

KARADENİZ TEKNİK ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ

İSTATİSTİK VE BİLGİSAYAR BİLİMLERİ ANABİLİM DALI

İKİL GÖRÜNTÜLERDE NESNELERİN BASKIN NOKTALARININ  
BELİRLENMESİNDE OPTİMİZASYON ALGORİTMALARI

YÜKSEK LİSANS TEZİ

Çiğdem GÜNGÖR

OCAK 2012  
TRABZON

KARADENİZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

İSTATİSTİK VE BİLGİSAYAR BİLİMLERİ ANABİLİM DALI

İKİL GÖRÜNTÜLERDE NESNELERİN BASKIN NOKTALARININ  
BELİRLENMESİNDE OPTİMİZASYON ALGORİTMALARI

Çiğdem GÜNGÖR

Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünde  
“YÜKSEK LİSANS (İSTATİSTİK)”  
Unvanı Verilmesi İçin Kabul Edilen Tezdir.

Tezin Enstitüye Verildiği Tarih : 20/12/2011  
Tezin Savunma Tarihi : 05/01/2012

Tez Danışmanı : Yrd. Doç. Dr. Orhan KESEMEN

Trabzon 2012

Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü  
İstatistik ve Bilgisayar Bilimleri Anabilim Dalında  
Çiğdem GÜNGÖR tarafından hazırlanan

İKİL GÖRÜNTÜLERDE NESNELERİN BASKIN NOKTALARININ  
BELİRLENMESİNDE OPTİMİZASYON ALGORİTMALARI

başlıklı bu çalışma, Enstitü Yönetim Kurulunun 20/12/2011 gün ve 1434 sayılı kararıyla  
oluşturulan jüri tarafından yapılan sınavda

YÜKSEK LİSANS TEZİ  
olarak kabul edilmiştir.

Jüri Üyeleri

Başkan : Doç. Dr. Ali GANGAL .....

Üye : Doç. Dr. Türkan ERBAY DALKILIÇ .....

Üye : Yrd. Doç. Dr. Orhan KESEMEN .....

Prof. Dr. Sadettin KORKMAZ  
Enstitü Müdürü

## ÖNSÖZ

Günümüzde yapay zeka ve otomatik sistemlerde üzerinde en çok çalışılan alanlardan biri sayısal görüntü işlemedir. Bu alandaki çalışmalar tıbbi bilişimden askeri uygulamalara kadar birbirinden bağımsız pek çok disiplinde artarak devam etmektedir. Fakat hangi alanda olursa olsun genel amaç, sayısal görüntüden ilgili alan için faydalı bilgiyi elde etmektir. Bunun için görüntüyü oluşturan bileşenlerin iyi bir çözümlemesinin yapılması gereklidir. Görüntünün önemli bir bileşeni olan nesnelerin sınır noktaları arasında o nesneyi niteleyebilecek en önemli noktalar baskın noktalarıdır. Çoğu zaman baskın noktalar bir nesneyi tek başına temsil edebilmek için yeterli olabilmektedir. Baskın noktaları ait oldukları nesnenin sınır noktaları arasında en kritik bilgiyi taşıyan noktalarıdır. Nesneyi temsil etmekteki başarısı dolayısıyla da bu çalışma gibi literatürde birçok çalışmanın konusunu oluşturmuştur.

Bana bu konuyu öneren, sevdiren ve beni araştırmaya yönlendiren Karadeniz Teknik Üniversitesi, İstatistik ve Bilgisayar Bilimleri bölümü öğretim üyelerinden danışman hocam Yrd. Doç. Dr. Orhan KESEMEN' e çok değerli bilgilerini benimle paylaştığı için ve bilim insanı olma yolundaki ilk adımlarımı atarken yolumu aydınlattığı için gönülden teşekkürü borç bilirim. Ayrıca yetişmemde katkısı olan tüm bölüm hocalarıma saygı ve minnetlerimi arz ederim.

Son olarak, onlardan aldığım güç ile hayattaki tüm zorlukları yenebileceğime inandığım, beni hep seven ve destekleyen aileme en derin sevgi ve teşekkürlerimi sunarım.

Bu araştırmanın literatüre katkı sağlamış olmasını diler, sevgili okurların çalışmadan faydalanabilmelerini temenni ederim.

Çiğdem GÜNGÖR

Trabzon 2012

## **TEZ BEYANNAMESİ**

Yüksek Lisans Tezi olarak sunduğum “İkil Görüntülerde Nesnelerin Baskın Noktalarının Belirlenmesinde Optimizasyon Algoritmaları” başlıklı bu çalışmayı baştan sona kadar danışmanım Yrd. Doç. Dr. Orhan KESEMEN’ in sorumluluğunda tamamladığımı, verileri kendim topladığımı, analizleri ilgili laboratuvarlarda yaptığımı, başka kaynaklardan aldığım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiğimi, çalışma sürecinde bilimsel araştırma ve etik kurallara uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim. 20/12/2011

Çiğdem GÜNGÖR

## İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ .....	III
TEZ BEYANNAMESİ .....	IV
İÇİNDEKİLER .....	V
ÖZET .....	VIII
SUMMARY .....	IX
ŞEKİLLER DİZİNİ .....	X
TABLolar DİZİNİ .....	XIV
SEMBOLLER DİZİNİ.....	XV
1. GENEL BİLGİLER.....	1
1.1. Sayısal Görüntü İşlemede Temel Kavram ve Teknikler .....	4
1.1.1. Sayısal Görüntünün Histogramı.....	7
1.1.2. Sayısal Görüntülerde Koordinat Sistemi .....	8
1.1.3. Sayısal Görüntünün Gözet Değerleri.....	9
1.1.4. Boz Ölçekli Görüntüler (Yoğunluk Görüntüleri).....	9
1.1.5. İkili Görüntüler.....	11
1.1.6. Renkli Görüntüler.....	12
1.1.7. Özel Görüntüler.....	13
1.1.8. Histogram Eşitleme (Histogram Equalization) .....	13
1.1.9. Görüntü Eşikleme (Thresholding) .....	14
1.1.10. Gürültü Temizleme (Filtering Noise).....	15
1.1.11. Kayan Ortalama Süzgeci .....	16
1.1.12. Sıralı Süzgeçler .....	17
1.1.13. Oyuk Açma-Kapama (Opening-Closing) .....	19
1.1.14. Etiketleme ile Oyuk Kapama.....	22
1.1.15. Sınır Algılama (Edge Detection).....	23

1.2.	Nesne Analizi ve Baskın Nokta Algılama .....	25
1.3.	Nesnenin Şekil Kavramı .....	25
1.4.	Nesnelerin Etiketlenmesinde Sıralı Etiketleme Yöntemi .....	25
1.5.	Nesnenin Özellikleri.....	28
1.5.1.	Nesnenin Çevresi .....	28
1.5.2.	Nesnenin Alanı.....	30
1.5.3.	Nesnenin Kütle Merkezi .....	30
1.5.4.	Nesnenin Ana Ekseni .....	31
1.5.5.	Nesneler İçindeki Boşluklar.....	32
2.	YAPILAN ÇALIŞMALAR.....	33
2.1.	Nesne Sınırlarına Çokgensel Yaklaşım .....	33
2.1.1.	Baskın Nokta.....	33
2.2.	Ön İşlemler .....	35
2.3.	Yapay Çokgen Oluşturma.....	38
2.3.1.	Rastgele Çokgen Oluşturma .....	38
2.3.2.	Sayısal Doğru Oluşturma.....	39
2.3.3.	Sayısal Çokgen Oluşturma .....	41
2.4.	Baskın Noktaların Belirlenmesinde Maliyet Fonksiyonu ve Uyum Değeri .....	42
2.5.	Yerel Anlamda Baskın Nokta Algılama İçin Önerilen Yöntemler .....	43
2.5.1.	Yerel Yöntemler İçin Baskın Noktaların Doğruluk Kriteri.....	44
2.5.2.	Göreceli Uzaklıklar Yöntemi.....	44
2.5.3.	Kesişim Açısı Yöntemi.....	47
2.5.4.	Belirtme Katsayısı Yöntemi.....	49
2.5.5.	Kütle-Zemin Oranı Yöntemi.....	51
2.6.	Genel Anlamda Baskın Nokta Algılama Yöntemleri.....	53
2.6.1.	Kesin Çözüm.....	54
2.6.2.	Baskın Nokta Algılanmasında Genetik Algoritma .....	54

2.6.3.	Baskın Nokta Algılanmasında Tabu Arama Algoritması.....	60
3.	BULGULAR VE SONUÇLAR.....	63
4.	KAYNAKLAR .....	70
ÖZGEÇMİŞ		



Yüksek Lisans Tezi

ÖZET

İKİL GÖRÜNTÜLERDE NESNELERİN BASKIN NOKTALARININ  
BELİRLENMESİNDE OPTİMİZASYON ALGORİTMALARI

Çiğdem GÜNGÖR

Karadeniz Teknik Üniversitesi  
Fen Bilimleri Enstitüsü  
İstatistik ve Bilgisayar Bilimleri Anabilim Dalı  
Danışman: Yrd. Doç. Dr. Orhan KESEMEN  
2012, 72 Sayfa

Nesne çözümlemesi çalışmalarında nesnelere çevresel noktaları pek çok yararlı bilgi sağlamak ve içsel noktalara gerek duyulmaksızın tek başına nesneyi belirtebilen ve en yaygın kullanılan tanımlayıcılardan biri olarak görülmektedir. Bir nesnenin baskın noktaları o nesneyi en iyi tanımlayan özelliklerinden biridir. Baskın noktaların olduğu bölgeler cisimlerin iki boyutlu görüntüleri olan nesnelere sınır noktaları arasında en fazla hareketliliğin gözlemlendiği bölgelerdir. Bunlar aynı zamanda o nesne ile ilgili en önemli bilgiyi taşıyan karakteristik bölgelerdir. Bu çalışmada görüntüler içindeki nesnelere baskın noktalarının elde edilmesi üzerinde çalışıldı. Sayısal görüntülerdeki nesnelere baskın noktalarının algılanması problemi hem doğrudan nesnenin sınır noktaları üzerinde araştırma yapılması ilkesine dayanan literatürden farklı dört yeni yerel yöntem hem de özgün bir amaç fonksiyonunun tanımlandığı ve iteratif çözümlerin yapıldığı iki genel yöntem ile çözülmeye çalışılmıştır. Çalışmanın ilk bölümünde konu ile ilgili genel bilgilere yer verildi. İkinci kısım olan yapılan çalışmalar bölümünde, önerilen yerel yöntemler verildi ve özgün amaç fonksiyonunun tanımlandığı genel yöntemler kullanılarak, literatürdeki bilinen bir şeklin baskın noktaları bulundu. Son olarak tüm yöntemler, göreceliliğin en aza indirilmesi amacıyla üretilen bin adet yapay çokgen üzerinde ayrı ayrı test edildi. Benzetim ve başarı sonuçları karşılaştırmalı tablo halinde sunuldu.

**Anahtar Kelimeler:** Baskın Nokta Algılama, İkil Görüntü Çözümleme, Nesne Tanıma.

Master Thesis

SUMMARY

OPTIMIZATION ALGORITHMS FOR DETECTION OF DOMINANT POINTS OF  
OBJECTS IN BINARY IMAGES

Çiğdem GÜNGÖR

Karadeniz Technical University  
The Graduate School of Natural and Applied Sciences  
Statistic and Computer Science Graduate Program  
Supervisor: Ass. Prof. Dr. Orhan KESEMEN  
2012, 72 Pages

In object analyzing studies environmental points of the object provide a lot of useful information and describe the object itself without need for internal points. Dominant points of an object are one of the features that best describe it. The regions of dominant points are those where the most mobility among the boundary points of the objects that are two dimensional images of substances. They are also the characteristic regions that carry the most important information about the object. The subject of this research is the detection of dominant points of objects in digital images. The problem is solved with either 4 new local methods that based on the principle that researching on boundary pixels directly or 2 global methods that an original cost function is defined and iterative resolutions are performed for. In the first part of the study some significant principles and techniques about digital image processing are given. In the second part the proposed local methods are given and the dominant points of a well-known shape in the literature are detected by using the global methods. In addition some new algorithms about generating synthetic polygons that are necessary for applications and testing the methods are given. Finally all the methods are tested by using synthetic polygons which are generated for the purpose that trials can be done as objectively as possible. Simulations and success tests results are given as a cross-table.

**Key Words:** Dominant Point Detection, Binary Image Analysis, Object Recognition.

## ŞEKİLLER DİZİNİ

### Sayfa No

Şekil 1.1. Dijital fotoğraf makinesi ile elde edilmiş sayısal görüntü örneği. Görüntünün bir alan kesiti büyütülerek gözetleri belirginleştirilmiştir. Burada sayısal görüntünün ayrıklığı net biçimde görülmektedir. Her bir gözet farklı bir tamsayı değeri aldığından geçişler sürekli değil kesiklidir.....	5
Şekil 1.2. Farklı alanlarda kullanılan değişik türlerde sayısal görüntü örnekleri. (a) Uydu görüntüsü; (b) Barkod; (c) Astronomik cisim görüntüsü; (d) Parmak izi görüntüsü; (e) İllüstrasyon, grafiksel resimleme; (f) Bitki dokusunun mikroskop görüntüsü; (g) Dijital fotoğraf, tabiat resmi; (h) Röntgen görüntüsü; (i) Poster, grafiksel resimleme; (j) Yapay radar görüntüsü; (k) Yapay animasyon görüntüsü; (l) Yapay bakteri, biyolojik görüntü. ....	6
Şekil 1.3. Bir 8-bit boz ölçekli görüntü ve onun 256 yoğunluk değerinin sıklık dağılımını gösteren histogram. ....	8
Şekil 1.4. Sayısal görüntü koordinatları. Sayısal görüntü işlemede genel olarak orijinin sol üst köşede yer aldığı koordinat düzlemi kullanılır ( $m = 0, n = 0$ ). ( $m, n$ ) Koordinatları görüntüde sırasıyla sütunları ve satırları gösterir. $N \times M$ boyutlarında bir görüntü için en büyük sütun numarası $m_{maks} = M - 1$ ve en büyük satır numarası da $n_{maks} = N - 1$ 'dir.....	8
Şekil 1.5. 4-bit boz ölçek. $2^4 = 16$ farklı yoğunluk değeri içermektedir. ....	9
Şekil 1.6. Boz ölçekli görüntü. ....	10
Şekil 1.7. Şekil 1.1. 'deki kesit görüntüsünün boz ölçeklendirilmiş matrissel gösterimi. Burada matrisin boyutu $13 \times 14$ olup boz ölçekli görüntü olduğundan gözetler 0-255 arasında değerler alabilmektedir. En aydınlık nokta $maks(I(u, v)) = 222$ değerinde ve en karanlık nokta $min(I(u, v)) = 29$ değerindedir. ....	10
Şekil 1.8. İkili görüntü. ....	11
Şekil 1.9. Histogram eşitlemesi. (a) Özgün görüntü; (b) Özgün görüntünün histogramı; (c) Histogram eşitlemesi yapılmış görüntü; (d) Histogram eşitlemesi yapılmış görüntünün histogramı. ....	14
Şekil 1.10. Görüntü eşikleme. (a) Boz ölçekli görüntü; (b) Eşikleme işlemi sonucu oluşan ikili görüntü. Eşiklenmiş görüntü 0,75 eşik değeri kullanılarak elde edilmiştir. ....	15
Şekil 1.11. Gürültü temizleme için ortalama değer yöntemi. Solda giriş görüntüsü matrisinin seçilen $3 \times 3$ 'lük bölgesindeki değerlerin aritmetik ortalaması yaklaşık 103,55 ... şeklinde bulunup en yakın tam sayıya yuvarlanmıştır. ....	16
Şekil 1.12. Gürültü temizleme için çekirdek yöntemi. (a) Giriş görüntüsü matrisi; (b) çekirdek matrisi; (c) çıkış görüntüsü matrisi. Giriş görüntüsü matrisinin seçilen $3 \times 3$ 'lük bölgesi çekirdek matrisi ile birebir işleme sokularak elde edilen değerlerin ağırlıklı ortalaması yaklaşık 186,4 şeklinde bulunup en yakın tam sayıya yuvarlanmış ve çıkış matrisinde aynı gözet konumuna yeni değer olarak atanmıştır. ....	17

