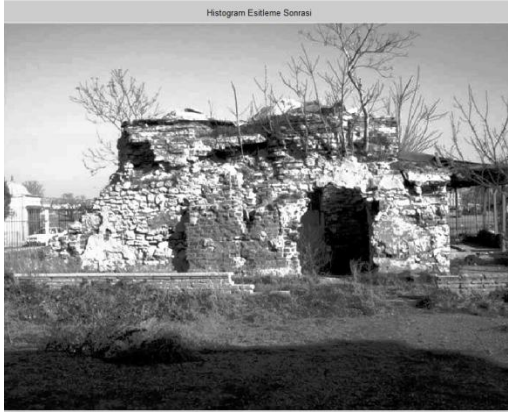


Şekil 4.2 : (a) Gri seviyeli görüntü (b) Gri seviyeli görüntünün histogramı.

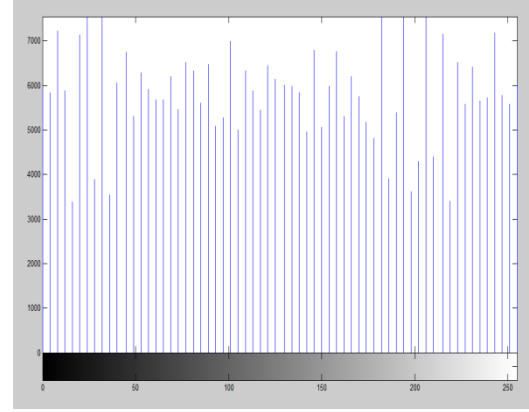


Şekil 4.3 : (a) Gri seviyeli görüntü (b) Gri seviyeli görüntünün histogramı.

Görüntüde Şekil 4.4 ve 4.5’te histeq komutu ile histogram eşitleme yapılarak görüntünün görünürliğini arttırılmaya çalışılmıştır. Histogram eşitlemedeki amacımız renk değerleri düzgün olmayan görüntülerin renk dağılımlarını düzenlemektir. Gri seviye dağılım aralıkları genişlediğinden ve yüksek seviyede toplandığından görüntüdeki parlaklık artmıştır.

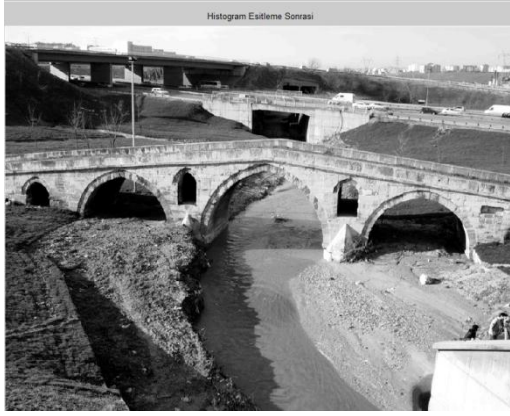


(a)

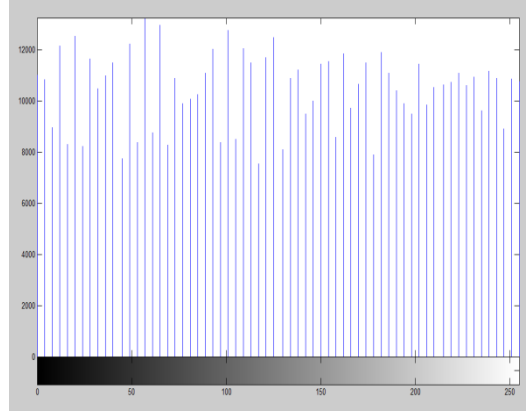


(b)

Şekil 4.4 : (a) Global histogram eşitlenmesi sonrası görüntü (b) Histogram eşitlenmesi sonrası görüntünün histogramı.



(a)



(b)

Şekil 4.5 : (a) Global histogram eşitlenmesi sonrası görüntü (b) Histogram eşitlenmesi sonrası görüntünün histogramı.

Dijital görüntü işlemenin ön işleme aşamaları arasında görüntü iyileştirme yöntemleri girmektedir. İyileştirme teknikleri arasında görüntü yumuşatma yöntemi ile görüntülerdeki gürültüler giderilmesi için çeşitli filtreler geçirilmiştir. Gürültü genellikle insanın görsel algısı için istenmeyen bir durumdur ve bu gürültü görüntüdeki önemli özelliklerin ve ayrıntıların görülme miktarlarını azaltmaktadır. Gürültünün etkisinin azaltılmasından sonra gerçekleştirilecek olan kenar belirleme, nesne tanıma vb. gibi görüntü işleme tekniklerinin başarılı sonuçlar verebilmesi için gürültünün etkisinin optimum şekilde giderilmesi gerekmektedir. Aşağıda çeşitli görüntüler üzerinden farklı gürültü azaltım filtreleri uygulanmıştır.

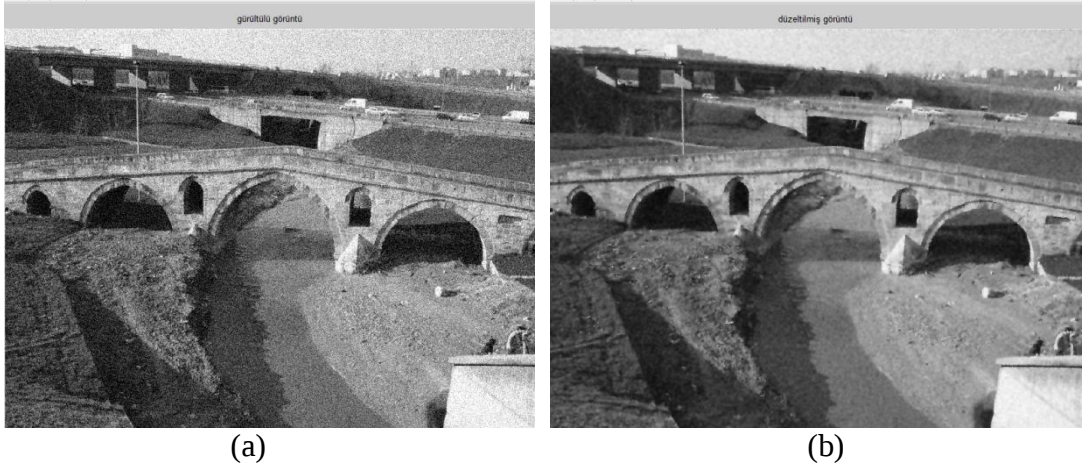
Alçak geçirgen bir filtre olan Gauss filtresi geçirilerek görüntülerden nasıl sonuçlar elde edileceğine bakılmıştır. Gauss filtresinde kullanıcının belirleyeceği maske

boyutunda bir matris ile tarar ve her bir matriste ağırlıklı ortalamayı hesaplar bu sebeble mean filtresine göre daha yumuşak blulaştırma sağlar. Aşağıda Şekil 4.6’da çeşitli standart sapmalar ile farklı filtre matrisleri geçirilmiştir.



Şekil 4.6 : (a) Çalışmada kullanılan örnek renkli görüntü (b) $\sigma = 1.5$ ve 5×5 Gauss filtresi (c) $\sigma = 3$ ve 3×3 Gauss filtresi (d) $\sigma = 10$ ve 7×7 Gauss filtresi.

MATLAB’de oluşturulacak lineer filtre maskesi için ($I = \text{fspecial}(\text{type}, \text{parametres})$) şekilde komut oluşturulur. Burada ‘type’ özel filtre tipini belirtir. Parametreler ise filtreyi tanımlayan değerlerdir. Yukarıda geçirilen Gauss filtresi type olarak belirlenir. Parametreler ise standart sapması pozitif olan ve boyutu $r \times c$ olan bir filtredir. Ayrıca Şekil 4.7’de 0.01 varyans değeri olan normal gürültülü görüntüye Gauss filtresi geçirilmiştir.



Şekil 4.7 : (a) Görüntüye 0.01 varyansla gürültü eklenmiş hali (b) Görüntünün 5x5 Gauss filtresi geçirildikten sonraki hali.

Averaj filtresi ile görüntüdeki her piksel yerine komşuları ile beraber ortalaması alınarak yeniden hesaplanır. Görüntüdeki gri düzeyler arasında keskin geçişler azalır; daha yumuşak geçişler söz konusu olur. Görüntü üzerindeki kenarlarda bulanıklaşmaya yol açarlar (Şekil 4.8). Ağırlıklı ortalama filtresi olan averaj filtresi MATLAB yazılımında ($I = \text{fspecial}('average', [r,c])$) şeklinde gösterilir. Averaj filtresinin varsayılan matris boyutu 3 x 3 tür. Eğer r,c yerine tek bir sayı yazılırsa kare matris filtredir (Şekil 4.9).



Şekil 4.8 : (a) 3 x 3 maske geçirilmiş averaj filtresi (b) 5 x 5 maske geçirilmiş averaj filtresi.

0.1111	0.1111	0.1111
0.1111	0.1111	0.1111
0.1111	0.1111	0.1111

(a)

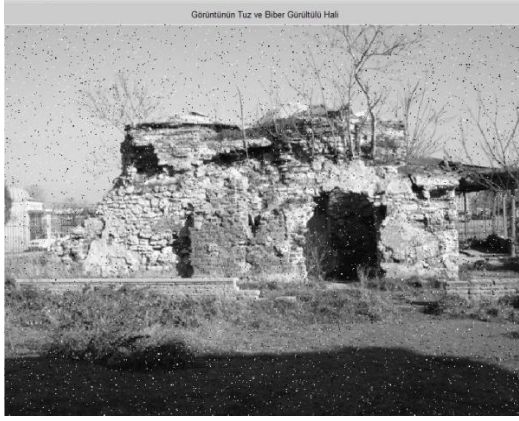
0.0400	0.0400	0.0400	0.0400	0.0400
0.0400	0.0400	0.0400	0.0400	0.0400
0.0400	0.0400	0.0400	0.0400	0.0400
0.0400	0.0400	0.0400	0.0400	0.0400
0.0400	0.0400	0.0400	0.0400	0.0400

(b)

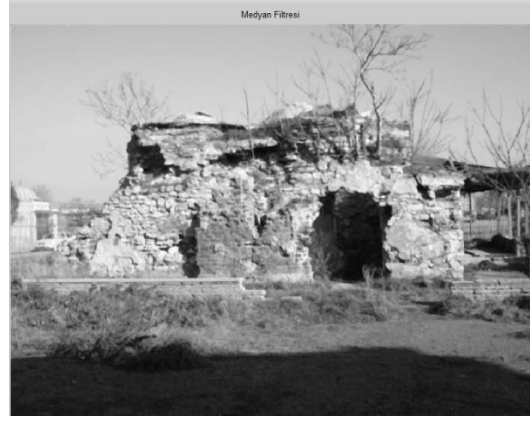
Şekil 4.9 : (a) 3 x 3 maske geçirilmiş averaj filtresinin değerleri (b) 5 x 5 maske geçirilmiş averaj filtresinin değerleri.

Bir görüntü üzerindeki gürültüyü temizlemenin bir diğer yolu da medyan filtresidir. Medyan filtre genellikle *tuz & biber* dediğimiz bozunma türünü gidermek için kullanılır. Tuz & biber gürültüsü; İmpulse gürültüsü veya binari gürültü olarak tanımlanır. Bu gürültü görüntüde keskin dağılımlı olmakla birlikte görüntü üzerinde belirgin olarak siyah ve beyaz noktalarşeklindedir. Tuz biber modeli tipik olarak kameranın algılayıcılarındaki piksel elemanlarının çalışmalarındaki bozukluklardan, hatalı bellek bölgelerinden ya da dijitize sürecindeki zamanlama hatalarından kaynaklanan gürültüleri modeller. Medyan filtre; Uzaysal çözünürlüğü bozmadan, kopuk (bağımsız) nokta veya çizgi gürültülerini temizlemek için kullanışlıdır. Bu nedenle ikili (binary) gürültülerde başarılı olmasına rağmen Gauss gürültüsünde kullanışlı değildir.

MATLAB’de: tuz, biber gürültüsünün eklenmesi ($A = \text{imnoise}(b, \text{'salt \& pepper'}, d)$) fonksiyonu ile işlenir (Şekil 4.10). Burada b orijinal görüntü, d ise tuz-biber gürültüsünün derecesidir. Default değeri ise 0.05’dir ve ayrıca d 0.1’den büyük olmamalıdır.

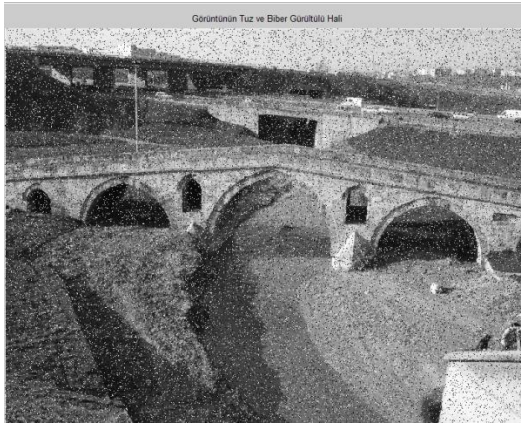


(a)

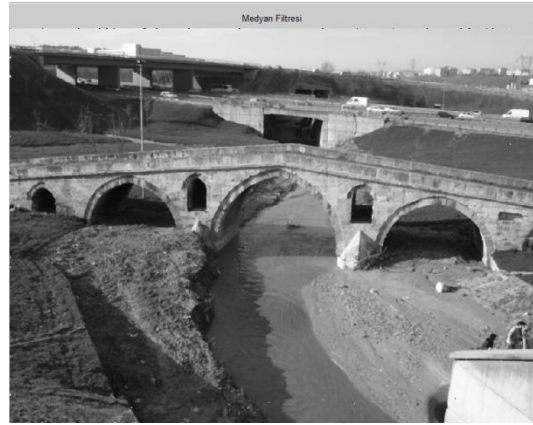


(b)

Şekil 4.10 : (a) 0.02 derece ile tuz & biber gürültüsü eklenmiş görüntü (b) 3 x 3 maskesi ile medyan filtresi geçirildikten sonra oluşan görüntü.



(a)



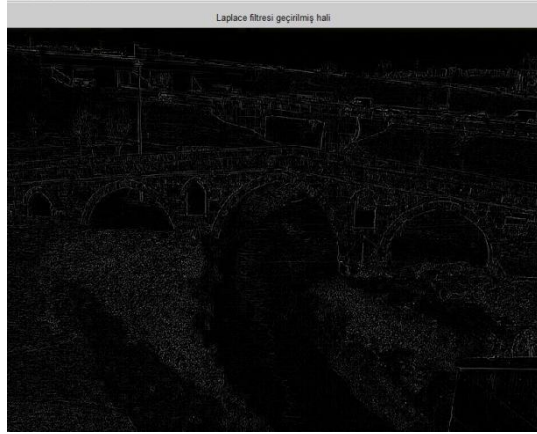
(b)

Şekil 4.11 : (a) 0.1 derece ile tuz & biber gürültüsü eklenmiş görüntü (b) 3 x 3 maskesi ile medyan filtresi geçirildikten sonra oluşan görüntü.

Görüntü iyileştirme teknikleri arasından bir diğer yöntem keskinleştirme filtreleridir. Bunlardan ikinci türeve dayanan laplace filtresi Şekil 4.12 ve 4.13’de örnek renkli görüntüler üzerinde uygulanmıştır. Laplace filtresi MATLAB’da (`I= fspecial ('laplacian', alpha)`) şeklinde komutuna dayanarak uygulanır. Buradaki α değeri olarak [0 -1] aralığında olmalıdır. Laplace filtresi için varsayılan α değeri 0.2 dir.



(a)

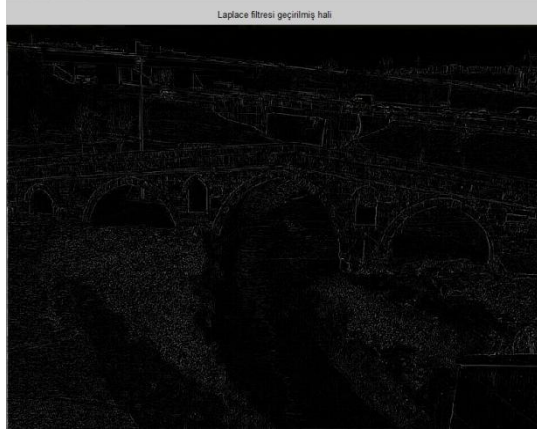


(b)

Şekil 4.12 : (a) Çalışmada kullanılmış renkli görüntülerden biri (b) $\alpha = 0.2$ kullanılarak laplace filtresi geçirildikten sonra oluşan görüntü.



(a)



(b)

Şekil 4.13 : (a) Çalışmada kullanılmış renkli görüntülerden biri (b) $\alpha = 0.5$ kullanılarak laplace filtresi geçirildikten sonra oluşan görüntü.

Keskinleştirme filtrelerinden bir diğeri unsharp filtresi Şekil 4.14 ve 4.15'te renkli görüntüler üzerinden uygulaması gösterilmiştir.

Unsharp filtresi, α parametresi ile laplace filtresinin negatifinden elde edilir. Unsharp filtresinin α parametreside $[0 -1]$ aralığında değer alır, varsayılan α değeri 0.2 dir.



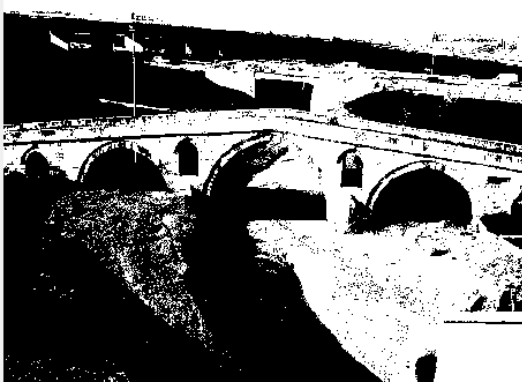
Şekil 4.14 : (a) Çalışmada kullanılmış renkli görüntülerden biri (b) $\alpha = 0.2$ kullanılarak unsharp filtresi geçirildikten sonra oluşan görüntü.