Şekil 1.13. Gürültü temizleme için ortanca yöntemi. (a) Giriş görüntüsü matrisi; (b) seçilen bölgedeki değerlerin sıralı biçimi; (c) çıkış görüntüsü matrisi. Giriş görüntüsü matrisinin seçilen $3 \times 3$ 'lük bölgesindeki gözek değerleri küçükten büyüğe sıralanmış ve tam ortadaki değer çıkış matrisinde aynı gözek konumuna yeni değer olarak atanmıştır. ....	18
Şekil 1.14. Açma ve kapama işlemlerinin uygulanacağı orijinal nesne (Gonzalez ve Woods, 2002). ....	19
Şekil 1.15. Açma işlemi. Sol üstte, $B$ yapı biriminin merkez noktasının $A$ nesnesinin farklı noktaları üzerindeki konumlanmaları; sağ üstte, $B$ yapı biriminin kullanılmasıyla yapılan aşındırma işlemi sonucu elde edilen $A \ominus B$ görüntüsü; sol altta $B$ yapı biriminin merkez noktasının $A \ominus B$ görüntüsünün etrafındaki konumlanmaları ve sağ altta $B$ yapı biriminin kullanılmasıyla yapılan genişletme işlemi sonucu elde edilen $A \circ B = (A \ominus B) \oplus B$ görüntüsü (Gonzalez ve Woods, 2002). ....	20
Şekil 1.16. Kapama işlemi. Sol üstte $B$ yapı biriminin merkez noktasının $A$ nesnesi etrafında gezindiği yörünge; sağ üstte $B$ yapı biriminin kullanılmasıyla yapılan genişletme işlemi sonucu elde edilen $A \oplus B$ görüntüsü; sol altta $B$ yapı biriminin merkez noktasının $A \oplus B$ nesnesinin çevresindeki konumlanmaları ve sağ altta $B$ yapı biriminin kullanılmasıyla yapılan aşındırma işlemi sonucu elde edilen $A \bullet B = (A \oplus B) \ominus B$ görüntüsü (Gonzalez ve Woods, 2002). ....	21
Şekil 1.17. Boşluklu nesne içeren ikil görüntü matrisi. Arka plan gözekleri ayrıklığın kolaylıkla seçilebilmesi için MsWord tasarım özelliğinin %25 daha açık siyahı ile gölgelendirilmiştir. ....	22
Şekil 1.18. İnsan sınır belirlemesi. 1000 görüntüyü aşan bir veri tabanından 4 ila 8 arasında nesneye bölünmüş görüntü örnekleri (Martin vd, 2004). Hatların kalınlığına göre çoğu insanın aynı konumları sınır olarak kabul ettiği yorumu çıkarılabilir. ....	23
Şekil 1.19. İkili görüntü içerisindeki bir nesne için sınır belirleme adımları. (a) Sayısal görüntü ve dikdörtgen içinde seçilmiş nesne; (b) büyütülmüş ve ayrıklaştırılmış görüntü ve sayısal görüntü matrisi; (c) sınırları belirlenmiş sayısal görüntü matrisi (Kesemen ve Güngör, 2011). ....	24
Şekil 1.20. Sıralı etiketlemenin ilk aşaması olan ön etiketlemenin adım adım gösterimi. ....	26
Şekil 1.21. Etiketleme işlemi tamamlanmış sayısal görüntü. ....	27
Şekil 1.22. Zincir kodları. ....	29
Şekil 1.23. Sayısal görüntü matrisinde noktalar arası uzaklık farkları. $A - B$ uzunluğu $A - C$ uzunluğundan daha kısadır. ....	29
Şekil 1.24. Sayısal görüntüdeki bir nesnenin gözek alanını hesaplayan algoritma. ....	30
Şekil 1.25. Sayısal görüntüdeki bir nesnenin kütle merkezini hesaplayan algoritma. ....	31
Şekil 1.26. Sayısal görüntüdeki bir nesnenin ana ekseninin koordinatlarını hesaplayan algoritma. ....	32
Şekil 2.1. Nesnelerin sınırlarındaki baskın noktalar ile gösterilmesi. (a) 21 adet nokta ile bir telefonun temsili; (b) 19 nokta ile bir elmanın temsili; (c) 19 nokta ile bir otomobilin temsili. ....	33

Şekil 2.2. Sayısal görüntünün ön işlem adımları. (a) Boz ölçekli özgün görüntü; (b) ters renkli görüntü; (c) önceki görüntünün art alan görüntüsü; (d) artalan çıkarılmış görüntü; (e) ayrımlılığı artırılmış görüntü; (f) eşiklenmiş görüntü; (g) boşlukları doldurulmuş ikil görüntü; (h) nesnelere etiketlenmiş görüntü; (i) nesnelere sınırları çıkartılmış görüntü. ....	36
Şekil 2.3. Arka planı çıkarılmış ve ayrımlılığı artırılmış görüntünün (a) logaritmik sıklık göstergesi (b) eşiklenmiş sıklık göstergesi .....	37
Şekil 2.4. İkili görüntü içerisindeki bir nesne için sınır belirleme adımları. (a) Sayısal görüntü ve dikdörtgen içinde seçilmiş nesne; (b) büyütülmüş ve ayrılaştırılmış görüntü ve sayısal görüntü matrisi; (c) sınırları belirlenmiş sayısal görüntü matrisi (Kesemen ve Güngör, 2011). ....	37
Şekil 2.5. Rastgele çokgen oluşturma algoritması. ....	39
Şekil 2.6. Sürekli doğrudan sayısal doğru parçası oluşturma. (a) sürekli doğru parçası; (b) sürekli doğrunun $\Delta$ aralığı ile elde edilmiş uzaysal örnekleme. ....	40
Şekil 2.7. Sayısal çokgen oluşturma, (a) sayısal doğrular yardımıyla oluşturulan kapalı eğri; (b) içi doldurulmuş çokgen nesne. ....	42
Şekil 2.8. Yapay çokgenin sınır gözleklerinin belirlenmesi, (a) üç köşeli sayısal görüntü; (b) sınır gözlekleri; (c) sınır gözleklerinin etiketlenmesi; (d) üç köşeli sayısal görüntü; (e) sınır gözlekleri; (f) sınır gözleklerinin etiketlenmesi.....	43
Şekil 2.9. Sınır gözleklerinin kütle-zemin oranlaması, (a) yapay üçgenin kenarından seçilmiş bir etkin gözeğin 3 komşuluğu; (b) yapay üçgenin köşesinden seçilmiş bir etkin gözeğin 3 komşuluğu; (c) yapay üçgenin tüm sınır gözlekleri için oranlama değerleri grafiği. (d) yapay dörtgenin kenarından seçilmiş bir etkin gözeğin 3 komşuluğu; (e) yapay dörtgenin köşesinden seçilmiş bir etkin gözeğin 3 komşuluğu; (f) yapay dörtgenin tüm sınır gözlekleri için oranlama değerleri grafiği. ....	46
Şekil 2.10. Sınır gözleklerinin göreceli uzaklık ölçütü, (a) yapay üçgenin kenarından seçilmiş bir etkin gözeğin ve zıt yönlerde $m$ gözeğin uzaklığında belirlenmiş iki gözeğin; (b) yapay üçgenin köşesinden seçilmiş bir etkin gözeğin ve zıt yönlerde $m$ gözeğin uzaklığında iki gözeğin; (c) yapay üçgene ait tüm etkin sınır gözlekleri için göreceli uzaklık değerleri grafiği; (d) yapay dörtgenin kenarından seçilmiş bir etkin gözeğin ve zıt yönlerde $m$ gözeğin uzaklığında iki gözeğin; (e) yapay dörtgenin köşesinden seçilmiş bir etkin gözeğin ve zıt yönlerde $m$ gözeğin uzaklığında iki gözeğin; (f) yapay dörtgene ait tüm etkin sınır gözlekleri için göreceli uzaklık değerleri grafiği. ....	48
Şekil 2.11. Etkin gözekten geçen iki bağlantım doğrusu arasındaki açının belirlenmesi, (a) yapay üçgene ait kenardan seçilmiş etkin gözeğin zıt yönlerinde $m$ tane gözeğin oluşturduğu iki değerler kümesi; (b) yapay üçgene ait köşeden seçilmiş etkin gözeğin zıt yönlerinde $m$ tane gözeğin oluşturduğu iki değer kümesi; (c) yapay üçgenin tüm etkin gözlekleri için bağlantım doğrularının kesişim açıları; (d) yapay dörtgene ait kenardan seçilmiş etkin gözeğin zıt yönlerinde $m$ tane gözeğin oluşturduğu iki değerler kümesi; (e) yapay dörtgene ait köşeden seçilmiş etkin gözeğin zıt yönlerinde $m$ tane gözeğin oluşturduğu iki değer kümesi; (f) yapay dörtgenin tüm etkin gözlekleri için bağlantım doğrularının kesişim açıları. ....	50

Şekil 2.12. Etkin gözek merkezli $2m$ sayıda gözeğin oluşturduğu kümenin belirtme katsayıları, (a) yapay üçgene ait kenardan seçilmiş etkin gözeğin zıt yönlerindeki $m$ tane gözeğin oluşturduğu değerler kümesi; (b) yapay üçgene ait köşeden seçilmiş etkin gözeğin zıt yönlerindeki $m$ tane gözeğin oluşturduğu değer kümesi; (c) yapay üçgenin tüm etkin gözlerleri için bağlantım doğrularından elde edilen $e_n$ değerlerinin grafiği; (d) yapay dörtgene ait kenardan seçilmiş etkin gözeğin zıt yönlerindeki $m$ tane gözeğin oluşturduğu değerler kümesi; (e) yapay dörtgene ait köşeden seçilmiş etkin gözeğin zıt yönlerindeki $m$ tane gözeğin oluşturduğu değer kümesi; (f) yapay dörtgenintüm etkin gözlerleri için bağlantım doğrularından elde edilen $e_n$ değerlerinin grafiği; .....	52
Şekil 2.13. İkili genetik algoritmanın akış şeması. ....	55
Şekil 2.14. Yapay dörtgen nesnesine genetik algoritma uygulaması, (a) birinci ebeveyn çözümü; (b) birinci ebeveyn çözümüne ilişkin $J$ görüntüsü; (c) maliyet fonksiyonunun değeri; (d) ikinci ebeveyn çözümü; (e) ikinci ebeveyn çözümüne ilişkin $J$ görüntüsü; (f) maliyet fonksiyonunun değeri; (g) birinci yavru çözümü; (h) birinci yavru çözümüne ilişkin $J$ görüntüsü; (i) maliyet fonksiyonunun değeri; (j) ikinci yavru çözümü; (k) ikinci yavru çözümüne ilişkin $J$ görüntüsü; (l) maliyet fonksiyonunun değeri.....	59
Şekil 2.15. Yapay dörtgen nesnesine tabu arama algoritmasının uygulaması, (a) başlangıç çözümü; (b) başlangıç çözümden elde edilen $J$ görüntüsü; (c) başlangıç çözüme ilişkin maliyet değeri; (d) 2. denemede çözümden elde edilen $J$ görüntüsü; (e) 2. denemede çözüme ilişkin maliyet değeri; (g) 4. denemede çözümden elde edilen $J$ görüntüsü; (h) 4. denemede çözüme ilişkin maliyet değeri.....	62
Şekil 3.1. Kromozom nesnesine genetik algoritma uygulaması, (a) 10. iterasyondaki çözüm; (b) 10. iterasyona ilişkin $J$ görüntüsü; (c) 10. iterasyona ilişkin maliyet değeri; (d) 15. iterasyondaki çözüm; (e) 15. iterasyona ilişkin $J$ görüntüsü; (f) 15. iterasyona ilişkin maliyet değeri; (g) 20. iterasyondaki çözüm; (h) 20. iterasyona ilişkin $J$ görüntüsü; (i) 20. iterasyona ilişkin maliyet değeri. ....	64
Şekil 3.2. Kromozom nesnesine tabu arama algoritma uygulaması, (a) 10. denemede çözümden elde edilen $J$ görüntüsü; (b) 10. denemede çözüme ilişkin maliyet değeri; (c) 20. denemede çözümden elde edilen $J$ görüntüsü; (d) 20. Denemede çözüme ilişkin maliyet değeri; (e) En iyi çözüm olan 12. Denemede çözümden elde edilen $J$ görüntüsü; (f) 12. Denemede çözüme ilişkin maliyet değeri; (g) En iyi çözüm olan 12. Denemede çözümden elde edilen $J$ görüntüsü; (h) 12. Denemede çözüme ilişkin maliyet değeri. ....	65

## TABLULAR DİZİNİ

	<b><u>Sayfa No</u></b>
Tablo 1.1. Temel görüntü türlerine ait bit derinlikleri ve tipik uygulama alanları .....	12
Tablo 3.1. Baskın nokta algılama ve araştırma yöntemlerinin başarı değerleri.....	66

## SEMBOLLER DİZİNİ

DIP	: Sayısal Görüntü işleme (Digital Image Processing)
AI	: Yapay Zeka (Artificial Intelligence)
RGB	: Kırmızı Yeşil Mavi renk uzayı (Red Green Blue colour space)
$I(u, v)$	: Sayısal görüntünün $(u, v)$ konumundaki gözetik yoğunluk değeri
$\hat{I}(u, v)$	: İşlenmiş sayısal görüntünün $(u, v)$ konumundaki yoğunluk değeri
◦	: Morfolojik Açma Operatörü (Morphological Opening Operator)
•	: Morfolojik Kapama Operatörü (Morphological Closing Operator)
$\oplus$	: Aşındırma Operatörü (Erosion Operator)
$\ominus$	: Genişletme Operatörü (Dilation Operator)
$\cap$	: Kesişim operatörü
$\approx$	: Yaklaşık olarak eşit
$\Delta x$	: Yatay yöndeki değişim miktarı
$\Delta y$	: Dikey yöndeki değişim miktarı
$\Delta$	: Genel çözünürlük miktarı
$d(A, B)$	: $A$ ve $B$ noktaları arasındaki uzaklık
$J(u, v)$	: Belirlenen baskın noktalara ilişkin yaklaşık çokgen görüntüsü
$E(I, J)$	: Belirlenen baskın noktalara ilişkin maliyet fonksiyonu
$F_{uyum}$	: Belirlenen baskın noktalara ilişkin uyum değeri
$e_n$	: Önerilen yerel yöntemler için anlamlılık değeri
$s_n$	: Kütle-Zemin Oranı yöntemi için hesaplanan kütleyle ait gözetiklerin sayısı
$d_n$	: Göreceli Uzaklık yöntemi için hesaplanan uzaklık değeri
$\theta_n$	: Kesişim Açısı yöntemi için hesaplanan açı değeri
$r_n^2$	: Belirtme Katsayısı yöntemi için hesaplanan belirtme katsayısı
GA	: Genetik Algoritma
$b_{p_i}$	: GA'da kromozoma ait her bir genin değeri
$s$	: GA'da çaprazlama noktası
TA	: Tabu Arama



## 1. GENEL BİLGİLER

Teknolojinin hızla geliştiđi günümüzde insanođlunun varmak istediđi nokta, hangi alanda olursa olsun en az kaynak tüketimi, en çok enerji ve zaman tasarrufuyla en yüksek verimi elde etmektir. Bu amaçlarla bilimsel ve teknolojik gelişmeler her gün bir önceki güne göre daha fazla yol kat etmektedir. 1960'lı yıllardan itibaren hızla gelişen bilgisayar bilimleri uygulamaları zaman içerisinde insanların vazgeçilmezleri haline gelmiştir. Bilgisayarların kullanımının yaygınlaşmaya başlamasından günümüze kadar, veri işleme ve saklama işi kağıtların dolaplarda tutulmasıyla değil, terabaytlar cinsinden ifade edilen sayısal bilgilerin yedekleme disklerinde tutulmasıyla yapılmaktadır. Gerektiğinde yetkili kullanıcı veriyi isteđi doğrultusunda kullanıp bir takım yararlı bilgiler üretmekte ve bu üretilen yeni veriyi de yine sayısal ortamda saklamaktadır.

Zaman ilerledikçe gelişen teknolojiyle birlikte meslekler, mesleklerin yapılma şekilleri, ihtiyaçlar, hatta zevk ve eğlence anlayışı değişmekte ve bu durum değişen ihtiyaçlara cevap üretebilen yeni icatların, yazılımların insanlığın hizmetine sunulmasını beraberinde getirmektedir. Tüm bu gelişimin sonucu olarak insanođlunun bilgisayardan beklentileri gün geçtikçe artmaktadır. Günümüzde insanođlu kendi yerine okuyabilen bilgisayarlar, kendi kendine park edebilen arabalar, ortam sıcaklığına göre kendi iklimlendirme ayarlarını yapabilen klimalar, kendi yerine savaşıyan uçaklar ve gözetleme yapacak elektronik gözler istemektedir. Kısaca insanođlu kendi yerine düşünüp karar verebilen makineler istemiştir. İşte tüm bu istekleri ve hayalleri insanı Yapay Zeka (Artificial Intelligence - AI) kavramına kadar getirmiştir. Yapay Zeka, bilgisayar programlarının akıllı davranışları taklit etmesi üzerinden insan zekasının doğasını anlamaya çalışan bir disiplindir (Searle, 1996). Tanımdan da anlaşıldığı gibi Yapay Zeka insan zekasının bir taklitler zinciri olarak karşımıza çıkmaktadır. Makine de tıpkı insan gibi herhangi bir sayısal ortamdan çıkardığı bilgiyi faydalı ya da faydasız diye sınıflandırmakta ve faydalı bilgiyi ayırıştırıp işlemektedir.