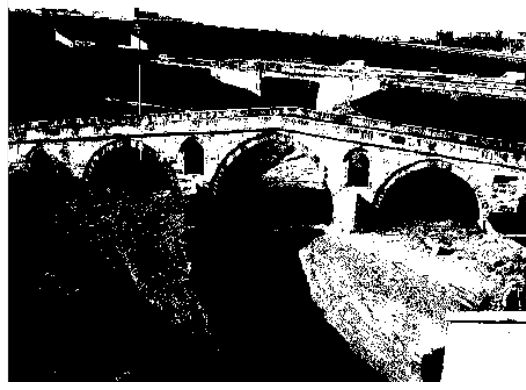
Şekil 4.15 : (a) Çalışmada kullanılmış renkli görüntülerden biri (b) $\alpha = 0.5$ kullanılarak unsharp filtresi geçirildikten sonra oluşan görüntü.

4.1.2 Eşikleme ve morfolojik işlemler

Eşikleme yöntemi dijital görüntü işlemenin en çok kullanılan konuları arasına girmektedir. Eşikleme yöntemleri arasında en fazla kullanılan klasik eşik değer belirleme ile otsu metod algoritması bu çalışmadaki görüntüler üzerinde denenmiştir. Klasik eşik değer belirleme yöntemi gri düzeyli bir görüntüden ikili görüntü (iki renkten oluşan) oluşturmak için kullanılır. Eşik değer belirleme işlemini gerçekleştirirken, her bir piksel bir obje gibi değerlendirilir ve gri değerinin belirlenen eşik değerinden büyük olup olmadığına dayanır. Genellikle büyük olması durumunda, 1 yani obje, olmaması durumunda ise 0, yani arka plan olarak değerlendirilir. Şekil 4.16’da klasik eşik değer yöntemi gösterilmiştir.



(a)



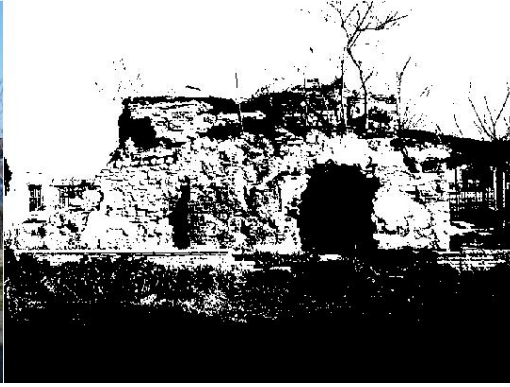
(b)

Şekil 4.16 : (a) 122 eşik değeri sonucunda oluşan görüntü (b) 145 eşik değeri sonucunda oluşan görüntü.

Ayrıca MATLAB yazılımının görüntü işleme fonksiyonlarından “im2bw” komutu kullanılarak gri seviyeli görüntüler binari görüntü haline dönüştürülebilir. “im2bw” fonksiyonu bir eşikleme sınır değeri alır. Sınır değerini histogram ile bulabiliriz. MATLAB’de ($I = \text{im2bw}(f, T)$) fonksiyonu kullanılır. Burada f , gri seviyeli görüntü olmakla birlikte, T değeri ise sınır değeridir. Sınır değeri $[0,1]$ aralığındadır (Şekil 4.17).



(a)



(b)

Şekil 4.17 : (a) Orijinal görüntü (b) 0.6 sınır eşik değeri sonucunda oluşan görüntü.

Bu tez çalışmasında uygulanan diğer bir eşikleme yöntemi otsu eşikleme metodudur. MATLAB’de tanımlı graythresh fonksiyonunu çağırarak görüntü için otomatik olarak ideal bir eşik sınırı elde edilir. Graythresh fonksiyonu otsu yöntemi ile çalışır. Otsu yöntemi histogram bazlı bir yöntemdir. Bu yöntemde algoritma önce görüntünün histogramını hesaplar, ardından sigma kareyi maksimum yapan bir değer döner (Şekil 4.18).



Şekil 4.18 : (a) Orijinal görüntü (b) Otsu eşik metoduna göre 0.4863 sınır eşik değeri sonucunda oluşan görüntü.

Tez çalışmasında kullanılan görüntüler karmaşık olup görüntünün her tarafında, aynı gri değere sahip olmayan objelerden ve aynı gri değere sahip olmayan arka plandan oluşmuştur. Bu da eşik değeri belirleme sonucunda çıkan görüntüde bazı yerlerde doğru bir bölütleme olmadığını göstermektedir.

Yapılan çalışmada, eşik değeri belirleme yöntemlerinin ayırt edemediği detayları ayırt edebilmek için matematiksel morfolojik operatörler kullanılmıştır. Matematiksel morfolojik operatörler, eşik değeri belirleme yöntemlerinin ayırt edemediği detayların piksel büyüklüklerini ve homojenliğini arttırmıştır.

MATLAB’de komut satırına yazılan *strel*, morfolojik işlemlerde kullanılır. *Strel* ball, diamond, disk, line, rectangle, square gibi çeşitleri bulunmaktadır. Morfolojik yapısal bir filtre elemanıdır. Morfolojik işlemlerin hangi şekil ve parametrelerle uygulanacağı *strel* ile belirlenir. Kullanımı, *strel*(filtreleme şekli, parametre) şeklindedir. Örneğin; $I = \text{strel}('disk', R)$ ifadesinde R yarıçaptır. $I = \text{strel}('square', L)$ L ise karenin bir kenarının uzunluğunu gösterir.

Uygulamada kullanılan görüntülerde her görüntü için farklı *strel* değerleri belirlenebilir. Bu çalışmada en uygun *strel* disk olarak belirlenmiştir. Binari hale getirilmiş olan görüntüyü Şekil 4.19’da ayırt edemediğimiz noktalardan kurtulmak amacıyla erozyona uğratılır. İlk olarak belirli bir *strel* belirlenerek MATLAB’de “*imerode*” komutuyla erozyona uğratılır (Şekil 4.21).



Şekil 4.19 : Çalışmada kullanılan gri seviyeli görüntülerden biri.

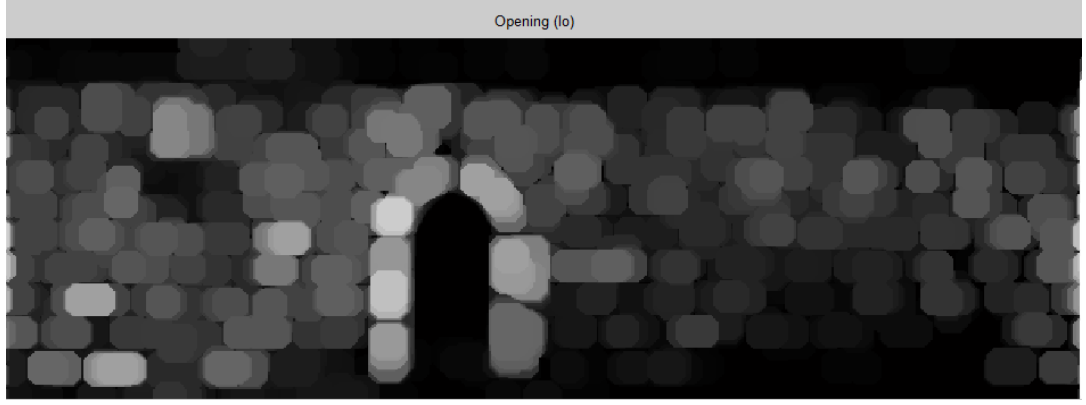


Şekil 4.20 : Eşik değeri 0.4 olan binari görüntü.



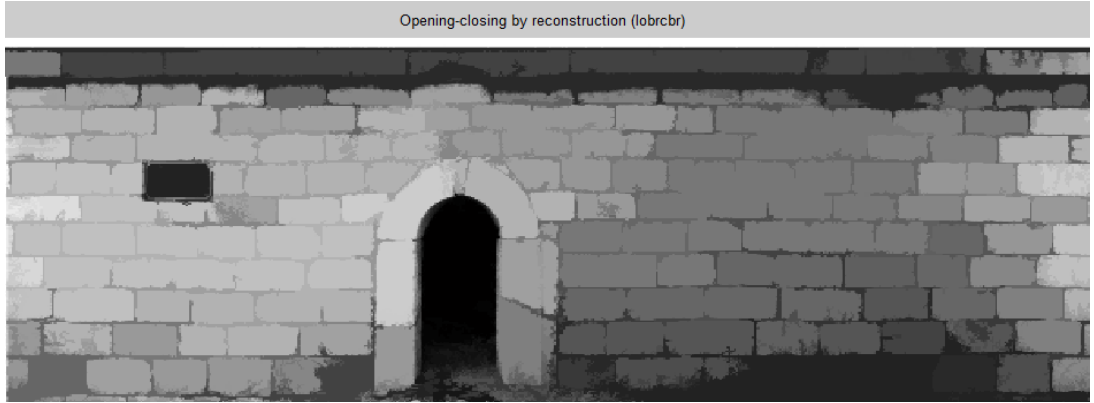
Şekil 4.21 : Strel (disk ,3) değeri 3 ile erozyon işlemi sonucunda oluşan binari görüntü.