Bilim dünyası yaşama yeni değerler katarken genellikle doğadan esin almıştır. Yapay Zeka'nın gelişme süreci de bu yolla olmuştur. İnsan beyninin halen tam anlamıyla keşfedilememiş olan mucizevi işlevleri Yapay Zeka'nın birbirinden farklı alt disiplinlerinin gelişmesi için esin kaynağı olmuştur. İnsanođlunun, beyninin yapısal ve işlevsel özellikleri bakımından, beyni olan diğer canlılardan farklı olduğu açıkça ortadadır. Tarihin ilk

çağlarından beri yaşamdaki canlı çeşitliliği içinde insanın şu anki yerini alması akıl yürütme yeteneği sayesinde olmuştur. Akıl yürütmesi insanın içinde bulunduğu problemleri çözmesini ve hayatını kolaylaştırmasını sağlamış, bu yeteneği insanı diğer canlılara üstün kılmıştır. Bununla birlikte insanın düşünsel işlevlerini kullanabilmesi için işlenebilir bilgiye ihtiyacı vardır.

İnsanın zihinsel yetileri ne kadar yüksek olursa olsun düşünme etkinliğinin ham maddesi olan bilgiye erişemedikçe beynini etkin bir biçimde kullanması mümkün değildir. İhtiyacı olan bu bilgileri ise çevresi ile iletişim kurabilmesine olanak veren beş duyusu aracılığıyla sağlar. Yaşadığı tecrübeler sonucunda oluşan bilgiyi bu beş duyusu aracılığıyla edinir. Aynı zamanda insanın, dış dünyada gerçekleşen, kendisine doğrudan ya da dolaylı olarak etki eden fiziksel ve kimyasal olayları kontrol altında tutabilmesini sağlayan yetilerinin neredeyse tamamı beş duyusu ile kazandıklarıdır. Bu duyulardan kuşkusuz en önemlisi görme duyusudur. Herhangi bir konuda karar almak için bilgi ve ölçütlere ihtiyaç duyulur. Yararlı bilginin büyük bir kısmı ise görme yeteneği ile karşılanır. Örneğin insan araba kullanırken, markete giderken, en basit şekliyle evde bir odadan diğerine geçerken etraftaki cisimleri algılayıp doğru yöne doğru adımı atmasında en kaliteli ve en hızlı bilgiyi çevresini görerek edinir.

İnsan zekasının doğal yapısına benzer şekilde yapay zekanın da en temel alt dallarından biri görüntü işlemedir. Görüntü işleme endüstride pek çok alanda yaygın olarak kullanılmaktadır ve uygulama alanlarının çeşitliliği sayesinde farklı bilim çevrelerince benimsenmiştir. Görüntü işlemeye ilişkin literatürü incelediğimizde ilgili alanlarda oluşan ihtiyaçlar doğrultusunda pek çok fonksiyon ve tekniğin geliştirilmiş olduğunu görürüz.

İçinde bulunduğumuz modern çağda görsel bilgi aktarımı iletişimdeki ana yöntem olarak kabul görmektedir. Bilgisayarlar tarafından görsel bilginin işlenmesi esasına dayanan *Sayısal Görüntü İşleme* disiplinler arası bir araştırma alanı olarak geliştirilmektedir. Görsel bilginin diğer tüm özellikleri arasında, görüntü içerisindeki nesnelerin *şekilleri* özel bir rol üstlenmektedir. Çözümleme ve sınıflandırma için bir araç teşkil eden nesnenin şekil temasının önemi, bu konudaki çalışmaların bilimin ve mühendisliğin çok sayıda farklı alanına yayılarak artmasından görülmektedir (Marji ve Siy, 2003).

Görüntü içerisinde birbirine bitişik olan ve zemin renginden farklı, aynı renkteki gözeklerin (pixel) oluşturduğu kümelere nesne veya tanecik denir. Görüntü içerisinde bulunan değişik biçimlerde ve boyutlardaki nesnelerin özelliklerinin araştırılması

uygulama alanlarında oldukça önemli bir yer tutmaktadır. Bu uygulama alanları: tıpta birçok hastalığın teşhisi; askeri alanlarda uydu görüntüleriyle istihbaratların derlenmesi, akıllı füzelerin hedeflerini bulması; bitki çeşitliliğinin araştırılması, hayvan kitlelerinin göçlerinde ve nüfuslarının incelenmesi; kayaçların mineral içeriklerinin araştırılması, vb. konular olarak sayılabilir (Kesemen vd, 2010).

Şekil çözümlemesi hesaplamalarında önemli adımlardan biri şekil gösterimidir öyle ki bu gösterim ele alınan şekli çözümleme için uygun forma getirme sürecini de kapsar. Önemli bir takım şekil çözümleme teknikleri ise görüntü içerisindeki nesnelerin dış hatlarının veya silüetinin gösterimini temel almaktadır. Doğal olarak sağladığı faydalı bilgilerden dolayı şekillerin dış hatları en yaygın olarak kullanılan şekil tanımlayıcılarından biri olup iki boyutlu nesnelerin bu özelliklerinin temsili için birçok yöntem geliştirilmiştir ve geliştirilmektedir. Çok başarılı olsun ya da olmasın tüm bu yöntemlerin genel amacı nesneyi en iyi temsil edebilecek özellikleri tespit etmektir (Costa ve Cesar, 2001).

Bir nesnenin en belirgin özelliği, nesnenin kenarlarındaki ani değişimin olduğu bölgelerin uçlarında bulunan merkezi noktalardır. Bu noktalar nesnenin baskın noktaları olarak tanımlanır. Sayısal görüntülerdeki nesneler, değişik geometrik şekillere sahip olabilmektedirler. Bu nesnelerin baskın noktaları yardımıyla çokgenlere benzetilip kenar sayılarının ve dolayısıyla köşe noktalarının belirlenmesi ve nesnelerin daha az veriyle temsil edilmesi uygulamada önemli problemlere çözüm olabilmektedir. Karmaşık biçimlerdeki nesnelerin daha basit bir çerçeveye temsil edilmesi görüntü tanıma yöntemlerinde kolaylık sağlamaktadır (Kesemen ve Güngör, 2011).

Baskın noktalar görüntü işleme uygulamalarında önemli bir yere sahiptir. Baskın noktalar, nesnenin sınırlarını oluşturan noktalar arasında en kritik bilgiyi taşıyan noktalardır. Buna bağlı olarak baskın noktalar büyük miktardaki ham veri yığnında ciddi oranda gereksiz verinin indirgenmesi ile nesne algılama algoritmalarının çok daha etkin ve hızlı çalışmasında birincil paya sahiptir. Nesneyi temsil etmesi açısından baskın nokta yaklaşımının gelişimi Psikoloji bilimindeki, şekillerle ilgili sayısız çalışmanın hızlı ve erken sonuçlanmasını sağlayarak o alanda yürütülen araştırmalara bir çatı oluşturmuştur (Attneave, 1954).

### 1.1. Sayısal Görüntü İşlemede Temel Kavram ve Teknikler

Görüntüleme kullanılan bir kameranın görüntü düzlemine düşen ışın, temelde iki boyutlu uzayda, zamana bağlı sürekli dağılımlı ışık enerjisi demetidir. Bu sürekli olarak değişen ışık dağılımının anlık görüntüsünü elde etmek için üç temel adım gereklidir. İlk adım uzaysal örnekleme adımıdır.

Bir görüntünün uzaysal örnekleme fiziksel görüntüden yansıyan sürekli ışığın ayrık düzlemde gösterilmesidir. İkinci adım olan zamansal örnekleme her bir sensör elemanı üzerine düşen ışık miktarının düzenli aralıklarla ölçülmesi ile gerçekleştirilir. Gözet değerlerinin oluşturulması adımı olan son adımında, ölçülen ışık miktarı görüntü değerlerini bilgisayarlarda saklamak ve işlemek için bir dizi tamsayı değerine dönüştürülür (Örneğin  $2^8 = 256$  ayrınç ya da  $2^{12} = 4096$  ayrınç). Tıbbi görüntüleme gibi profesyonel uygulamalarda ondalıklı ölçek kullanılmaktadır. Dönüştürme işlemi adımı olan son adımda doğrudan sensör elemanlarına yerleştirilen bir dönüştürücü ya da özel arayüz donanımları kullanılarak analog sinyal dijital sinyale dönüştürülür. Bu şekilde görüntü kullanılmaya ve saklanmaya hazır hale getirilmiş olur (Burge ve Burger, 2009).

Bu üç aşama sonucu, bir tamsayılar matrisinden oluşan sayısal görüntü tanımlanır (Şekil 1.1). Matematiksel olarak,

$$I(u, v) \in \mathbb{P}, \quad u, v \in \mathbb{N} \quad (1)$$

şeklinde gösterilir. Burada  $I$  sayısal görüntüsü  $\mathbb{P}$  olası gözet değerleri kümesinde tanımlı,  $\mathbb{N}^2$ 'nin tam sayı koordinatlarında iki boyutlu bir fonksiyondur (Burge ve Burger, 2009).

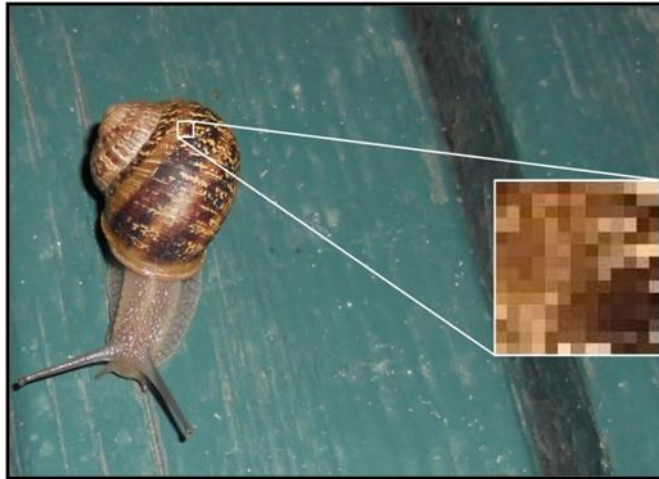
Sayısal görüntü işleme sürekli bir görüntü alanını onun bir eşdeğer sayısal durumuna dönüştürülmesi esasına dayanır. Sayısal görüntü işleme sistemleri genellikle bir fiziksel görüntünün ayrık örnekleme noktalarından elde edilen sayı dizileriyle ilgilenir. Yapılan işlemler sonucunda bir başka sayılar dizisi oluşturulur ve bu sayılar görüntüyü yeniden yapılandırmada kullanılır (Pratt, 2007).

Yukarıdaki ifadelerden yola çıkarak,  $\mathbb{N}$  Doğal Sayılar Kümesi ve  $U, V \in \mathbb{N}$  olmak üzere (2)'deki gösterimi elde ederiz.

$$I(u, v) = \begin{bmatrix} I(0,0) & I(0,1) & \dots & I(0, V-1) \\ I(1,0) & I(1,1) & \dots & I(1, V-1) \\ \vdots & \vdots & \ddots & \vdots \\ I(U-1,0) & I(U-1,1) & \dots & I(U-1, V-1) \end{bmatrix} \quad (2)$$

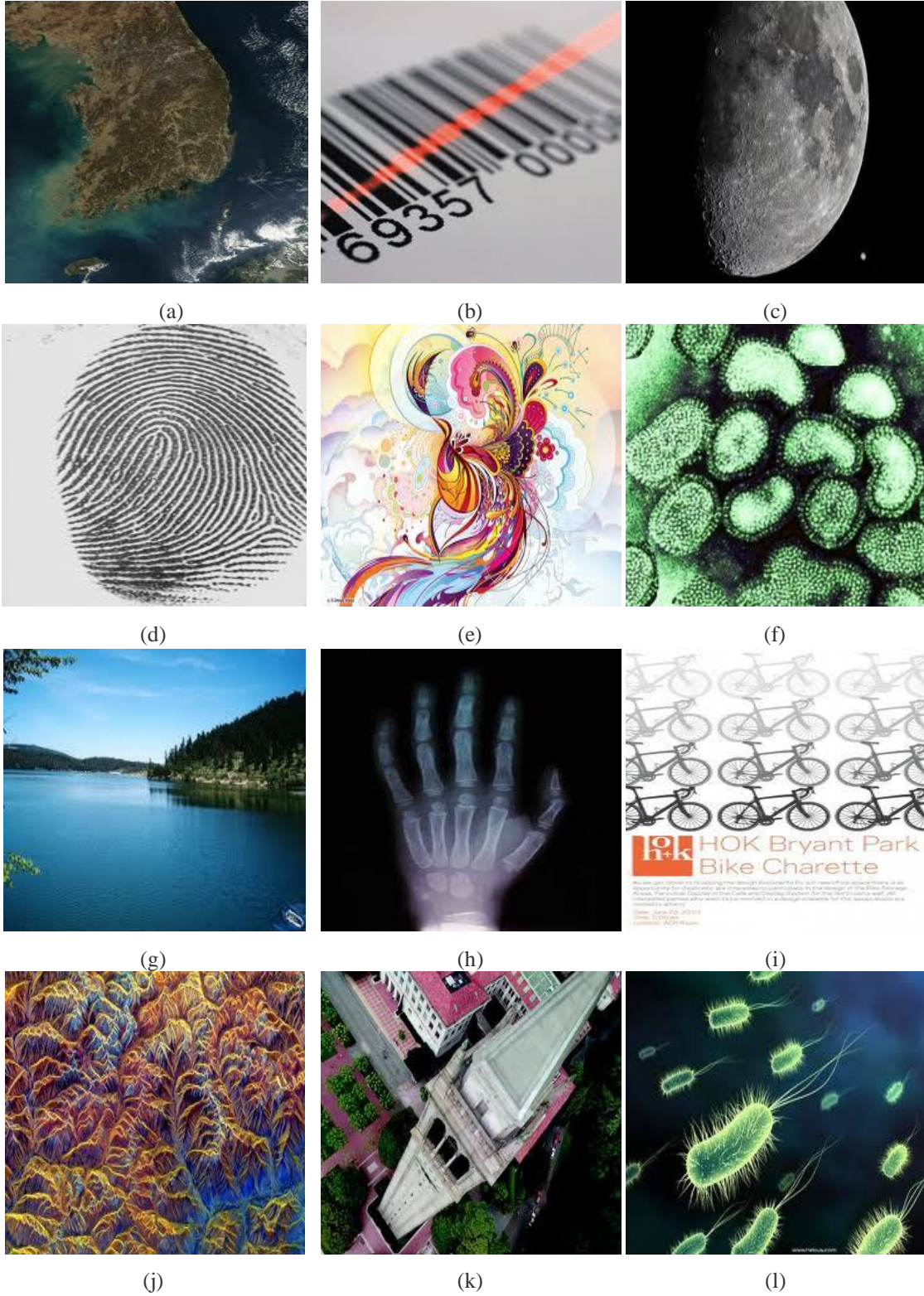
Eşitliğin solundaki  $I$  sayısal görüntüsünü tanımlayan matrisin her bir elemanı görüntünün gözet değerlerini vermektedir.

Aşağıda Şekil 1.1 ile sürekli bir görüntünün ayırık örnekleme olan bir sayısal görüntü ve görüntünün küçük bir kesiti büyütülerek gösterilmiştir. Görüntünün elde edildiği zaman anındaki ışık miktarına bağlı olarak, sayısal görüntüye ait gözet değerleri oluşturulmuştur. Her bir gözette farklı bir renk değeri tutulmaktadır. Kesitin boyutu gözetlerin gözle görülür şekilde büyütüldüğünden görüntünün ilgili koordinatlarına denk düşen renklerin ayrınçları net biçimde farkedilebilmektedir.



Şekil 1.1. Dijital fotoğraf makinesi ile elde edilmiş sayısal görüntü örneği. Görüntünün bir alan kesiti büyütülerek gözetleri belirginleştirilmiştir. Burada sayısal görüntünün ayrıklığı net biçimde görülmektedir. Her bir gözet farklı bir tamsayı değeri aldığından geçişler sürekli değil kesiklidir.

Sayısal görüntüler yaşamın kolaylaşması ve güzelleşmesi için endüstrinin, sanatın, bilmin ve teknolojinin vs. değişik alanlarında sıklıkla kullanılmaktadır ve günlük hayatta önemli bir yere sahiptir. Bu duruma birçok farklı örnek mevcuttur. Uydu görüntüleri (Şekil 1.2 (a)) yeryüzündeki jeofiziksel ve jeolojik olayların incelenmesi ve daha çoğu farklı amaç için gerekli olan bilginin edinilmesinde yaygın olarak kullanılmaktadır. Canlı dokularının mikroskopik görüntüleri (Şekil 1.2 (f)) sayesinde çeşitli incelemeler yapılabilmekte ve bu incelemeler sonucu elde edilen bilgiler birçok bilimsel araştırmaya ışık tutmaktadır.



Şekil 1.2. Farklı alanlarda kullanılan değişik türlerde sayısal görüntü örnekleri. (a) Uydu görüntüsü; (b) Barkod; (c) Astronomik cisim görüntüsü; (d) Parmak izi görüntüsü; (e) İllüstrasyon, grafiksel resimleme; (f) Bitki dokusunun mikroskop görüntüsü; (g) Dijital fotoğraf, tabiat resmi; (h) Röntgen görüntüsü; (i) Poster, grafiksel resimleme; (j) Yapay radar görüntüsü; (k) Yapay animasyon görüntüsü; (l) Yapay bakteri, biyolojik görüntü.

Dijital fotoğraflar (Şekil 1.2(g)) günlük hayatın vazgeçilmezleri haline gelmiştir. Bugün üretilen cep telefonlarının hemen hemen hepsinde yüksek kalitede resim çekebilme özelliği mevcut bulunmaktadır. Röntgen görüntüleri (Şekil 1.2 (h)) incelenerek vücudun hastalıklı veya hasarlı dokuları tespit edilebilmektedir. Poster ve illüstrasyon (Şekil 1.2 (e),(i)) gibi grafiksel resimleme görüntüleri reklam endüstrisinde ve sanatsal çalışmalarda yaygın olarak kullanılmaktadır. Barkod görüntülerinin (Şekil 1.2 (b)) kullanıldığı sistemler sayesinde ürün sayısının ve çeşitliliğinin fazla olduğu mağazalarda stok kontrolü oldukça kolay hale gelmiştir. Astronomik cisimlerin görüntülerinden (Şekil 1.2 (c)) uzay araştırmaları ile ilgili sayısız yararlı bilgiye ulaşılabilmekte ve yapay radar görüntüleri (Şekil 1.2 (j)) kullanılarak jeofiziksel araştırmalar için benzetimler yapılabilmektedir. (d) Parmak izi görüntüleri ile kriminal bulgulara ulaşılmakta ve bu görüntüler genel olarak güvenlik sektöründe de kullanılmaktadır (Şekil 1.2 (d)). Eğlence sektörünün elektronik ayağı sayılan bilgisayar oyunları için ise yapay animasyon görüntüleri (Şekil 1.2 (k)) vazgeçilmez görsel öğelerdendir.

### 1.1.1. Sayısal Görüntünün Histogramı

Histogramlar genel anlamda sıklık dağılımlarıdır. Görüntünün histogramı bir görüntü içindeki görüntü birimlerinin aldığı yoğunluk değerlerinin sıklığını belirtir (Şekil 1.3).  $[0, L]$  aralığında  $I(u, v)$  yoğunluk değerlerini alan  $I$  boz ölçekli görüntüsü için bir  $h$  histogramı  $L$  tane sınıf aralık değeri içerir. Genel bir 8-bit boz ölçekli görüntü için,  $L = 2^8 = 256$ 'dır.

Histogramdaki her bir sıklık değeri matematiksel olarak,

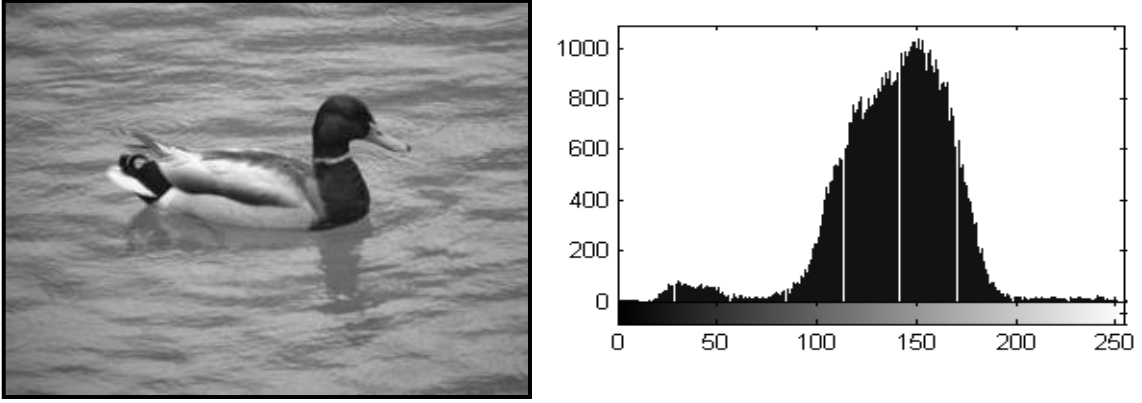
$$h(r) = \sum_{u=0}^U \sum_{v=0}^V \delta(I(u, v) - r), \quad r = 0, 1, 2, \dots, L - 1 \quad (3)$$

şeklinde hesaplanır (Kesemen, 2011). Burada,

$$\delta(t) = \begin{cases} 1, & t = 0 \\ 0, & t \neq 0 \end{cases} \quad (4)$$

biçimde olan direkt delta işlevidir.

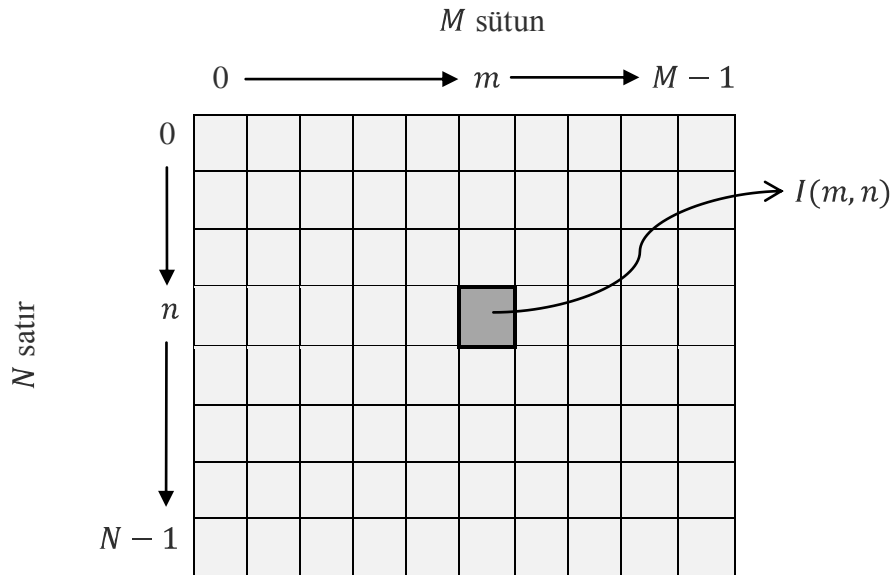




Şekil 1.3. Bir 8-bit boz ölçekli görüntü ve onun 256 yoğunluk değerinin sıklık dağılımını gösteren histogram.

### 1.1.2. Sayısal Görüntülerde Koordinat Sistemi

Bir görüntü matrisinin ( $I$ ) büyüklüğü, sütun sayısı (genişlik,  $M$ ) ve satır sayısı (yükseklik,  $N$ ) ile belirlenir. Görüntü üzerindeki bir noktanın hangi görüntü birimine karşılık geleceğini bilmek için noktaları bir koordinat sistemine (Şekil 1.4) oturtmak gerekir.



Şekil 1.4. Sayısal görüntü koordinatları. Sayısal görüntü işlemede genel olarak orijinin sol üst köşede yer aldığı koordinat düzlemi kullanılır ( $m = 0, n = 0$ ).  $(m, n)$  Koordinatları görüntüde sırasıyla sütunları ve satırları gösterir.  $N \times M$  boyutlarında bir görüntü için en büyük sütun numarası  $m_{maks} = M - 1$  ve en büyük satır numarası da  $n_{maks} = N - 1$  'dir.



Matematik bilgilerinden hatırlanan koordinat sisteminin normal gösteriminin aksine görüntü işlemede koordinat sistemi, koordinat düzleminin yatay eksen etrafında çevrilmesiyle orijini sol üst köşede konumlanmış,  $y$ -ekseninin yukarıdan aşağıya doğru arttığı bir sistem olur (Burge ve Burger, 2009).

### 1.1.3. Sayısal Görüntünün Gözet Değerleri

Bir görüntü birimi ile ilgili bilgi, o görüntü biriminin temsil edildiği veri türüne bağlıdır. Gözet değerleri uygulamada her zaman  $k$  uzunluğunda ( $k$ -karakterden oluşan) ikil kelimelerdir (1 ve 0'lardan oluşan dizgiler) öyle ki  $2^k$  farklı değerden herhangi biri ile temsil edilebilirler. Burada  $k$  değerine görüntünün bit derinliği ya da sadece derinliği denir. Belirli bir gözetin kesin bit seviyesi düzeni görüntünün türüne (örneğin ikil, boz ölçekli veya RGB renkli) göre değişir. (Burge ve Burger, 2009).

### 1.1.4. Boz Ölçekli Görüntüler (Yoğunluk Görüntüleri)

Bir gözetin yoğunluğu, belirli bir aralık içerisinde en küçük ve en büyük (dahil) değer arasında ifade edilir. Bu aralık soyut bir şekilde 0 (siyah) ve 1 (beyaz) arasındaki rasyonel sayı değerleri ile temsil edilir. Bilgi işlem alanında boz ayrıncı yapılan görüntü işlemleri rasyonel sayılar üzerinden yapılsa da görüntü birimleri bellekte ikil ifadeler halinde saklanır. Genel bir ifade ile  $k$ -bit görüntüler için mümkün yoğunluk değerlerinin aralığı  $[0 \dots 2^k - 1]$  dir diyebiliriz.

Bazı erken dönem boz ayrıncı gösteren monitörler sadece 16 (4-bit) farklı ayrıncı gösterebilirdi (Şekil 1.5).



Şekil 1.5. 4-bit boz ölçek.  $2^4 = 16$  farklı yoğunluk değeri içermektedir.

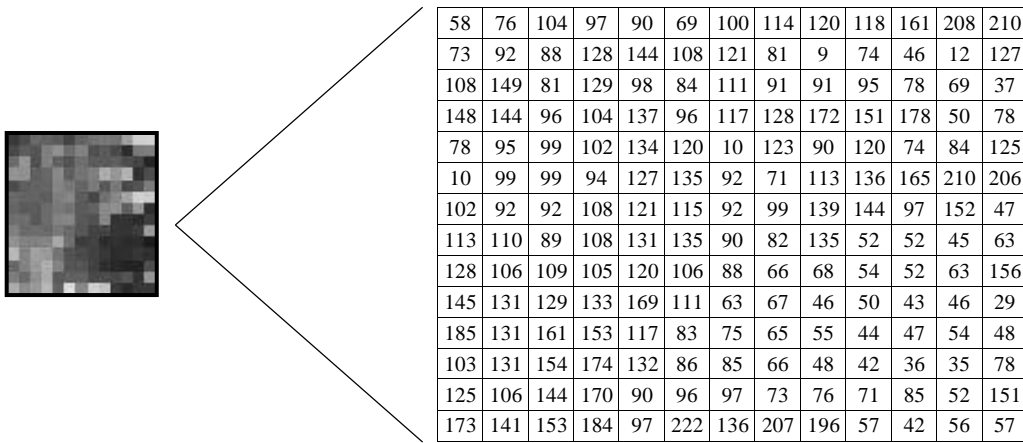
Ancak bugün fotoğraf gibi görsel ekran (ekran üzerinde ve basılı) için tasarlanmış boz ölçeklemeli görüntüler yaygın olarak örneklenmiş gözet başına 8 bit ile saklanır. Bu da boz ayrıncıları 256 farklı yoğunlukta kaydedebilmemize olanak sağlar.

Şekil 1.6 ile boz ayrıncılı görüntü verilmiştir. Görüntü gözekleri renk değeri içermeyip her biri yalnızca yoğunluk değerine sahiptir ve 256 farklı yoğunluk değerinden birini almaktadır (Rosenfeld, 1969).



Şekil 1.6. Boz ölçekli görüntü.

Boz ölçekli görüntüler tek bir gözeğin tek bir bayt yer tutması sebebiyle programlama için oldukça uygundur. Tıbbi görüntüleme ya da uzaktan algılama uygulamaları gibi teknik kullanımlar sensör hassasiyetinden tam olarak faydalanmak için ve hesaplamaları yuvarlama hatalarına karşı korumak için genellikle daha ileri seviye (örneğin gözek başına 10 veya 12 bit) gerektirmektedir. Şekil 1.7 ile Şekil 1.1 deki kesit görüntüsünün yoğunluk değerleri gösterilmektedir. Gözeklerin aldığı yoğunluk değerleri ise sağdaki matriste verilmiştir.



Şekil 1.7. Şekil 1.1. 'deki kesit görüntüsünün boz ölçeklendirilmiş dizeysel gösterimi. Burada matrisin boyutu  $13 \times 14$  olup boz ölçekli görüntü olduğundan gözekler 0-255 arasında değerler alabilmektedir. En aydınlık nokta  $\max\{I(u, v)\} = 222$  değerinde ve en karanlık nokta  $\min\{I(u, v)\} = 29$  değerindedir.

### 1.1.5. İkili Görüntüler

İkili görüntüler en basit ve yerine göre en kullanışlı görüntü türleridir. Bir ikili görüntü görüntüyü oluşturan her bir gözeğin 0 (siyah) ya da 1 (beyaz) değerlerinden sadece birini aldığı görüntüdür. Genellikle ikili görüntülerin iki unsurdan oluştuğunu kabul edebiliriz: ön planı oluşturan nesne(ler) ve arka plan. Burada nesnelerin 1 ve arka planın 0 değerini alması yaygın bir kabuldür. Bunun böyle olmasını gerektiren sezgisel ya da rasyonel birçok geçerli sebep vardır. Bunlardan birinin aritmetik işlem kolaylığı sağlaması olduğu söylenebilir. İkili görüntüler doğal olarak birçok farklı görüntü işleme probleminde ortaya çıkarlar. Bazı görüntü türleri özünde ikili görüntü olduğundan doğrudan işlemlerde kullanılabilirler. Bir tarayıcı yardımı ile ya da bir optik kalem ile elde edilebilen, tek renkten oluşan düzgün bir arka plan üzerindeki ikili grafik çizimleri ve baskı karakterleri, parmak izleri, barkod görüntüleri, taranmış el yazısı görüntüleri vs. örnek olarak gösterilebilir.



Şekil 1.8. İkili görüntü

Bunun yanında alternatif olarak renkli veya boz ölçekli görüntülere örneğin eşikleme ve bölütleme gibi belirli görüntü işleme tekniklerinin uygulanması sonucunda da ikili görüntüler elde edilir (Costa ve Cesar, 2001).

İkili görüntüler özellikle bu çalışmada çok fazla önem taşımaktadır, çünkü görüntü içindeki nesnelerin baskın noktaları araştırılırken genellikle bütünüyle nesnelere veya sadece onların dış hatlarını oluşturan gözetimlerden yararlanılmış ve bellekte gereksiz yer işgal edeceği için diğer bilgilere ihtiyaç duyulmamıştır. Günlük hayatta yaygın kullanılan bazı görüntü türlerinin özellikleri Tablo 1.1’de özetlenmiştir.

Tablo 1.1. Temel görüntü türlerine ait bit derinlikleri ve tipik uygulama alanları.

Gri ölçekli Görüntüler (Yoğunluk Görüntüleri):

Kanal	Bit/Gözetek	Değer Aralığı	Kullanım Alanı
1	1	0,1	İkil görüntü: belge, grafik resimleme, fax
1	8	0 ... 255	Genel: fotoğraf, tarama, yazdırma
1	12	0 ... 4095	Yüksek kalite: fotoğraf, tarama, yazdırma
1	14	0 ... 16383	Profesyonel: fotoğraf, tarama, yazdırma
1	16	0 ... 65535	En yüksek kalite: tıp, astronomi

Renkli Görüntüler:

Kanal	Bit/Gözetek	Değer Aralığı	Kullanım Alanı
3	24	[0 ... 255] <sup>3</sup>	RGB, Genel: fotoğraf, tarama, yazdırma
3	36	[0 ... 4095] <sup>3</sup>	RGB, Yüksek kalite: fotoğraf, tarama, yazdırma
3	42	[0 ... 16383] <sup>3</sup>	RGB, Profesyonel: fotoğraf, tarama, yazdırma
4	32	[0 ... 255] <sup>4</sup>	CMYK, Dijital baskı öncesi

Özel Görüntüler:

Kanal	Bit/Gözetek	Değer Aralığı	Kullanım Alanı
1	16	-32768 ... 32767	Poz./Neg. Tamsayı değerleri: süzgeçler, duvaklar
1	32	$\pm 3.4 \times 10^{38}$	Ondalıklı değerler: tıp, astronomi
1	64	$\pm 1.8 \times 10^{308}$	Ondalıklı değerler: profesyonel ses verileri analizi

### 1.1.6. Renkli Görüntüler

RGB renk modeli ile temsil edilen görüntüler her bir ana renk (kırmızı, yeşil, mavi) için üç resim bileşeninden oluşurlar. RGB uzayında her bir gözeği temsil etmek için kullanılan bit sayısına gözetek derinliği denilmektedir. Bu koşullar altında RGB renk uzayında kırmızı, yeşil ve mavi görüntülerin her biri için gözetek derinliği 8 bit olduğundan RGB renkli gözeteklerin 24 bit derinliğinde olduğu söylenir. Bir 24-bit RGB renkli görüntüyü adlandırmak için sıklıkla tam renkli görüntü ifadesi de kullanılır. Bu durumda bir 24-bit RGB renkli görüntüdeki toplam renk sayısı  $(2^8)^3 = 16,777,216$  (Gonzalez ve Woods, 2002).