Görüntüde küçük parçalardan kurtulmak için açma işlemi kullanılmaktadır. MATLAB’de açma fonksiyonu “ bwareaopen” komutu ile gerçekleştirilmektedir (Şekil 4.22).



Şekil 4.22 : Strel (disk,3) değ r sınırı ile a ma i lemi sonucunda olu an binari g r nt .

G r nt de erozyon i leminin vermi  oldu u bo lukları kapatmak i in geni letme i lemi kullanılmaktadır. Bu i lem i in “imdilate” komutu ile ger ekle tirilmektedir. Kapama i lemi ile birlikte g r nt m z i erisindeki noktalar birbirlerini kapatarak, g r nt deki ana hatlar daha da dolgunla tırılmaya  alı ıldı (Şekil 4.23).



Şekil 4.23 : A ma kapama i lemi sonucunda olu an g r nt .

A ma kapama i lemi sonucunda olu an g r nt   zerinde keskinle tirme filtresi ge irilerek g r nt n n netle tirilmesi sa lanmaya  alı ılmı tır (Şekil 4.24).



Şekil 4.24 : Açma kapama sonucunda oluşan görüntü üzerinden keskinleştirme filtresi geçirildikten sonra oluşan görüntü.

Morfolojik çalışmalar sonucunda çıkan görüntü ile Şekil 4.23'teki gri seviyeli görüntü arasında alınan fark görüntüsü ile detayların çıkarılması denenmiştir (Şekil 4.25).



Şekil 4.25 : Morfolojik işlemler sonucunda çıkan detay görüntüsü.

4.1.3 Detay çıkarımı

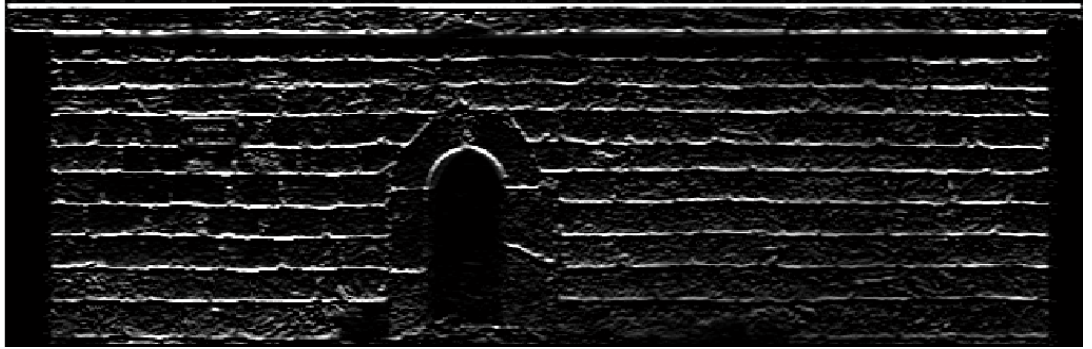
Nesnelerin tanımlanması için sınırlarının belirlenmesi gerekliliğinden, kenar bulma, dijital görüntü işleme sistemleri için en önemli aşamalardan biri olmuştur. Görüntü içinde belirgin geçişler nesne kenarlarını gösterdiğinden, kenar bulma algoritmaları bu geçişleri bulması amaçlanır. Detay çıkarımında kenar belirleme algoritmaları önem teşkil etmektedir. Kenar belirleme aşamasında görüntü gradientinin oluşturulması gerekir. Seçilen kenar belirleme yöntemine göre uygulanan kernellere bağlı olarak, görüntü gradienti x ve y yönünde filtreleme yapar (Şekil 4.27 ve Şekil 4.28).



Şekil 4.26 : Çalışmada kullanılan görüntülerden biri.



Şekil 4.27 : Prewitt gradientinin G_x yönünde oluşan görüntü.

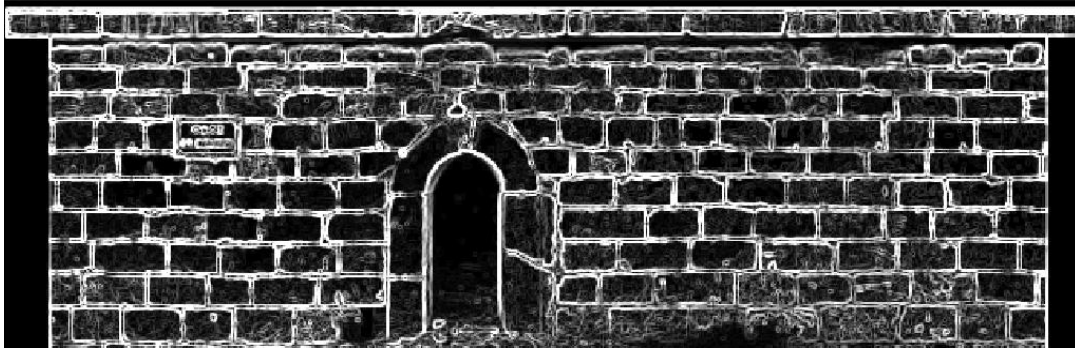


Şekil 4.28 : Prewitt gradientinin G_y yönünde oluşan görüntü.

Sobel ve Prewitt kenar belirleme operatörlerinin G_x ve G_y yönlerinde filtreleme yaptıktan sonra bu yönlerdeki filtreler toplamı Şekil 4.29 ve Şekil 4.30'da gösterilmiştir.



Şekil 4.29 : Prewitt gradientinin $G_x + G_y$ yönünde oluşan görüntü.



Şekil 4.30 : Sobel gradientinin yönünde oluşan görüntü.

Seçilen kenar belirleme algoritmasına uygun olarak görüntü gradientinin oluşumundan sonra belli bir eşik değeri altında detay çıkarımı çalışılmıştır. MATLAB’de Sobel kenar belirleme döngülerle uğraşmadan hazır komutlarla;

`BW=edge(I,'Sobel')`

`BW=edge(I,'Sobel',thresh)`

`BW = edge(I,'Sobel',thresh,direction)` şeklinde yapılabilir. Burada “thresh” eşik değeri, “direction” ise seçilecek yön olarak gösterilir. Eğer komuta bu parametreler girilmez ise program varsayılan değeri alır (Şekil 4.31).



Şekil 4.31 : Sobel kenar belirleme operatöründe 45 derece yön ve 150 eşik değeri sonucu oluşan görüntü.

Prewitt kenar belirleme operatörü de Sobel gibi MATLAB programında aynı şekilde çağırılarak koşturulur. Şekil 4.32 ve Şekil 4.33’de Sobel ile Prewitt kenar belirleme operatörlerinin belli bir eşik değeri altında çıkan detaylar gösterilmiştir.

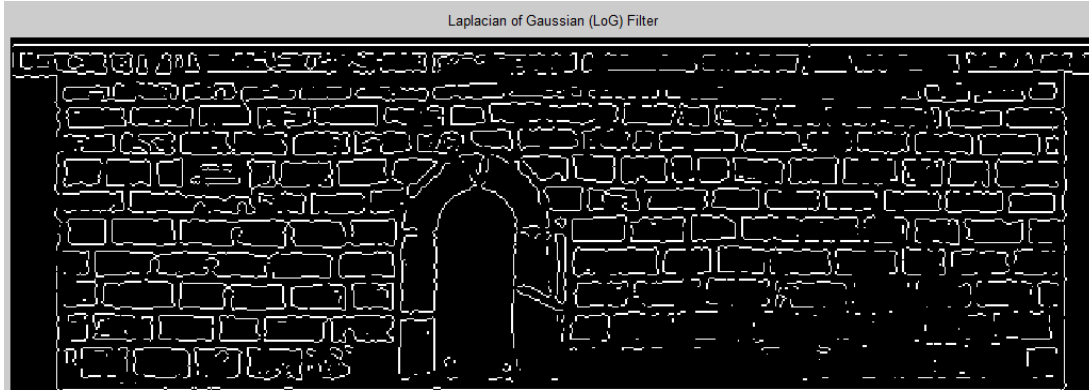


Şekil 4.32 : Prewitt kenar belirleme operatöründe 0.1 eşik değeri sonucu oluşan görüntü.

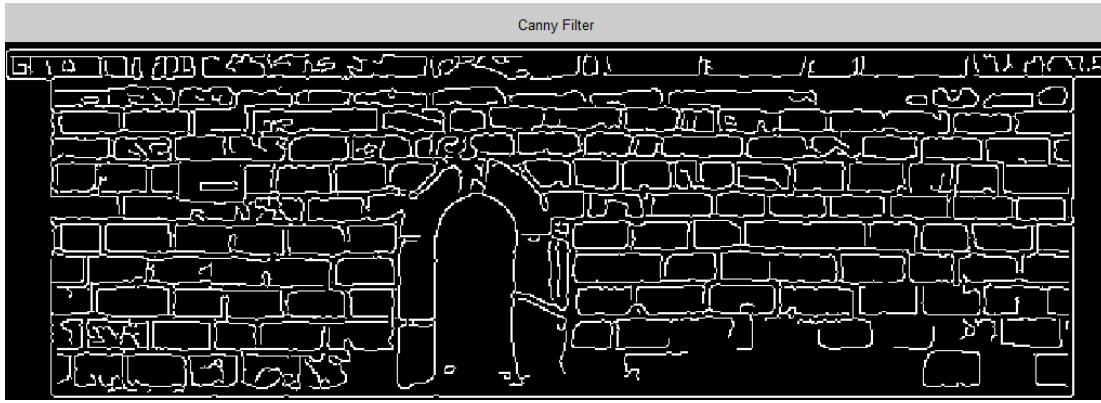


Şekil 4.33 : Sobel kenar belirleme operatöründe 0.1 eşik değeri sonucu oluşan görüntü.