### **1.1.7. Özel Görüntüler**

Görüntü değerlerini temsil etmek için Tablo 1.1'deki standart biçimlerin hiçbiri yeterli olmadığında özel görüntüler için tanımlı biçimlerin kullanılması gerekir. Özel görüntüler için sık rastlanan iki örnek, negatif değerli görüntüler ve ondalıklı değerli görüntülerdir. Negatif değerli görüntüler kenar algılama süzgeci gibi görüntü işleme tekniklerinin kullanımında, ondalıklı değerli görüntüler ise sıklıkla daha geniş sayısal aralık ve hassasiyet gerektiren tıbbi, biyolojik ve astronomik uygulamalarda kullanılır. Bu özel biçimler çoğunlukla uzmanlık uygulamalarında kullanılmakta olup, standart görüntü işleme araçlarında yer almamaktadır (Burge ve Burger, 2009).

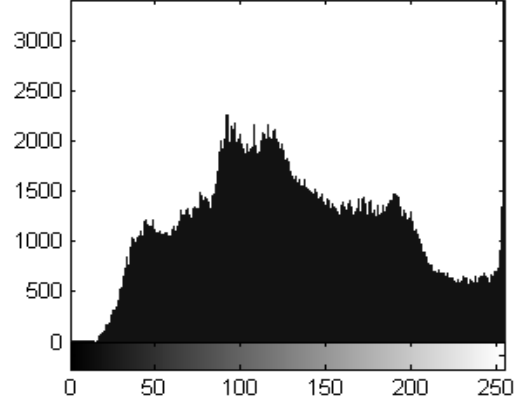
### **1.1.8. Histogram Eşitleme (Histogram Equalization)**

Herhangi bir görüntünün histogramından yola çıkarak o görüntü ile ilgili birçok istatistiki bilgiye ulaşmak mümkündür, fakat histogramdan görüntünün yoğunluk değerleri ile ilgili yargılara varmak en yaygın kullanımlardan biridir. Örneğin karanlık bir görüntüde gözek değerleri, 0'a yakın bölgede kümelenecek, aydınlık bir görüntüde 1'e yakın bölgede kümelenecek, uygun karşıtlık özelliğine sahip bir görüntüde ise yoğunluk değerleri iyi ve dengeli bir yayılım gösterecektir (Şekil 1.9).

Literatürde histogram düzeltilmesi yapabilen farklı yöntemler mevcuttur fakat birçoğu parametre girişi gerektirdiğinden en yaygın kullanılan yöntemlerden biri histogram eşitlemedir. (Gonzalez vd, 2004). Böylece görüntünün yoğunluk değerleri düzeltilmiş olur. Orijinal görüntüye uygulanan histogram eşitleme işlemi sonucunda ayrımlılığı artırılmış bir görüntü elde edilir.



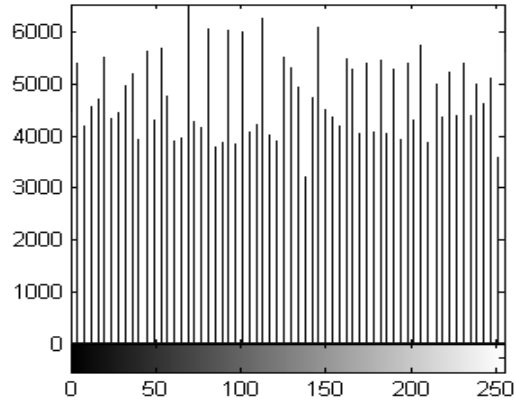
(a)



(b)



(c)



(d)

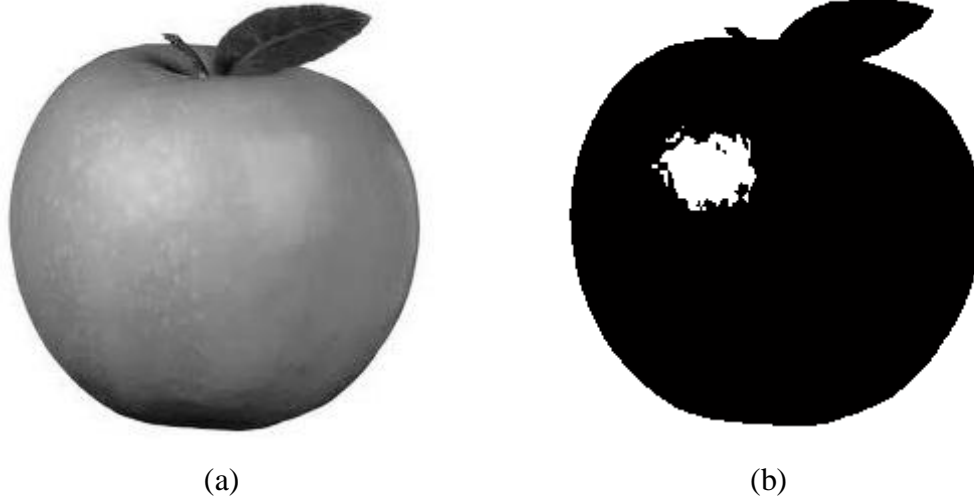
Şekil 1.9. Histogram eşitlemesi. (a) Özgün görüntü; (b) Özgün görüntünün histogramı; (c) Histogram eşitlemesi yapılmış görüntü; (d) Histogram eşitlemesi yapılmış görüntünün histogramı.

### 1.1.9. Görüntü Eşikleme (Tresholding)

Eşikleme, boz ölçekli bir görüntüyü yalnızca siyah ve beyaz gözek değerleri içeren ikil görüntüye dönüştürme sürecidir (Thai, 2003). İkil görüntü elde etmek için boz ölçekli görüntünün her bir görüntü birimine bir eşik değeri uygulanır. Eğer gözek değeri eşik değerinden yüksekse yeni gözek değeri 255 (beyaz) olarak ayarlanır. Aksi takdirde gözeğin değeri 0 (siyah) olur (Davies, 2003). İkil görüntülerde genellikle siyah değerli gözekler arka plana, beyaz gözekler ise ön plana karşılık gelir. Eğer eşikleme işlevi sonucu

Şekil 1.10'daki gibi nesneye ait gözetler 0 değeri almış iseler görüntünün tersi alınarak ön plan gözetleri 1 arka plan gözetleri ise 0 değerli duruma getirilir.

Eşik değerini görüntünün yoğunluk histogramını inceleyerek de belirlemek mümkündür. Çünkü ön planı oluşturan nesnelere gözet yoğunluğu arka plan gözetlerinin yoğunluğundan farklıdır. Bu durumda iki farklı dağılımdan diğerine göre basık olmayan değerler kümesinin ön plan nesnelere gözetlerine ait olduğu beklenir dolayısıyla ön planı arka plandan ayırmak için eşik değeri iki kümelene bölgesinin arası olarak belirlenir (Fisher vd, 2000).



Şekil 1.10. Görüntü eşikleme. (a) Boz ölçekli görüntü; (b) Eşikleme işlemi sonucu oluşan ikil görüntü. Eşiklenmiş görüntü 0,75 eşik değeri kullanılarak elde edilmiştir.

### 1.1.10. Gürültü Temizleme (Filtering Noise)

Gürültüyü görüntü sinyalindeki dış etkilerin sebep olduğu herhangi bir bozulma olarak tanımlamak mümkündür. Bir görüntü elektronik bir ortamdan başka bir ortama taşınırken görüntü sinyalinde birtakım hataların oluşması gözlenebilir bir durumdur. Bu hatalar çıkış görüntüsüne sinyallerdeki kötü etkinin türüne göre farklı şekillerde yansacaktır. Genellikle hatanın türü ve buna bağlı olarak görüntüdeki gürültü tahmin edilir, dolayısıyla da kötü etkiyi azaltmak için en uygun yöntem uygulanır. Gürültüye maruz kalmış bir görüntüyü temizleme süreci görüntü işlemede çok önemli bir adımı oluşturmaktadır. Nitekim tıbbi görüntü analizleri gibi ayrıntılı görüntü çözümlerinde

en küçük bir veri, işlem sonuçlarını önemli düzeyde etkileyebilmektedir. Gürültü azaltma yöntemlerinin tamamı şu iki varsayımdan yola çıkmaktadır (Arce vd, 2000): ‘Görüntünün en küçük birimleri olan gözekler önemli tüm detaylardan daha küçüktürler’ ve ‘mevcut birçok gözek, komşularıyla aynı yapıyı gösterir’. Bu varsayımlardan yola çıkılarak birbirinden farklı birçok algoritma geliştirilmiştir.

### 1.1.11. Kayan Ortalama Süzgeci

Ortalama yaklaşımı için en basit yöntem belirli bir alandaki tüm gözek değerlerinin ortalaması alınarak bu ortalama değerlerden yeni görüntüyü oluşturmaktır (Şekil 1.11). Bu yönteme ortalama değer (mean) adı verilir (Buzuloiu vd, 2001). Bu işlem sonucunda daha az yoğunluk değeri içeren bir çıkış görüntüsü üretileceği için bu yöntem görüntünün kalitesinde ciddi bir azalmaya ve önemli ayrıntıların yok olmasına sebep olmaktadır.

58	76	104	97	90	69	100	58	76	104	97	90	69	100
73	92	88	128	144	108	131	73	92	88	128	144	108	131
108	149	81	129	98	84	111	108	149	81	129	98	84	111
148	144	96	104	137	96	117	148	144	96	104	137	96	117
78	95	99	102	134	120	101	78	95	104	102	134	120	101
106	99	99	94	127	135	92	106	99	99	94	127	135	92

Şekil 1.11. Gürültü temizleme için ortalama değer yöntemi. Solda giriş görüntüsü matrisinin seçilen  $3 \times 3$ 'lük bölgesindeki değerlerin aritmetik ortalaması yaklaşık 103,55 ... şeklinde bulunup en yakın tam sayıya yuvarlanmıştır.

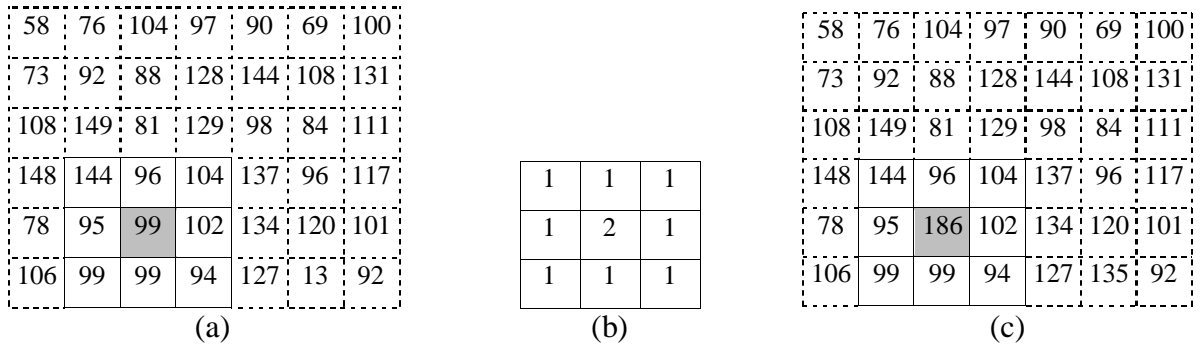
Yukarıdaki şemada aritmetik ortalama yönteminin uygulanma prensibi görülmektedir. Buna göre  $3 \times 3$ 'lük olarak belirlenen alan içerisindeki tüm elemanların aritmetik ortalaması alınarak çıkış matrisinde aynı gözek konumuna yeni değer olarak atanır.

Sıklıkla uygulanan bir diğer yöntem çekirdek yöntemidir. Buna göre, giriş görüntü matrisindeki her bir gözeğin, kendisiyle birlikte belirli bir çerçevedeki gözeklerin değerlerinin ağırlıklı ortalamasının alınması sonucunda elde edilen değer, çıkış matrisinin aynı konumdaki gözeğine değer olarak atanmasıyla yeni bir görüntü



oluşturmaktır (Şekil 1.12) (Davies, 1988). Bu yöntemde kullanılan çekirdekler sadece  $n \times n$  boyutunda tanımlanabilmektedir.

Çekirdek merkezindeki gözek etkin gözek ile çakışacak şekilde, çekirdek üzerindeki her bir ağırlık katsayısının bire bir çakıştığı gözek değeriyle çarpılmasıyla elde edilen değerlerin toplamının ağırlıklar toplamına bölünmesiyle oluşan değer kullanılır. Burada etkin gözeğin kenara ve köşelere yakın olduğu durumda sanal gözeklere ihtiyaç duyulmaktadır. Sanal gözeklere atanacak değerler için simetrik alma, sıfırlama, sarmal değer atama gibi yaklaşımlar kullanılabilir.



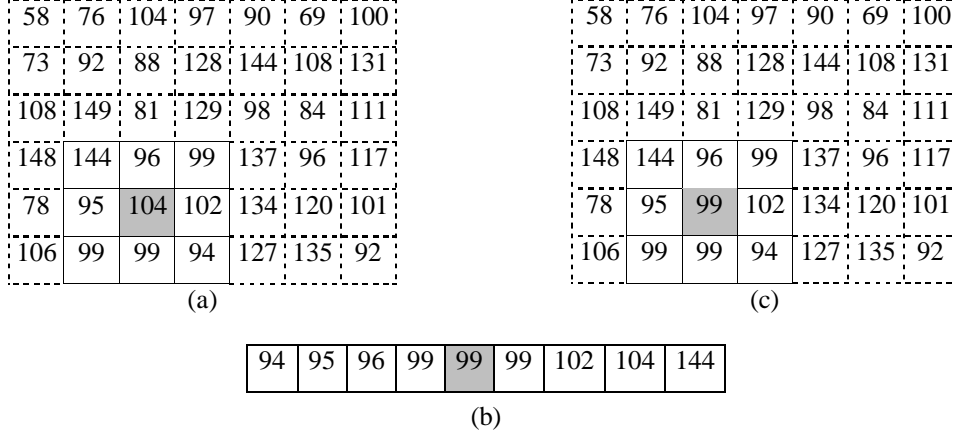
Şekil 1.12. Gürültü temizleme için çekirdek yöntemi. (a) Giriş görüntüsü matrisi; (b) çekirdek matrisi; (c) çıkış görüntüsü matrisi. Giriş görüntüsü matrisinin seçilen  $3 \times 3$ 'lük bölgesi çekirdek matrisi ile birebir işleme sokularak elde edilen değerlerin ağırlıklı ortalaması yaklaşık 186,4 şeklinde bulunup en yakın tam sayıya yuvarlanmış ve çıkış matrisinde aynı gözek konumuna yeni değer olarak atanmıştır.

### 1.1.12. Sıralı Süzgeçler

Daha önce de belirtildiği gibi ortalamaya dayalı yöntemler görüntü üzerindeki nesnelerin kenarlarında bulanıklaşmaya ve ayrıntıların yok olması gibi istenmeyen sonuçlara sebep olmaktadır. Literatürde gürültü temizleme problemi için ortalamaya dayalı yöntemler yerine orijinal görüntüdeki tüm verilerin işleme sokulmadığı parametrik olmayan yöntemler de kullanılmaktadır (Huang, 1979). Bu yöntemlerden biri ortanca (median) yöntemidir.

Bu yöntem etkin gözek dahil komşu gözek değerlerinin sıralanması ve ortadaki değer çekirdeğin merkezinde bulunan etkin gözeğe yeni değer olarak atanması esasına dayanmaktadır (Şekil 1.13). Burada yeni görüntü oluşturulurken gözek değerleri aritmetik işleme tabi tutulmadığından yalnızca orijinal görüntünün gözek değerleri kullanılır.

Böylece yeni değer, komşu gözeteklerin sahip olabileceği çok küçük veya çok büyük sayısal değerlerden doğrudan etkilenmeyeceğinden hata oranının da azalması sağlanacaktır. Dolayısıyla doğal olarak yöntem, cihazlardan kaynaklanan, gözetek kayıpları vb gibi hataların oluşturduğu belirli tip gürültülerin giderilmesi için çok kullanışlıdır.



Şekil 1.13. Gürültü temizleme için ortanca yöntemi. (a) Giriş görüntüsü matrisi; (b) seçilen bölgedeki değerlerin sıralı biçimi; (c) çıkış görüntüsü matrisi. Giriş görüntüsü matrisinin seçilen  $3 \times 3$ 'lük bölgesindeki gözetek değerleri küçükten büyüğe sıralanmış ve tam ortadaki değer çıkış matrisinde aynı gözetek konumuna yeni değer olarak atanmıştır.

Ortanca yönteminin Ortalama değer yöntemine göre üstün özellikleri genel olarak şöyledir:

- Ortanca yöntemi yeni ve aritmetik bir değer yerine komşu gözeteklerin mevcut değerlerini kullandığından geçişler arası değer aralıkları küçülmez.
- Ortanca yönteminde Ortalama değer yönteminde görülen, komşu değerlerin birbirlerine karşı büyüklüklerinden kaynaklanan, kenarların ötelenmesi durumu görülmez.

Bu sebeplerden dolayı yöntem görsel araştırmalarda tercih edilen bir yöntemdir (Huang, 1979).

### 1.1.13. Oyük Açma-Kapama (Opening-Closing)

Genişletme (dilation) işlevi ikil görüntülerdeki nesnelere kalınlaştıran ve büyüten bir işlemdir. Bu kalınlaştırma ve büyüme işlemi yapı birimi (structure element) denilen, boyutları ve biçimi önceden tanımlanmış bir şekil ile kontrol edilir. Yapı birimi esasında 1'lerden ve 0'lardan oluşan matrislerdir. Matematiksel olarak  $A$ 'nın  $B$  ile genişletilmesi  $A \oplus B$  ile gösterilir ve aşağıdaki şekilde tanımlanır (Gonzalez vd, 2004),

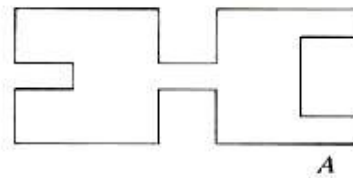
$$A \oplus B = \{z | (\hat{B})_z \cap A \neq \emptyset\}$$

Burada  $\emptyset$  boş küme ve  $B$  yapı birimidir. Sözel olarak ifade etmek gerekirse  $A$ 'nın  $B$  ile genişletilmesi,  $B$  yapı biriminin merkez noktasının gezindiği görüntü konumları ikil görüntüdeki nesneye ait sınır gözeteklerinin konumları ile örtüştüğünde nesnenin, yapı elementinin yarıçapı kadar büyümesidir.

Aşındırma (erosion) işlevi ikil görüntülerdeki nesnelere inceltme ve küçültme bir işlemdir. Bu inceltme ve küçültme işlemi genişletmede olduğu gibi yapı birimi ile kontrol edilir. Matematiksel olarak  $A$ 'nın  $B$  ile aşındırılması  $A \ominus B$  ile gösterilir ve aşağıdaki şekilde tanımlanır (Gonzalez vd, 2004),

$$A \ominus B = \{z | B_z \cap A^c \neq \emptyset\}$$

Yani  $A$ 'nın  $B$  ile aşındırılması,  $B$  yapı elementinin merkez noktasının gezindiği görüntü konumları ikil görüntüdeki  $A$  nesnesine (Şekil 1.14) ait sınır gözeteklerinin konumları ile örtüştüğünde, nesnenin yapı elementinin yarıçapı kadar küçülmesidir.



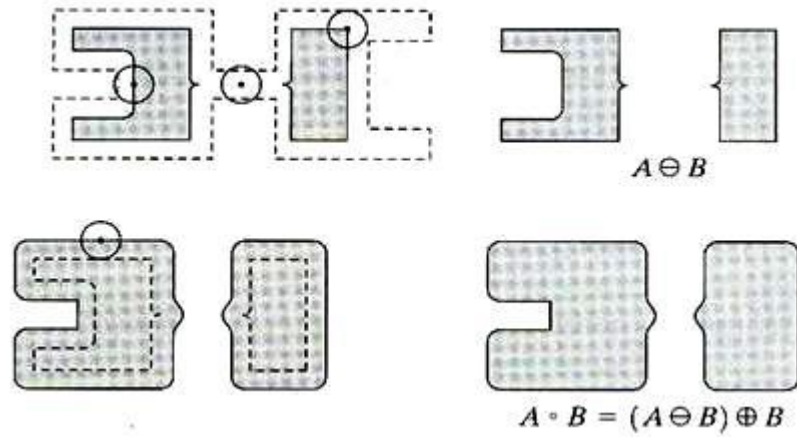
Şekil 1.14. Açma ve kapama işlemlerinin uygulanacağı orijinal nesne (Gonzalez ve Woods, 2002).

Aşındırma işlemi çok küçük nesnelere görüntüden gidermede faydalı bir işlemdir. Ne var ki bu işlem önemli bazı ayrıntıların da yok olmasına sebep olan bir işlemdir. Bu tür kötü etkiyi gidermek için genellikle aşındırma işleminin ardından nesneye aynı yapı birimi ile genişletme işlemi uygulanır. Açma işlemi bu iki işlemin bileşkesi niteliğinde olduğundan bu iki işlemi sırasıyla uygulamak yerine tercih edilmektedir. Açma işlemi herhangi bir noktada tamamen yapı birimini içeren tüm nesnelere elekten geçirme durumu söz konusudur, ancak nesne boyutundaki genel daralmayı da önler. Bu işlem ayrıca yapı biriminin çapından daha küçük kalınlıktaki çizgilerin silinmesi problemi için idealdir (Jahne, 2002).  $A$ , görüntü içindeki birbirine bağlı gözetler kümesi,  $B$ , açma işleminde kullanılan yapı birimi olsun.  $A$  kümesinin açılması işlemi  $A \circ B$  şeklinde gösterilir aşağıdaki gibi tanımlanır,

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

Yani  $A$ 'nın  $B$  tarafından açılması işlemi,  $A$ 'nın  $B$  tarafından aşındırılması, sonrasında tekrar  $B$  ile genişletilmesi şeklinde ifade edilebilir (Gonzalez ve Woods, 2002).

Şekil 1.15'te  $A \circ B$  işleminin şematik gösterimi verilmiştir.



Şekil 1.15. Açma işlemi. Sol üstte,  $B$  yapı biriminin merkez noktasının  $A$  nesnesinin farklı noktaları üzerindeki konumlanmaları; sağ üstte,  $B$  yapı biriminin kullanılmasıyla yapılan aşındırma işlemi sonucu elde edilen  $A \ominus B$  görüntüsü; sol altta  $B$  yapı biriminin merkez noktasının  $A \ominus B$  görüntüsünün etrafındaki konumlanmaları ve sağ altta  $B$  yapı biriminin kullanılmasıyla yapılan genişletme işlemi sonucu elde edilen  $A \circ B = (A \ominus B) \oplus B$  görüntüsü (Gonzalez ve Woods, 2002).

Burada adım adım önce  $A$ 'nın  $B$  yapı birimi ile aşındırılması daha sonra da aşınmış görüntünün genişletilmesi sonucu oluşan açma işlemi gösterilmektedir.

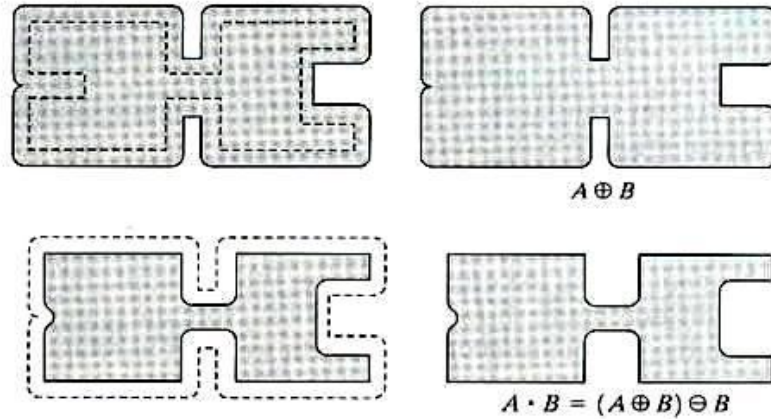
Diğer taraftan genişletme işlemi nesnelere alansal olarak genişleterek küçük boşlukları doldurup nesne içindeki oyukları kapatırken, onun ardından uygulanan aşındırma işlemiyle yapı biriminin boyutuna ve şekline bağlı olarak meydana gelen, nesnelere genel büyümenin etkisi azaltılır. Açma işleminde olduğu gibi bu iki işlemin (bu sıradaki) uygulamasının yerine genellikle ikisinin birleşimi niteliği gösteren kapama işlemi tercih edilmektedir (Jahne, 2002).

Aynı şekilde bir  $A$  kümesinin  $B$  yapı birimi ile kapanması işlemi  $A \bullet B$  şeklinde gösterilir aşağıdaki gibi tanımlanır,

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

Yani sözel olarak yukarıdaki ifade  $A$ 'nın  $B$  ile kapanması işlemi,  $A$ 'nın  $B$  tarafından genişletilmesi, sonrasında oluşan görüntünün tekrar  $B$  ile aşındırılması şeklinde açıklanır (Gonzalez ve Woods, 2002).

Şekil 1.16'te  $A \bullet B$  işleminin şematik gösterimi verilmiştir (Gonzalez ve Woods, 2002). Burada adım adım önce  $A$ 'nın  $B$  yapı birimi ile genişletmesi daha sonra da genişletilmiş görüntünün aşındırılması sonucu oluşan kapama işlemi gösterilmektedir.

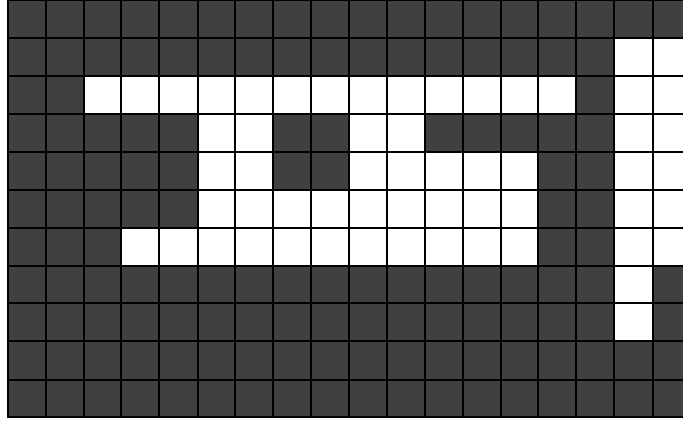


Şekil 1.16. Kapama işlemi. Sol üstte  $B$  yapı biriminin merkez noktasının  $A$  nesnesi etrafında gezindiği yörünge; sağ üstte  $B$  yapı biriminin kullanılmasıyla yapılan genişletme işlemi sonucu elde edilen  $A \oplus B$  görüntüsü; sol altta  $B$  yapı biriminin merkez noktasının  $A \oplus B$  nesnesinin çevresindeki konumlanmaları ve sağ altta  $B$  yapı biriminin kullanılmasıyla yapılan aşındırma işlemi sonucu elde edilen  $A \bullet B = (A \oplus B) \ominus B$  görüntüsü

Açma ve Kapama işlemleri sonucunda görüntü içindeki nesnelerin dış hatlarının köşe bölgelerinde yumuşamalar görülür.

#### 1.1.14. Etiketleme ile Oyuk Kapama

Genişletme (dilation) ve kapama (closing) işlemleri nesnelerin fazla büyümesine dolayısıyla ayrı nesnelerin birbirine bağlanmasına ve dış hatlarında istenmeyen yumuşamalara sebep olabilmektedir. Bu tür durumların gözlenmeyeceği alternatif bir diğer boşluk doldurma işlemi ise etiketleme tekniği kullanılarak yapılabilir. Diğerlerine benzer şekilde burada da yöntemin yalnızca ikil görüntülerde kullanılabildiğini vurgulamak gerekir. Boşluk doldurma işleminin yapılacağı ikil görüntüde arka plan gözeklerinin 0, ön plan yani nesne gözeklerinin 1 değerinde olması gerekir. Şekil 1.17 ile etiketleme ile oyuk doldurma algoritmasının işleyişine örnek olacak ikil görüntü matrisi verilmiştir.



Şekil 1.17. Boşluklu nesne içeren ikil görüntü matrisi. Arka plan gözekleri ayrıklığın kolaylıkla seçilebilmesi için MsWord tasarım özelliğinin %25 daha açık siyahı ile gölgelendirilmiştir.

Burada ilgilenilen yani kontrol edilen gözekler 0 değerli gözeklerdir. İstenilen görüntü elde edildiğinde uygulanacak adımlar şöyledir:

**Adım 1.** Görüntü sol üstten başlanarak taranır. Eğer etkin gözek kenar gözeği ise  $-1$  ile etiketlenir.

**Adım 2.** Eğer etkin gözek  $-1$  değerli bir gözek ile sağdan, soldan veya üstten komşu ise arka plan ile ilişkilidir (bağlıdır) demektir ve  $-1$  ile etiketlenir.

**Adım 3.** Eğer etkin gözek, solundaki ve üstündeki gözleklerden her ikisi de 1 değerli ise (yani nesneye ait ise)  $-2$  ile etiketlenir.

**Adım 4.** Etkin gözeğin solundaki gözeğin değeri  $-2$ , üstündeki gözeğin değeri 1 ise etkin gözek nesne içindeki boşlukla ilişkili olabilir demektir ve  $-2$  ile etiketlenir.

**Adım 5.** Eğer etkin gözeğin (örneğin Şekil 1.17'deki  $I(4,16)$ ) solundaki gözek (Şekil 1.17'deki  $I(4,15)$ ) boşluk gibi algılanıp  $-2$  ile etiketlenmiş ise ve aynı zamanda etkin gözek  $-1$  değerli bir gözek (Şekil 1.17'deki  $I(3,16)$ ) ile komşu ise Adım 2'ye gidilir.

Tüm görüntü tarandıktan ve etiketleme işlemleri bittikten sonra  $-1$  değerli arka plan gözleklerine 0 ve  $-2$  değerli nesne içindeki boşluk gözleklerine 1 değeri atanarak etiketleme ile boşluk doldurma işlemi tamamlanmış olur (Soille, 1999).

### 1.1.15. Sınır Algılama (Edge Detection)

Nesneler içindeki karakteristik noktalar 2 boyutlu nesnelerin eşleştirilmeleri işlemlerinde görüntü konumlarının doğru olarak bulunması için yararlı olsa da sınır noktaları çoğu zaman nesne için daha fazla önem taşıyan anlamlı yapılar olmuşlardır. İnsan zihninde ilk çağrışım yapan şekliyle nesnelerin sınırları, genellikle gözle görülür dış hatlar biçiminde tarif edilmektedir (Şekil 1.18).



Şekil 1.18. İnsan sınır belirlemesi. 1000 görüntüyü aşan bir veri tabanından 4 ila 8 arasında nesneye bölünmüş görüntü örnekleri (Martin vd, 2004). Hatların kalınlığına göre çoğu insanın aynı konumları sınır olarak kabul ettiği yorumu çıkarılabilir.

Diğer tür sınırlar nesnenin yüzey yönelimlerinin hızlı değişimler gösterdiği gölge sınırlar veya buruşma ve kırışıklık sınırlarıdır.

Daha genel ifade edilmesi gerekirse kenarlar, farklı renk, yoğunluk veya doku bölgeleri arasında oluşum gösteren yapılardır denilebilir. Fakat sınırların belirlenmesi için sıklıkla tüm bu bilgiler yerine sadece yerel bilgilerin kullanımı tercih edilmektedir (Szeliski, 2011).

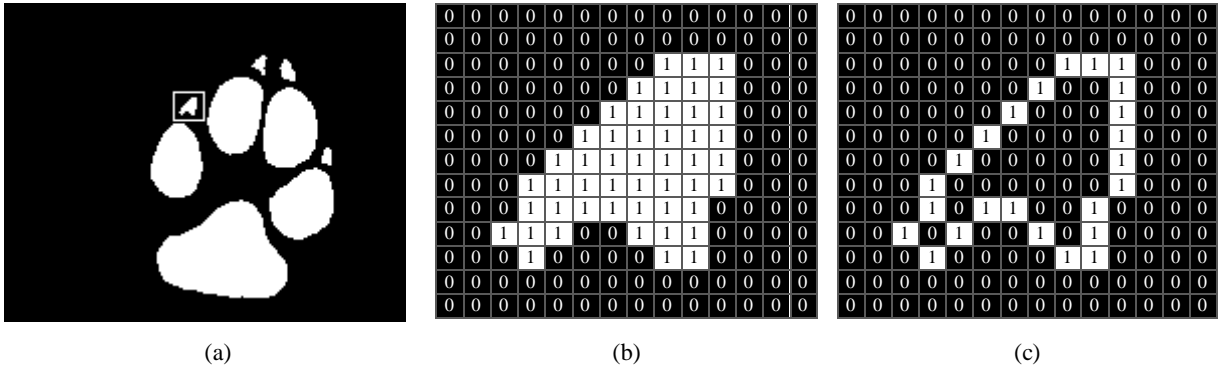
Bu koşullar göz önünde bulundurulmuş ve bu çalışmada da nesnelerin sınırları belirlenirken renkli görüntülerdeki renk değişimlerini esas almak yerine boz ölçekli ve ikil görüntülerdeki hızlı yoğunluk değişimi esas alınmıştır. Buna göre her nesne ayrı ayrı ele alınarak, aşağıdaki adımlar uygulanır ve nesnenin sınır gözekleri bulunur (Kesemen ve Güngör, 2011),

**Adım 1.** Boz ölçekli görüntüdeki gözekler sol üst köşeden başlanarak, ilk nesneye ait ilk gözektan, son nesneye ait son gözeğe kadar denetlenir.

**Adım 2.** Gözeklerin 4-komşuluklarında herhangi bir hücrede 0 değeri mevcut ise o gözek sınır gözeği olarak kabul edilir ve yeni değeri 1 olarak atanır.

**Adım 3.** Denetlenen gözeğin 4-komşuluğundaki herhangi bir gözeğin değeri ait olduğu nesnenin yoğunluk değerine eşit ise etkin gözeğin yeni değeri 0 olarak atanır.