Diğer kenar belirleme algoritmalarından Canny ve log (Laplace of gauss) kenar belirleme yöntemleri Şekil 4.34 ve Şekil 4.35'te gösterilmiştir.



Şekil 4.34 : Log (laplacian of gauss) kenar belirleme operatörü sonucu oluşan görüntü.



Şekil 4.35 : Canny kenar belirleme operatöründe 0.1 eşik değeri sonucu oluşan görüntü.

Bu görüntü için Sobel kenar belirleme yöntemi en uygun yöntem olduğu görülmüştür. Detay çıkarımında karşılaşılan en büyük sorunlardan biri arka plan ile nesnede ayırt edilemeyen kısımların mevcut olmasıdır. Bu duruma objenin üzerindeki yosunlaşma ve gölge etkisi gibi faktörler neden olmuştur.

Bu çalışma kapsamında kullanılan metod ve filtreleme yöntemleri açısından elde edilen en uygun sonuç Sobel ile elde edilmiştir. Log ve Canny metodu Gauss kullanmasına rağmen uygun sonuç vermediği görülmüştür. Yazılan kodlar MATLAB m.file dosyası içerisinde yazılmıştır. Guide kısmında arayüz tasarlanmıştır. Bu m.file dosyaları guide editörü arasında ilişki kurularak guide kısmının çalıştırılması sağlanmıştır (Şekil 4.36).



Şekil 4.36 : Arayüz tasarımı.

5. SONUÇLAR VE ÖNERİLER

Dijital görüntü işleme teknikleri gün geçtikçe çok farklı ve geniş uygulama alanları bulmaktadır. Bu alanlardan biri de yersel fotogrametri uygulamalarıdır. Yersel fotogrametri uygulamaları arasında; tarihi eser görüntülerinin onarılması, kalitesinin artırılması tarihimizin yaşatılması ve gelecek nesillere aktarabilmesi açısından önem taşımaktadır. Görüntüleri çekilen tarihi eserlerin orijinalleri de aynı şekilde zamanın verdiği etkiyle çevresel koşullar nedeniyle, kırılmış, dökülmüş, üzerindeki önemli ayrıntıları kaybetmiş hatta yok olmaya yüz tutmuş olabilmektedir. Eserin geçmişten günümüze kazandırılması için yani restorasyon çalışmalarına katkıda bulunması için kullanılacak tek seçenek bazen bu görüntüler olabilmektedir. Bu durumda görüntülerden dijital görüntü işleme tekniklerini kullanarak detaylarının en iyi şekilde çıkarılması restorasyon gibi çeşitli yersel fotogrametri uygulama alanları için önemlidir.

Bu çalışmalarda elde edilen sonuçlar incelendiğinde, uygulanan filtre metotları içerisinde medyan filtresi tuz – biber gürültüsünü giderirken Gauss filtresi normal gürültüleri gidermekle birlikte medyan filtresi ikili görüntülerde Gauss filtresine göre daha başarılıdır. Diğer bir taraftan Gauss filtresi avaraj (mean) filtresine göre görüntüde daha yumuşak blulaştırma sağlar. Görüntü yumuşatan filtrelerin yanı sıra görüntüyü keskinleştiren unsharp filtresi alfa parametresine bağlı olarak görüntü kalitesini arttırmıştır. İdeal görüntüler elde edildiğinde nesnenin ayırt edilebilmesi için kullanılan eşikleme değeri her görüntü ve yöntemle göre farklılık gösterir. Nesne ile arka planın ayırt edilemediği karmaşık görüntülerde eşikleme yöntemi yeterli olmayıp morfolojik işlemler ve kenar belirleme yöntemleri ile birlikte daha iyi sonuç elde edilebilir. Bu çalışmada uygulanan kenar belirleme yöntemlerinden Sobel yöntemine en yakın sonuç veren yöntem Prewitt yöntemidir ve Prewitt kenar belirleme yöntemi, Sobel yöntemine göre görüntü daha gürültülüdür. Ayrıca Canny metodu Gauss filtresini kullanmasına rağmen uygun sonuç vermediği görülmüştür.

Yapılan bu çalışmada kullanılan algoritmaların çalıştırılması MATLAB programıyla gerçekleştirilmiştir. MATLAB, oluşturduğu matematiksel işlem ortamı yanında görüntü

işleme gibi birçok alandaki güncel algoritmaları hazır fonksiyonlar olarak sağlamaktadır. Bu sayede kullanıcılara karmaşık algoritmalarından kaçınmasını sağlayarak daha hızlı ve daha test edilebilir şekilde kolaylık sağlamaktadır. Ayrıca görsel arayüz tasarlama imkanı ile hazır fonksiyonlar birleştirilerek amaca özel bir yazılım gereci oluşturulabilmektedir.

Bütün bunlar göz önünde bulundurulduğunda bu çalışmanın amaç ve hedeflerine paralel olarak, dijital görüntü işleme teknikleri kütüphanesini içinde barındıran MATLAB yazılımının uygulanabilirliği, işlevselliğinin gösterilmesi amacıyla birbirinden farklı çeşitli mekanlarda çekilmiş görüntüler üzerinden gerçekleştirilmiştir. Bu tezde kullanılan algoritmalar genel olarak gürültü temizleme algoritmaları, görüntü netleştirme algoritmaları, morfolojik yöntemler ve kenar bulma algoritmaları olarak sınıflandırılmıştır. Bu algoritmalar arasında en iyi sonuç veren algoritmalar seçilerek, çeşitli görüntülere uygulanmıştır. Buradan her görüntüde çeşitli olumsuz koşullar olabilmekte ve her görüntüden uygulanmak istenen yöntem farklılık arz etmektedir. Bundan dolayı eşikleme yöntemlerinde farklı bir eşik değeri, görüntü zenginleştirme yöntemlerinde farklı parametreler, morfolojik işlemlerde matris üzerinde geçirilen farklı yapıtaşları elemanları ve kenarlar için farklı kenar belirleme algoritmaları kullanılabilmektedir. Kullanılan tekniklerin birbirlerine göre avantajları ve dezavantajları olmakla birlikte genel olarak elde edilen sonuçlar optimum düzeydedir.

KAYNAKLAR

- [1] **Gonzalez, R.C., Woods, R.E.** (2002). Digital Image Processing, 3rd Edition, Prentice Hall.
- [2] **Luhman, T., Robson, S. ve Kyle, S.** (2011). Close Range Photogrammetry Principles, Methods and Applications.
- [3] **Qidwai, U. ve Chen, C.H.** (2010). Digital image processing: an algorithmic approach with MATLAB, CRC Press.
- [4] **Krutsch, R. ve Tenorio, D.** (2011). Histogram Equalization, Uygulama Notları
- [5] **Polat, R.** (2007). Biyomedikal Görüntü Bölütleme, Yüksek Lisans Tezi, Fırat Üniversitesi Bilgisayar Mühendisliği Anabilim dalı, Elazığ.
- [6] **Bayram, B.** (t.y.). *Dijital Görüntü İşleme*, Lisans Ders Notları Yıldız Teknik Üniversitesi Jeodezi ve Fotogrametri Mühendisliği Bölümü, İstanbul.
- [7] **Bellanger, M.** (2000). Digital Processing of Signal: Theory and Practice, John Wiley and Sons, USA.
- [8] **Öztaş, O.** (t.y.). *Görüntü İşleme*, Lisans Ders Notları İstanbul Üniversitesi Bilgisayar Mühendisliği Bölümü, İstanbul.
- [9] **Boykov, Y., Jolly, M.P.** (2001). Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images. *In Proceedings of the International Conference on Computer Vision (ICCV)*, Cilt I, Sayfa: 105–112.
- [10] **Uslu, R.** (2007). Elektronik Bir Hücreli Yapay Sinir Ağı Gerçekleşmesi Olan ACE16K Üzerinde Görüntü Bölütleme. Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.
- [11] **Morse, B.S.** (2000). *Digital Image Processing*, Ders Notları, Brigham Young Üniversitesi Bilgisayar Mühendisliği Bölümü, Utah.
- [12] **Jahne B.** (2002). Digital Image Processing. Springer, Heidelberg.
- [13] **Acar, U.** (2011). *Uydu Görüntüleri ve Tıbbi Görüntülerden Benzer Görüntü İşleme Teknikleriyle Bilgi Çıkarımı*, Doktora Tezi, Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul.
- [14] **Otsu, N.** (t.y.). A Threshold Selection Method from Gray-Level Histograms, *IEEE Transactions on Systems, Man, and Cybernetics*.
- [15] **URL-1** < <http://www.cse.unr.edu/~bebis/CS791E/Notes/Thresholding>>, alındığı tarih: 15.10.2014.
- [16] **Altuntaş, C. Ve Çorumluoğlu, Ö.** (2011). Filtreleme Yöntemi ile Digital Görüntü Zenginleştirme ve Örnek Bir Yazılım, *Teknik-Online Dergi*, Cilt. 10, Sayı 99-106