Şekil 1.19’de (a) ikil görüntüdeki nesnelardan biri kesit olarak alınıp büyütülmüş, (b) görüntü matrisi biçiminde gösterilmiş ve yukarıdaki algoritmanın işleyişi sonucu nesnenin (c) sınır gözekleri matrisi elde edilmiştir.



Şekil 1.19. İkili görüntü içerisindeki bir nesne için sınır belirleme adımları. (a) Sayısal görüntü ve dikkörtgen içinde seçilmiş nesne; (b) büyütülmüş ve ayrıklaştırılmış görüntü ve sayısal görüntü matrisi; (c) sınırları belirlenmiş sayısal görüntü matrisi (Kesemen ve Güngör, 2011).



## 1.2. Nesne Analizi ve Baskın Nokta Algılama

Bu bölümde ele alınacak şekiller, fotoğraf makinelerinden, tarayıcılardan ve benzeri elektronik cihazlardan alınan sayısal görüntülerdeki nesnelere aittir. Ayrık bir şekil, ortogonal kafeste temsil edilen noktaların oluşturduğu sonlu elemanlı kümeyle karşılık gelir. Çoğu ayrık şekil önceden tanımlanmış bir uzaysal örnekleme şemasına göre bazı sürekli şekillerin örnekleme sonucunu olarak elde edilir. (Costa ve Cesar, 2001). Sayısal görüntülerdeki nesnelere iyi bir çözümlemesinin yapılabilmesi için onların tam olarak ne gibi özellikler gösterdikleri bilinmelidir. Bu bölümde şekillerin belli başlı karakteristik özelliklerine değinilse de asıl üzerinde durulan, şekilleri çokgensel yaklaşımlarla temsil etmeye yarayan ve en önemli özelliklerinden biri olan baskın noktalarıdır.

## 1.3. Nesnenin Şekil Kavramı

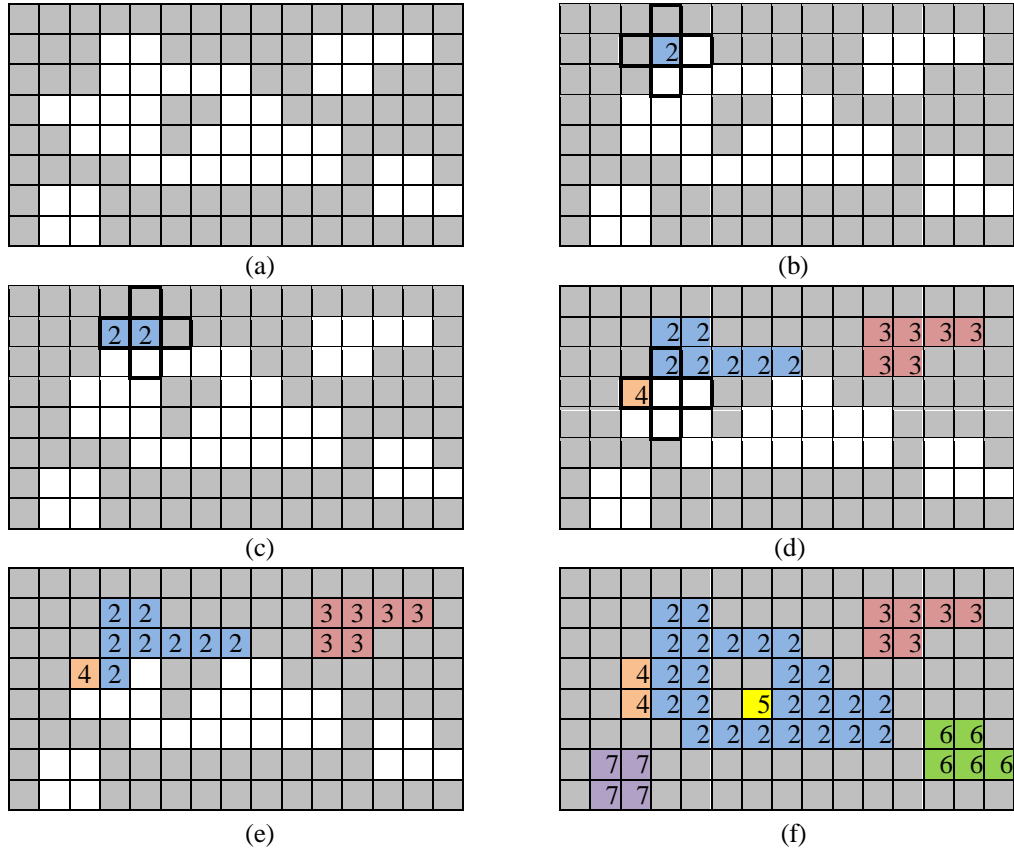
Şekil matematiksel olarak ifade edilmesi güç bir kavramdır ve literatürde genel geçer bir şekil kavramı yoktur. Matematiksel açıklaması zayıf olsa da insanda şekil algısı oldukça yüksek seviyelerdedir. Genel bir tanımlama yapılamamakla birlikte şekil kelimesi bazı bilim çevrelerinde farklı yollardan tanımlanabilmektedir. Örnekte tanımlama alanı için şekil, iki boyutlu bir düzlemde bağlantılı noktaların yönlerine ve konumlarına bağlı bir fonksiyondur (Marshall, 1989). Örnekte tanımlama için Marshall tarafından önerilen bu tanım iki boyutlu görüntülerdeki nesnelere için uygundur. Bu tür tanımlar ilgilenilen görüntüde şekil olarak algılanacak ve çözümlenecek olan nesnelere belirlenmiş olmasını gerektirmektedir. Nitekim şekil çözümlemesi için ilk görüntü genellikle bazı ön işlemlerden geçmektedir. Diğer taraftan nesne çözümlemesi çalışmalarında şekil gösterimi için farklı çözümler üretilmektedir.

## 1.4. Nesnelere Etiketlenmesinde Sıralı Etiketleme Yöntemi

Sıralı etiketleme literatürdeki klasik tekniklerden biridir. Algoritma 2 adımdan oluşur: 1) görüntüdeki nesnelere bir ön etiketleme ve 2) daha sonra aynı nesnedeki birden fazla etiket sayısının birine indirilmesi.

Bu algoritma nispeten karmaşık olmasına rağmen özellikle ikinci aşamada düşük bellek gereksinimi olması onu tercih edilebilir kılmaktadır.

Ön etiketleme adımında önce görüntü sol üstten başlanarak sağ alta kadar her ön plan gözeğine bir etiket atamak için sırayla taranır. Daha önce de belirtildiği gibi burada da arka plan gözeklerinin 0, ön plan gözeklerinin ise 1 değerli olduğu kabul edilir. Önceden tanımlanan komşuluk (4-komşuluk veya 8-komşuluk) kullanılarak her bir gözeğin doğrudan bağlı olduğu diğer komşu gözekler incelenir. Görüntünün kenar gözeklerinin komşuları için genellikle sanal gözeklerin arka plana ait olduğu varsayımıyla sıfırlama tekniği kullanılır. Etiketleme için 4-komşuluk tanımlandığı kabul edelim. Kayan komşuluk bölgesinde etkin gözek ön plan gözeği olduğunda ya yeni bir sayı ile ya da komşuluğundaki kendinden önce etiketlenen -üst ve sol- ön plan gözeklerinin değerlerinden küçük olan sayı ile etiketlenir. Bu şekilde soldan sağa ve yukarıdan aşağıya tüm gözeklerin etiketlenmesiyle, ele alınan sayısal görüntüdeki nesnelerin ön etiketlenmesi işlemi tamamlanır (Burge ve Burger, 2009).



Şekil 1.20. Sıralı etiketlemenin ilk aşaması olan ön etiketlemenin adım adım gösterimi.



## 1.5. Nesnenin Özellikleri

Bu bölümde şekillerin metrik ölçüleri üzerinde durulmuştur. Bu özellikler genellikle şeklin boyutu ve büyüklüğü ile ilgilendiği durumlarda (biyomedikal uygulamalarda bir tümörün büyüklüğünün ölçümü) ön plana çıkmaktadır. Bu kısımda ele alınan bütün metrik özellikler gözek tabanında işlenmiştir. Bu ölçülerin genellikle metre ya da santimetre gibi özgün ekran birimlerine dönüştürülmesi için, uygun çözünürlük kullanılmalıdır ve uygun bir dönüşüm gereklidir. Metrik özelliklerin, komşuluk ve uzaklık tanımının seçiminden etkilenebileceği de göz önünde bulundurması gerekir. Literatürde önerilen çoğu şekil ölçüsü genellikle istenilen şekilde çalışır, fakat bazı durumlarda farklı şekillerin aynı metrik özellikleri göstermesi sık karşılaşılan bir durumdur. Bu durum, genellikle şekle ait özelliğin tanımından doğabilecek eksikliğin bir sonucudur. Bu açıdan bakıldığında pek çok şekil aynı özellik değerlerini üretebilmektedir (Costa ve Cesar, 2001). Örneğin şekil ölçüsü olarak ‘alan’ı ele alalım. Biçimlerindeki büyük farklılıklara rağmen aynı alana sahip olan bütün şekiller aynı özellik değerini üretecektir. Bu tip özellikler bu yönleri ile literatürde pek kabul görmemektedir. Bu noktada her bir uygulama için en uygun özellik kümesini seçmenin önemi ortaya çıkmaktadır.

### 1.5.1. Nesnenin Çevresi

Nesnelerin çevreleri birbirine bağlı eğri parçaları kümesi olarak da tanımlanabilirler. Eğrilere istatistiksel çözümleme uygulayarak bu şekilde tanımlanan nesnelerin önemli özellikleri kestirilebilmektedir, örneğin maksimum, minimum ya da eğrinin ortalama yay uzunluğu. Bu tür özellikler sözdizimsel örüntü tanımada kullanılabilir (Fu, 1982). Uzaysal örnekleme sonucu elde edilen bir sayısal eğrinin toplam yay uzunluğu yani çevresi birçok yaklaşım kullanılarak tahmin edilebilir.

Çevre hesaplamaları için 8 farklı duruma farklı kod numaraları tanımlaması yapılmıştır. Zincir kodları literatürde sık kullanılan kavramlardan biridir. Şekil 1.22’te de gösterildiği gibi eğer ölçüm esnasında etkin gözekten bir sonraki gözek doğu yönünde ise 0, kuzey-doğu yönünde ise 1, kuzey yönünde ise 2, kuzey-batı yönünde ise 3, batı yönünde ise 4, güney-batı yönünde ise 5, güney yönünde ise 6 ve güney-doğu yönünde ise 7 ile kodlanacaktır. Bu kodlara göre nesnelerin çevreleri hesaplanabilmektedir.

3	2	1
4		0
5	6	7

Şekil 1.22. Zincir kodları.

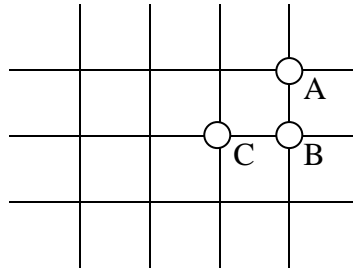
Varsayalım ki bir şeklin dış çevresi bir ikil görüntü içinde soyutlanmış olarak gösterilsin, şeklin çevre uzunluğunu tahmin etmek için en basit yaklaşım aşağıdaki algoritma ile verilmiştir:

**Adım 1.** Girilen ikil görüntüyü oku;

**Adım 2.** İkil görüntüye kenar algılama algoritmasını uygula;

**Adım 3.** Çevre = Adım 2’de algılanan kenar gözeteklerinin sayısı.

Yukarıdaki algoritma 4-komşulukta iyi sonuç verse de 8-komşuluk kullanıldığında sorun olabilmektedir. Çünkü ikinci durumda ardışık gözetekler arasındaki mesafeler sabit değildir. Örneğin gözetekler köşegen komşulukta ise aradaki uzunluk  $\sqrt{2}$ , diğer durumda ise 1’dir. Şekil 1.23’de üç ardışık noktalı ( $A, B, C$ ) bir dış çevre parçası gösterilmektedir.



Şekil 1.23. Sayısal görüntü matrisinde noktalar arası uzaklık farkları.  $A - B$  uzunluğu  $A - C$  uzunluğundan daha kısadır.

Burada  $d(A, B) = 1$  iken  $d(A, C) = \sqrt{2}$  dir. Örnekten de anlaşılacağı gibi ardışık noktalar arasındaki yerel farklar toplam yay uzunluğunu etkilemektedir. Fakat yukarıdaki algoritma bu durumu hesaba katmamaktadır. Bu sorunu engellemenin en kolay yolu şeklin dış çevresinden yay uzunluğunu tahmin etmektir. Örneğin dış çevreyi temsil etmek için 8 komşuluktan yararlanılıyorsa yay uzunluğu aşağıdaki gibi tahmin edilebilir (Castleman, 1996).

$$P = N_e + N_o\sqrt{2}$$

Burada  $N_e$  ve  $N_o$  sırasıyla kod zincirindeki çift ve tek kodların sayısını gösterir.

### 1.5.2. Nesnenin Alanı

Bir nesnenin alanını tahmin etmek için en basit yaklaşım o nesneyi temsil eden gözetlerin sayısını hesaplamaktır. Varsayalım ki  $I$  bir ikil görüntü olsun, burada nesneye ait gözetler için  $I(u, v) = 1$  ve görüntünün taban gözetleri için  $I(u, v) = 0$  olsun. Şekil 1.24'teki algoritma ile nesne alanı tahmini için bir yöntem verilmektedir. Bu algoritma şeklin ikil görüntü içinden soyutlandığını varsayar. Ayrıca birden fazla şekil (nesne) içeren görüntülerde her bir nesnenin etiketlenmiş olmasını gerektirir ve ilgili nesneye ait alanı tahmin eder (Kesemen, 2011).

```
function [a] = alan(I, e)
% sayısal görüntüdeki nesnenin gözet alanı
% I : etiketlenmiş sayısal görüntü
% e : nesne etiket numarası
%
a = 0;
[U,V] = size(I);
for u = 1: U
    for v = 1: V
        a = a + (I(u,v) == e);
    end
end
end
```

Şekil 1.24. Sayısal görüntüdeki bir nesnenin gözet alanını hesaplayan algoritma.

### 1.5.3. Nesnenin Kütle Merkezi

Bir şeklin kütle merkezini tahmin etmek için nesneye ait gözet koordinatlarının ortalama değerlerinin bulunması yeterli olacaktır. Varsayalım ki  $I$  ilgilenilen şekli içeren ikil görüntü olsun, burada nesne gözetleri için  $I(u, v) = 1$  ve taban gözetleri için  $I(u, v) = 0$  değerlerini atayalım. Şekil 1.25 ile verilen algoritma  $I$ 'daki nesnenin kütle merkezini hesaplar (Kesemen, 2011).

```

function [Gu,Gv] = merkez(I, e)
% sayısal görüntünün merkez noktası
% I : etiketlenmiş sayısal görüntü
% e : nesne etiket numarası
%
a = 0;
Gu = 0;
Gv = 0;
[U,V] = size(I);

for u = 1: U
    for v = 1: V
        if (I(u,v) == e)
            Gu = Gu + 1;
            Gv = Gv + 1;
            a = a + 1;
        end
    end
end
Gu = Gu/a;
Gv = Gv/a;
end

```

Şekil 1.25. Sayısal görüntüdeki bir nesnenin kütle merkezini hesaplayan algoritma.

#### 1.5.4. Nesnenin Ana Eksenini

Ana eksen genel olarak bir şeklin herhangi iki noktası arasındaki en büyük uzaklık olarak tanımlanır. Kabaca bir şeklin uzun eksenini hesaplamamanın yolu, nesnenin sınır gözeteklerini oluşturan tüm noktaları ele alıp bunların içindeki her nokta çiftinin arasındaki uzaklıklardan en büyüğünü araştırmaktır. Varsayalım  $I$  ilgili nesneyi içeren ikil bir görüntü olsun. Burada nesneye ait gözetekler için  $I(u, v) = 1$ , ve görüntünün taban gözetekleri için  $I(u, v) = 0$  olsun. Şekil 1.26'daki algoritma şeklin ana eksenini ve  $I$ 'deki nesnenin en uzak noktalarının koordinatlarını hesaplar (Costa ve Cesar, 2001).

```

function [Ci,Cj] = uzunEksen(J)
% sayısal görüntünün ana ekseninin noktalarının
% bulunması
% J : sınır gözetlerinin koordinatları
% J(*,1): yatay eksen değerleri
% J(*,2): düşey eksen değerleri
%
N=length(J);
dmax= 0;

for i = 1: (N-1)
    for j = (i+1): N
        d = sqrt( (J(i,1)-J(j,1))^2 + (J(i,2)-J(j,2))^2 );
        if (dmax<d)
            dmax = d;
            Ci = i;
            Cj = j;
        end
    end
end
end

```

Şekil 1.26. Sayısal görüntüdeki bir nesnenin ana ekseninin koordinatlarını hesaplayan algoritma.

Yukarıdaki algoritmada  $C_i$  ve  $C_j$  en uzak noktaları gösterir, böylece şeklin çapı  $d_{maks}$  belirlenir.