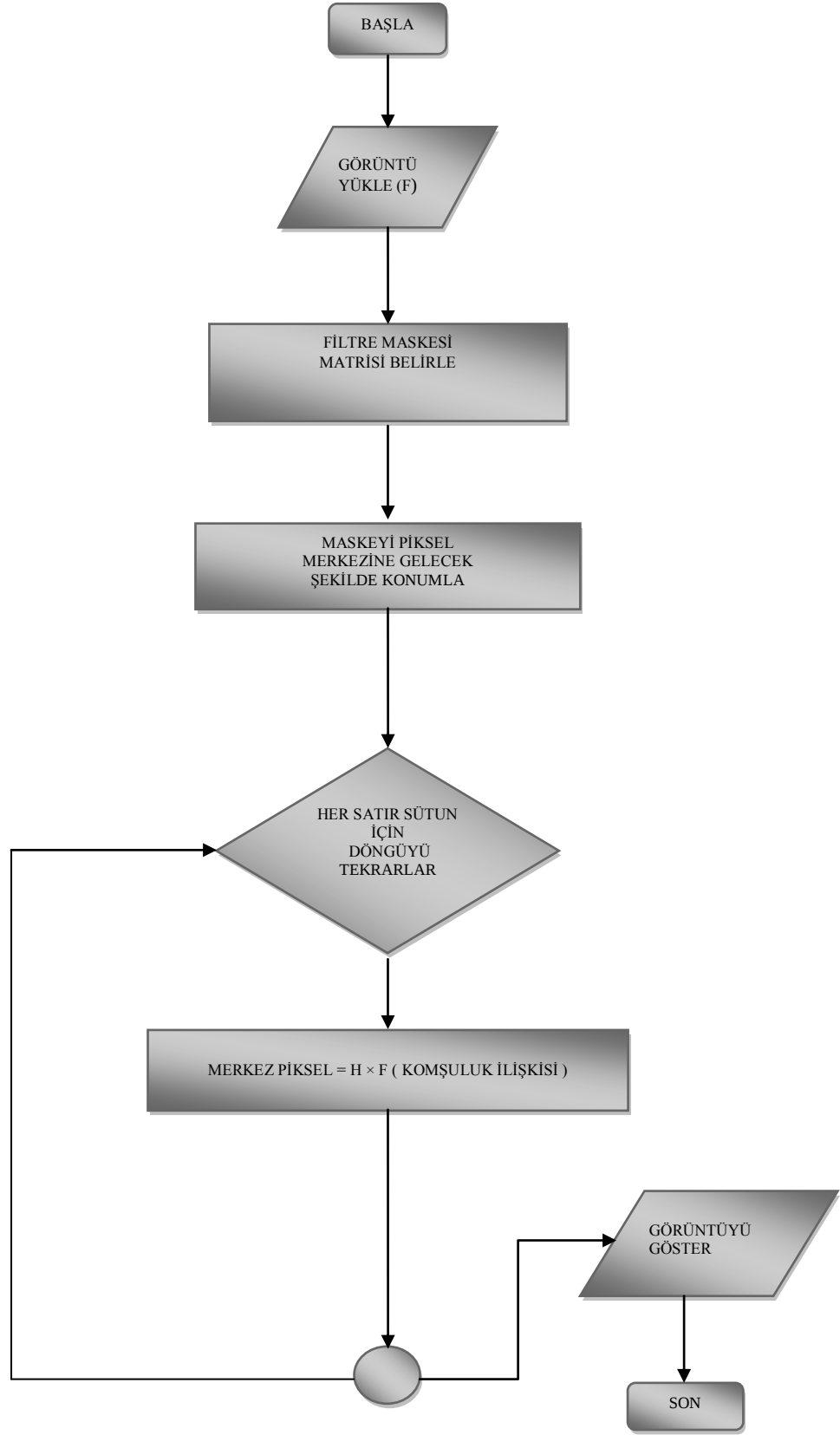
- [17] **Önder, M.** (1997). *Uydu görüntülerinden Ulusal Coğrafi Bilgi Sistemine Temel Oluşturacak Nitelikte Topografik Harita Üretimine Yönelik Analiz ve Öneriler*, Doktora Tezi, Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul
- [18] **URL-2** <<http://homepages.inf.ed.ac.uk/rbf/HIPR2/gsmooth.htm>>, alındığı tarih: 20.10.2014.
- [19] **URL-3**
<https://www.cs.auckland.ac.nz/courses/compsci373s1c/PatricesLectures/Gauss%20Filtering_1up.pdf>, alındığı tarih: 20.10.2014.
- [20] **Young, I.T., Gerbrands, J.J., Viliet, L.J.V.** (2007). *Fundamentals of Image Processing*, Delft University of Technology.
- [21] **Gonzalez, R.C. ve Woods, R.E.** (2009). *Digital Image Processing Using Matlab*, Gatesmark.
- [22] **Gündüz, M.** (2006). *Dijital görüntülerin Radyometrik Özelliklerine Göre Vektörel Çizim Dosyalarının Oluşturulması*, Doktora Tezi, Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya.
- [23] **URL-4** <<http://www.bioss.ac.uk/people/chris/ch3.pdf>>, alındığı tarih: 01.11.2014.
- [24] **Serra J.** (t.y). *Image Analysis and Mathematical Morphology*, Academic Press, London.
- [25] **Acar U. Ve Bayram B.** (2009). *Building Extraction With Morphology*, *International Conference on Modelling Optimization and Computing*, Cilt. 38, Sayfa: 3573-3578.
- [26] **URL-5** <<http://homepages.inf.ed.ac.uk/rbf/HIPR2/morops.htm>>, alındığı tarih: 05.11.2014.
- [27] **Girod, B.** (2007). *Digital Image Processing*, Ders Notları, Stanford Üniversitesi Elektrik Mühendisliği Bölümü, Kaliforniya.
- [28] **Nixon, M. S.** (2012). *Feature extraction & Image Processing for Computer Vision*, London: Elsevier Academic.
- [29] **URL-6** <<http://tr.wikipedia.org/wiki/MATLAB>>, alındığı tarih: 14.11.2014.
- [30] **Demir, Ö.** (2006). *MATLAB Gereçleri ile Görüntü İşleme Uygulamaları*, Yüksek Lisans Tezi, Marmara Üniversitesi Elektronik Bilgisayar Eğitimi Anabilim dalı, İstanbul.
- [31] **URL-7** <<http://www.mathworks.com>> alındığı tarih: 14.10.2014.

EKLER

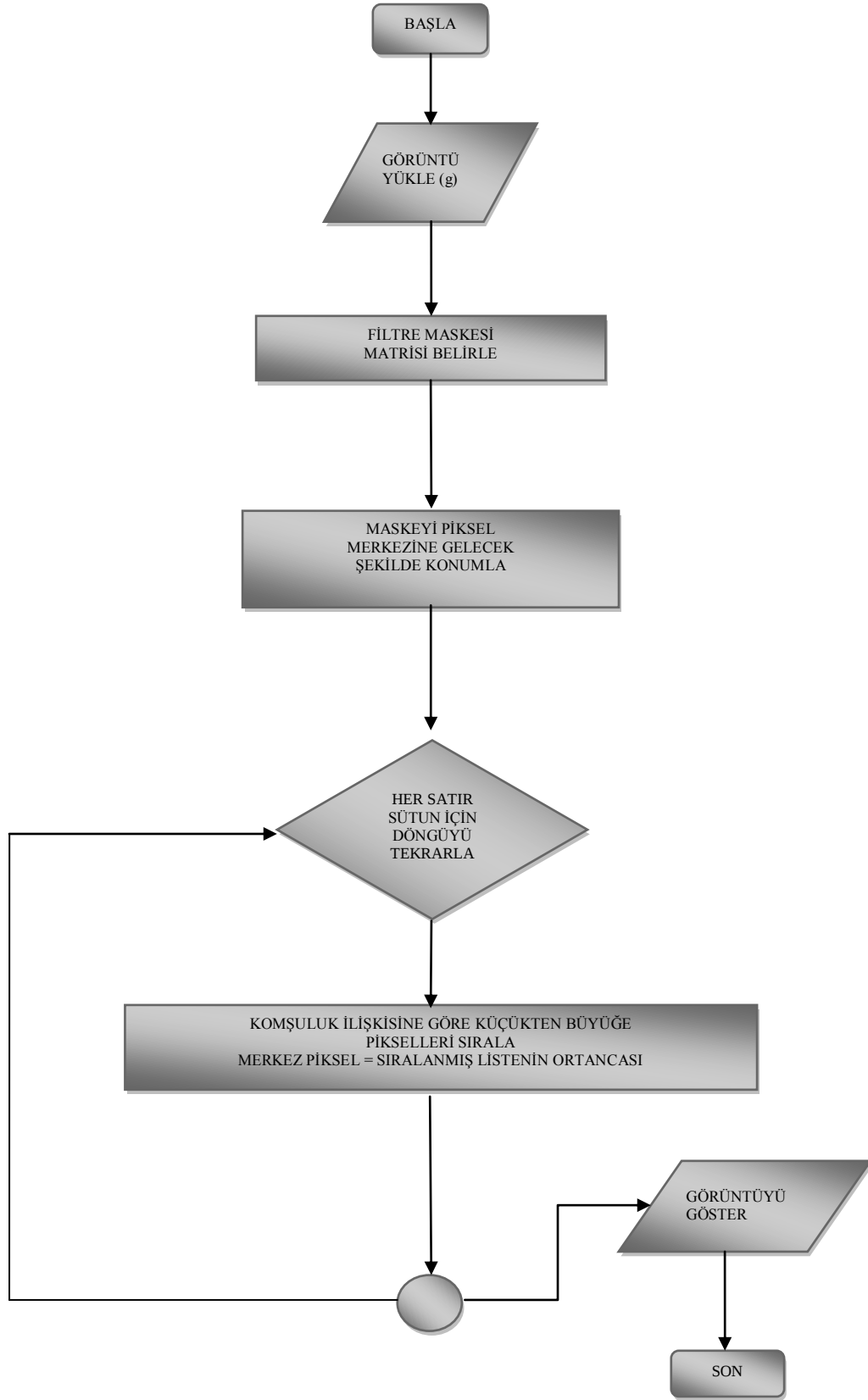
Sayfa

EK A: Kernel Tabanlı Uzaysal Filtre Algoritmasının Temel İşleyişi	61
EK B: Medyan Filtresinin Temel İşleyişi	62
EK C: Konvülasyon Kernelleri Kullanılarak Kenar Belirleme Algoritmasının Temel İşleyişi	63
EK D: Canny Kenar Belirleme Algoritmasının Temel İşleyişi	64
EK E: Morfolojik Operatörlerin Temel İşleyişi	65
EK F: Genel Akış Diagramı	66

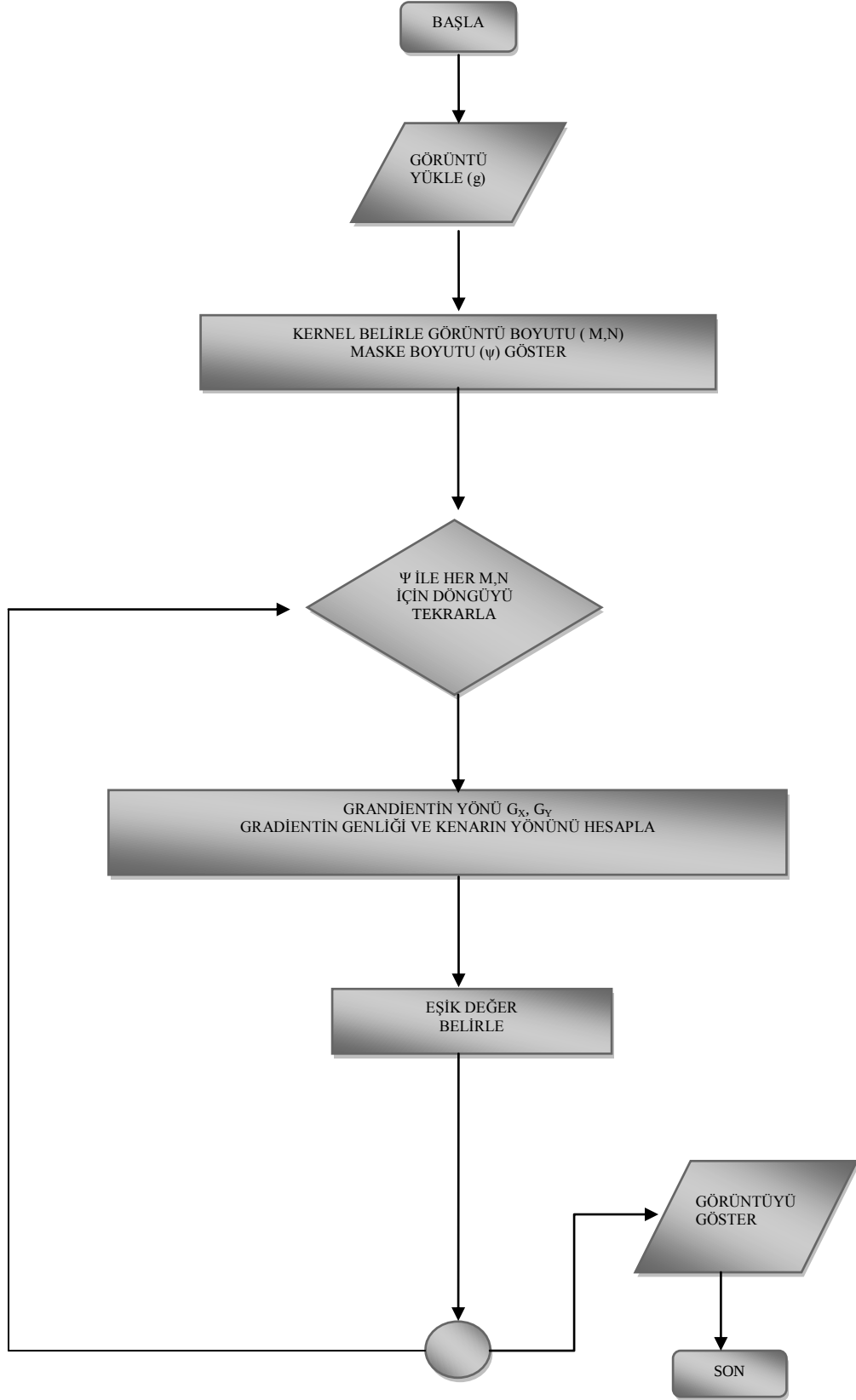
EK A: Kernel Tabanlı Uzaysal Filtre Algoritmasının Temel İşleyişi



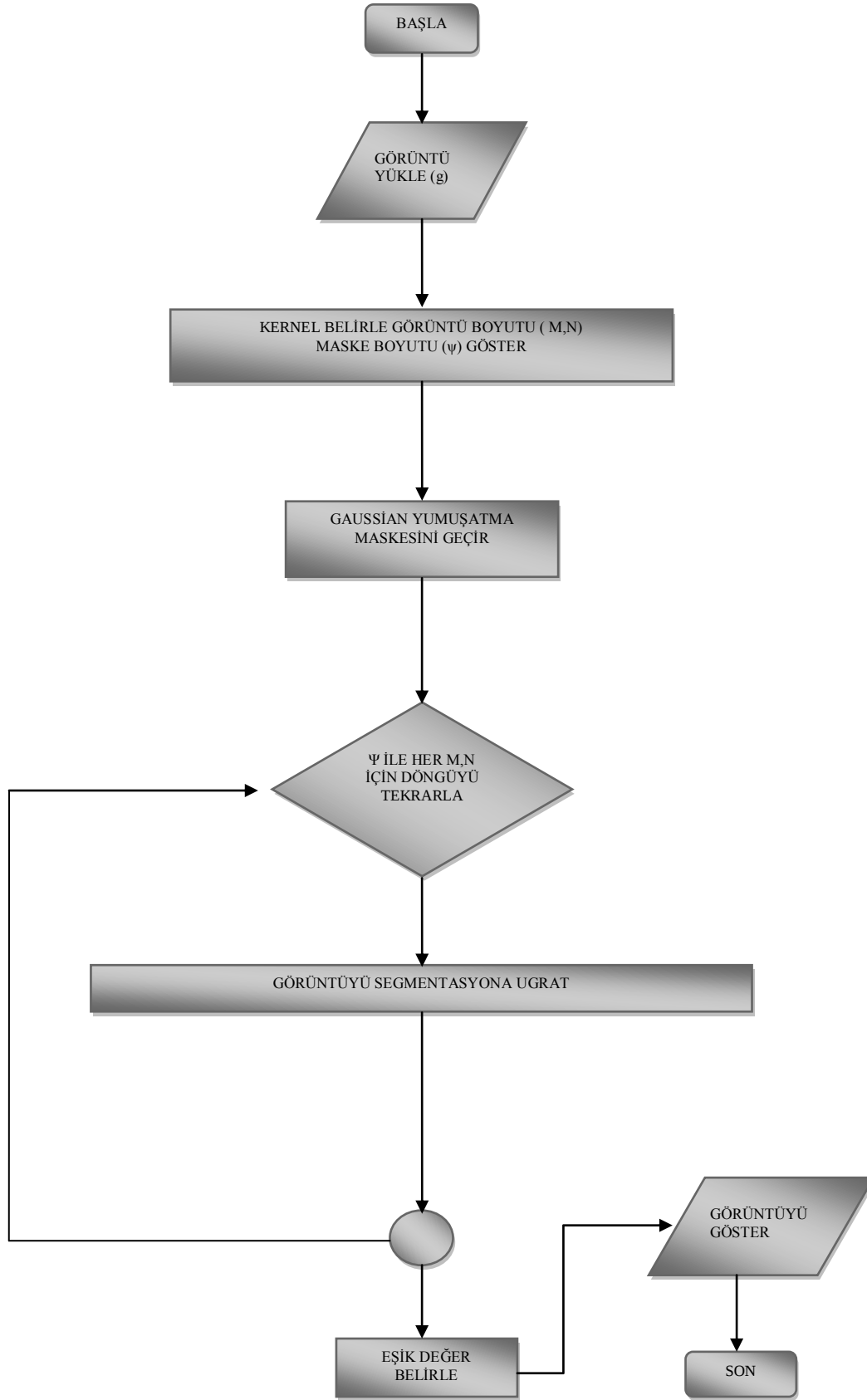
EK B: Medyan Filtresinin Temel İşleyişı



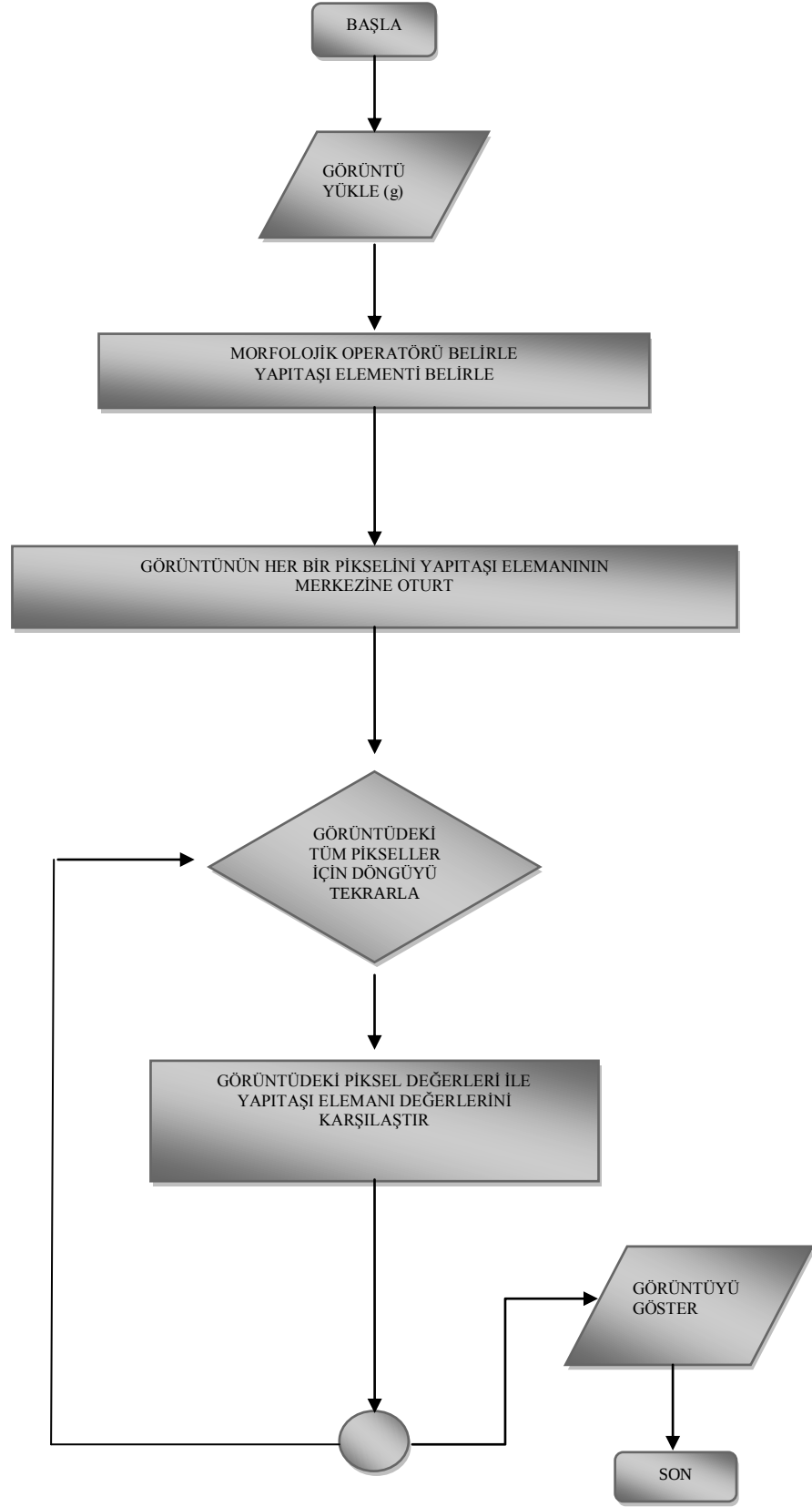
EK C: Konvülyasyon İşlemi Kullanılarak Kenar Belirleme Algoritmasının Temel İşleyişi



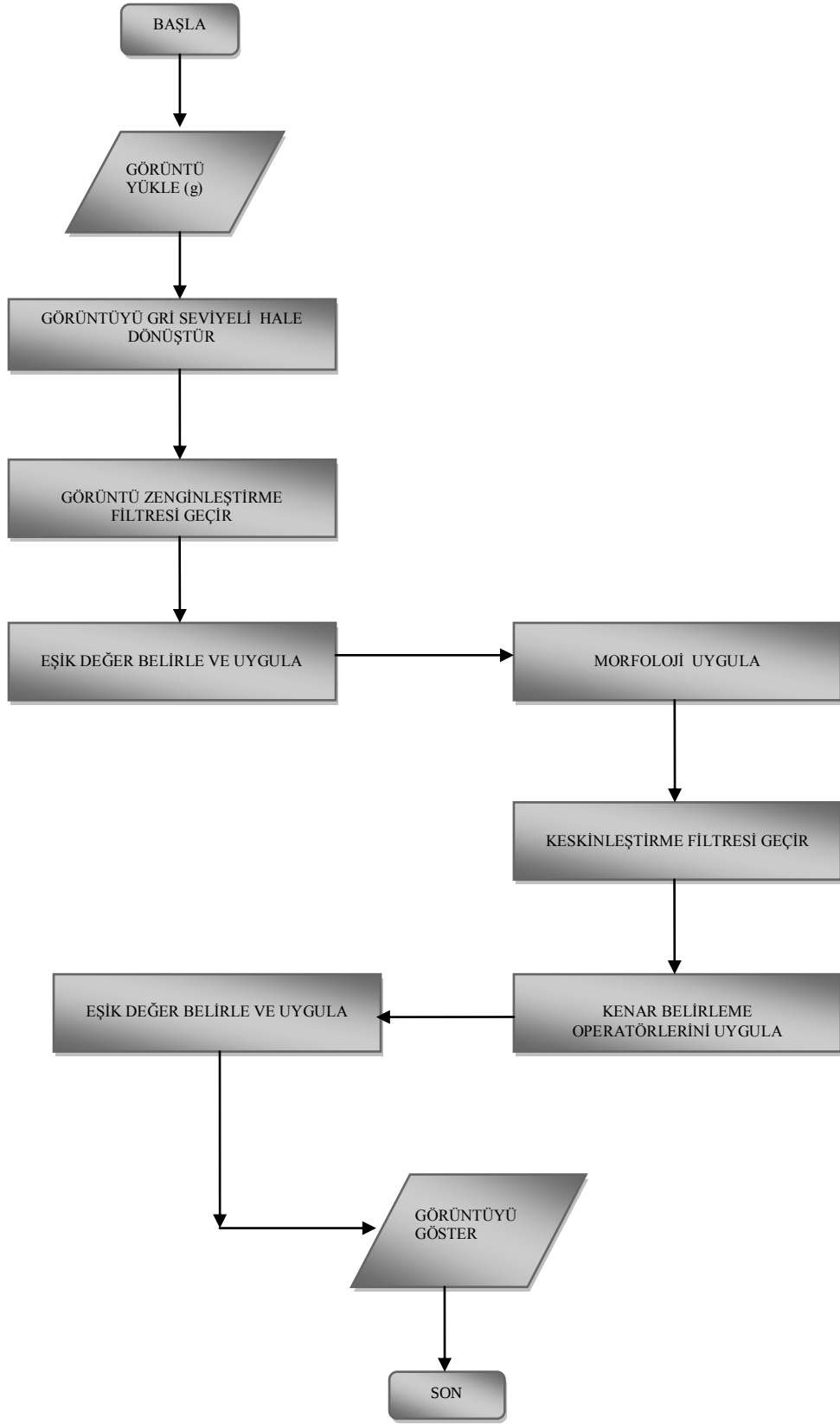
EK D: Canny Kenar Belirleme Algoritmasının Temel İşleyişi



EK E: Morfolojik Operatörlerin Temel İşleyişi



EK F: Genel Akış Diagramı



ÖZGEÇMİŞ



Ad Soyad: Güzide Miray PERİHANOĞLU

Doğum Yeri ve Tarihi: VAN, 01.01.1987

Adres: Fulya Mahallesi Üçobalar sokak No:14 Fulya Şişli/İSTANBUL

E-Posta: gmirayper@gmail.com

Lisans: Yıldız Teknik Üniversitesi Jeodezi ve Fotogrametri Mühendisliği