### 1.5.5. Nesnelere İçindeki Boşluklar

Şekillerdeki mevcut boşlukların çözümlenmesinde, bu boşluklar bazı şekilleri ayıklamak için referans kabul edilebilir. Aslında her bir boşluğun bir nesne gibi davrandığı durumlarda bu böyledir. Boşluk tabanlı özellikler de örnek olarak alan, çevre, ana eksen, vb. gibi boyutları içerebilir (Costa ve Cesar, 2001).



## 2. YAPILAN ÇALIŞMALAR

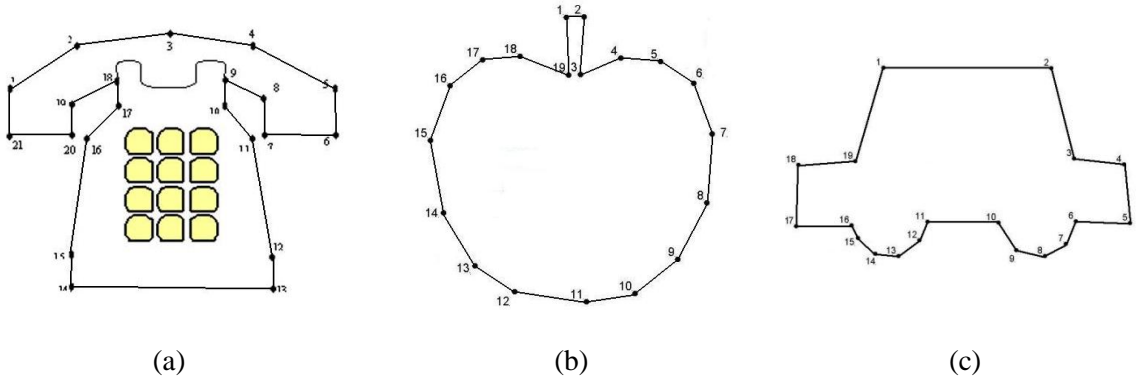
### 2.1. Nesne Sınırlarına Çokgensel Yaklaşım

Bir nesnenin dış hatlarının gösterimi için bir çokgen ile iyi bir yaklaşım yapıldığında, nesnenin birçok özelliğini belirlemek ve temsil etmek mümkün olmaktadır.

Nesne çözümlemesindeki en önemli problemlerden biri nesnenin çevresinin anlamlı parçalara bölünmesidir. Nesnenin çevresinin parçalara ayrılmasını tanımlayan sınır gözeteklerine baskın noktalar denir ve birçok farklı yoldan elde edilebilir. Çevrenin bölütlenmesi iki aşamada gerçekleştirilir: İlk aşamada çevre boyunca baskın noktalar belirlenir, ikinci aşamada bu noktalar sayısal doğrularla birleştirilerek her bir bölüt yeniden temsil edilir. Noktaları birleştiren yapılar sayısal doğrular olduğundan bu sürecin sonuç görüntüsü özgün nesnenin sınırlarına ait eğrinin bir çokgensel gösterimidir. Bu yaklaşıma çokgensel yaklaşım denir (Ramer, 1972).

#### 2.1.1. Baskın Nokta

Baskın noktalar bir nesnenin sınır noktaları içinde nesneye ait en önemli bilgileri taşıyan noktalardır. Nesnelerin baskın noktalarının araştırılmasının hedefi sayısal görüntüdeki bir nesnenin çevresel noktalarını sınırlı sayıda nokta ile temsil edebilmektir. Bu sayede bütün bir nesne sadece birkaç nokta ile temsil edilecektir (Şekil 2.1).



Şekil 2.1. Nesnelerin sınırlarındaki baskın noktalar ile gösterilmesi. (a) 21 adet nokta ile bir telefonun temsili; (b) 19 nokta ile bir elmanın temsili; (c) 19 nokta ile bir otomobilin temsili.

Dolayısıyla örüntü tanıma tekniklerinin yaygın olarak kullanıldığı pek çok bilimsel ve endüstriyel alandaki analitik hesaplamaların hızlı ve etkin bir biçimde yapılması sağlanacaktır. Bu problem için birçok kişi tarafından birçok farklı yöntem önerilmiştir.

Bu yöntemler bölme-birleştirme (Pavlidis ve Horowitz, 1974; Wu ve Leou, 1993), sıralı yaklaşım (Ray ve Ray, 1991), çevre bölütleme (Leu ve Chen, 1988), Newton yöntemi (Pavlidis, 1977) ve minimax yöntemi (Kurozumi ve Davis, 1981) gibi konuları kapsamaktadır. Ancak bu yöntemlerin çoğu sıralı ve yerel bir optimizasyon yapma eğilimindedir. Bunlardan Teh-Chin algoritması (Teh ve Chin, 1989) sayısal bir kapalı eğri üzerindeki baskın noktaların tespiti için iyi bir örnektir. Algoritma ilk olarak her bir nokta için kendi yerel özelliklerine göre bir destek bölgesi belirler, sonra bir anlamlılık ölçüsü belirler ve her nokta için en iyi değeri hesaplar. Baskın noktaların tespiti yalnızca anlamlılık ölçüsünün doğruluğuna dayanmayıp, aynı zamanda seçilen bölgenin de kesin ve tam olarak belirlenmesine dayanır. Teh-Chin algoritması gürültü ile bozulmamış kapalı eğrilerde iyi çalışır. Ne var ki gürültü olması durumunda doğru baskın noktaları bulamamaktadır. Yöntemde etkin gözeğe zıt yönlerdeki aynı uzaklıkta iki gözek ele alınır. Her iki gözek ile etkin gözeğin oluşturduğu iki vektör arasındaki açı hesaplanır. Bu durumda şayet etkin gözeğe zıt yönlerde seçilen gözekler kenardan fazla sapmış ise yani gürültülü bir yapı gösteriyorsa bunlara ilişkin vektörlerden elde edilen açı değeri baskın noktanın belirlenmesinde yanıltıcı olmaktadır. Gürültü etkisini azaltmak için Ansari ve Huang baskın nokta tespiti için Gauss yumuşatması ve destek bölgesi kavramı içeren bir yöntem (Ansari ve Huang, 1991) tanıtmışlardır.

Baskın noktaların farklı disiplinlerde birbirinden farklı birçok uygulama alanı vardır. Bu uygulama alanları: tıpta birçok hastalığın teşhisi; askeri alanlarda uydu görüntüleriyle istihbaratların derlenmesi; akıllı füzelerin hedeflerini bulması; yeryüzündeki bitki çeşitliliğinin araştırılması, hayvan kitlelerinin göçlerinin ve nüfuslarının incelenmesi; kayaçların mineral içeriklerinin araştırılması, uydu görüntülerinden yer küredeki fay hatlarının kırılma noktalarının incelenmesi vb. konular sayılabilir (Kesemen vd, 2010).

Genel olarak baskın nokta algılama için kullanılan yöntemler 2 yaklaşım altında sınıflandırılır. Bunlardan ilki uyum iyiliğine bazı kısıtlar getirerek sayısal eğrilerin çokgensel yaklaşımının elde edildiği genel (global) yaklaşımdır. Diğeri doğrudan açı veya köşe algılama sistemleri yoluyla baskın noktaların tespit edildiği yerel (local) yaklaşımdır. Belirtmek gerekir ki değişken kırılma noktaları ile yapılan çokgensel yaklaşım baskın noktalar olarak bölgesel köşeleri bulma eğilimindedir.

Bu çalışmada problemin çözümü için hem doğrudan köşe algılama teknikleri kullanılarak yerel anlamda dört farklı yaklaşım, hem de Tabu arama algoritması (Glover, 1989) ve Genetik Algoritmadan (Haupt ve Haupt, 2004) yararlanılarak önceden tanımlanan bir amaç fonksiyonunun değerinin optimize edilmesine yönelik genel anlamda yaklaşımlar uygulanmıştır.

## 2.2. Ön İşlemler

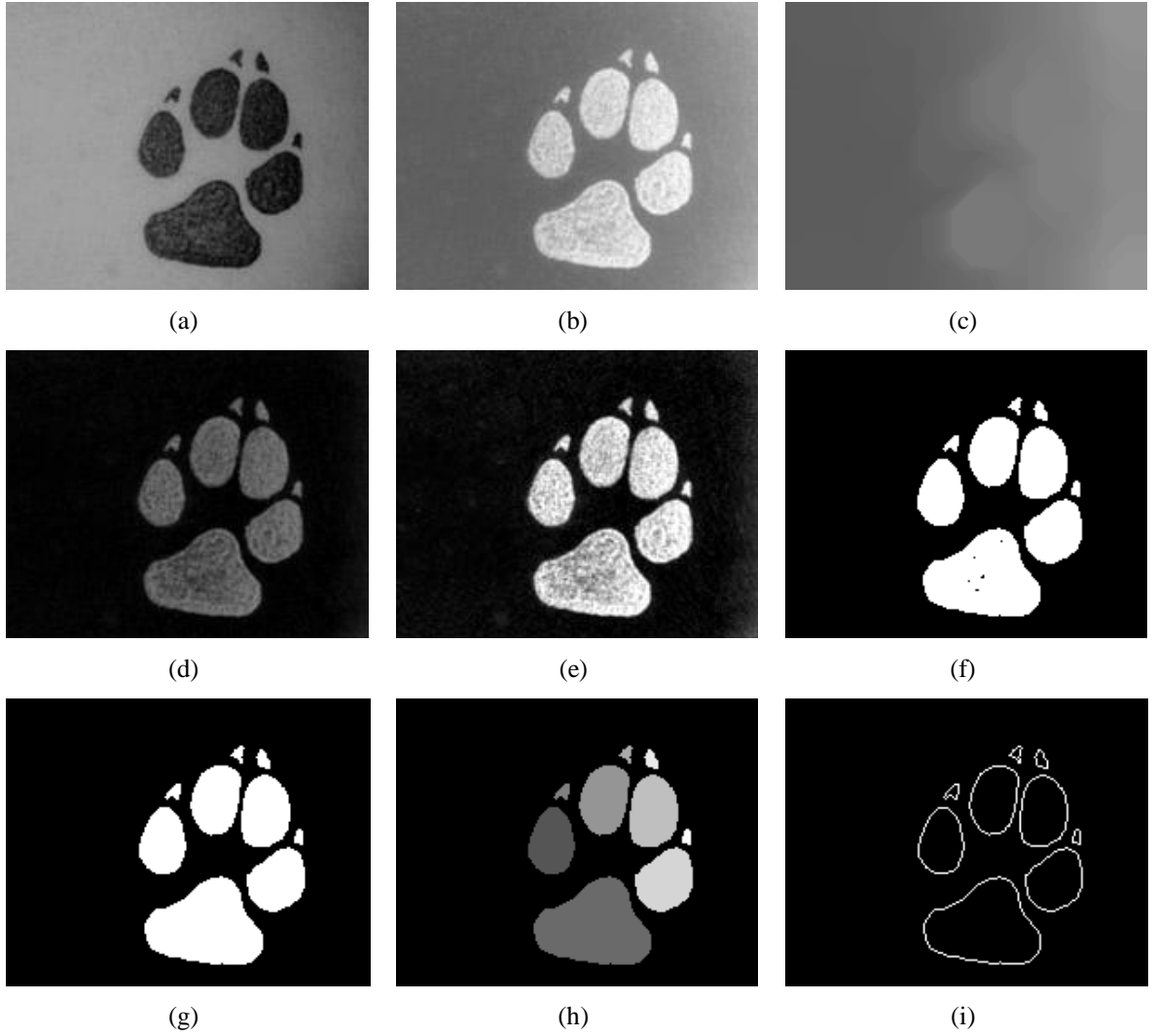
Görüntü çözümlene yöntemleri genellikle ham (işlenmemiş) görüntü üzerinde uygulanmamaktadır. Görüntüler onları oluşturan bileşenler içinde çözümlenmesi yapılacak olan bileşenler elde edilinceye ve bu bileşenler hesaplamalar için uygun duruma getirilinceye kadar ön işlemlerden geçirilirler.

Ele alınan bir ham sayısal görüntü, öncelikle boz ayrınçlı bir görüntüye dönüştürülür (Şekil 2.2(a)). Görüntü içerisinde ilgilenilen nesnelere artalan renginden daha koyu ise ters renk görüntüsü alınarak (Şekil 2.2(b)) nesnelere açık renge dönüştürülmesi sonraki işlemlerde kolaylık sağlayacaktır. Sayısal görüntünün elde edildiği fiziksel şartlardan dolayı görüntünün her bir noktası eşit derecede aydınlık olmayabilir. Bu durumun doğal bir sonucu olarak görüntünün artalanı genellikle tek renk değerinden oluşmayacağından renk farklılıkları söz konusu olur. Kayan ortalama süzgeci (Buzuloiu, 2001) yardımıyla görüntünün artalan görüntüsü elde edilir ve bu görüntü önceki görüntüden çıkarılarak artalan sıfırlanmış görüntü elde edilir (Şekil 2.2(c)).

Elde edilen son görüntü değerleri artalanın çıkarılmasından dolayı dar bir ayrınç aralığına sıkıştığı için görüntü ayrımlılığında (contrast) düşme meydana gelmektedir. Aşağıdaki özayrımılık eşitliği kullanılarak görüntünün ayrımlılığı artırılır (Şekil 2.2(d)).

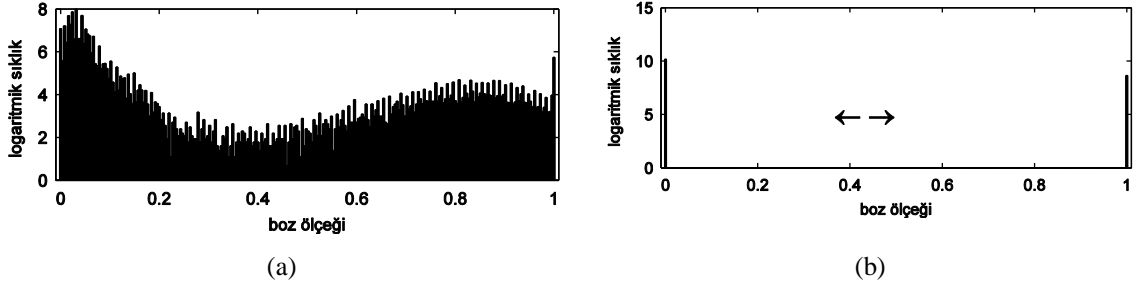
$$\hat{I}(u, v) = \frac{I(u, v) - \text{düşük}}{\text{yüksek} - \text{düşük}} \quad (5)$$

Burada  $I(u, v)$ ,  $(u, v)$  noktasındaki gözetik değeri ( $0 \leq I(u, v) \leq 255$ ), *düşük* ve *yüksek* değişkenleri ise sayısal görüntüdeki sırasıyla en düşük ve en yüksek gözetik değerleridir. Ayrıca  $\hat{I}(u, v)$ , işlenmiş görüntü gözetiklerini göstermektedir. Bu yöntem bir anlamda bir birimleştirme (normalization) yöntemi olarak da kullanılır.



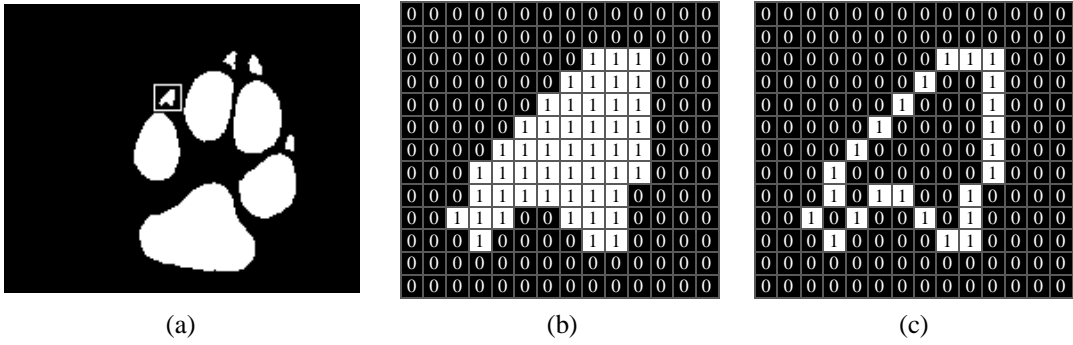
Şekil 2.2. Sayısal görüntünün ön işlem adımları. (a) Boz ölçekli özgün görüntü; (b) ters renkli görüntü; (c) önceki görüntünün artalan görüntüsü; (d) artalan çıkarılmış görüntü; (e) ayrımlılığı artırılmış görüntü; (f) eşiklenmiş görüntü; (g) boşlukları doldurulmuş ikil görüntü; (h) nesnelere etiketlenmiş görüntü; (i) nesnelere sınırları çıkartılmış görüntü.

Şekil 2.2(d)'deki görüntünün iki ayrı veri kümesinin (artalan-nesnelere) birleşimi olduğu Şekil 2.3(a)'daki sıklık göstergesinde açıkça görülmektedir. Bu iki kümenin eşik değeri Otsu algoritması (Otsu, 1979) yardımıyla 0.42 bulunarak, bu noktadan küçük olanlar siyah, büyük olanlar beyaza eşiklenirse görüntü Şekil 2.2(e)'ye dönüşür.



Şekil 2.3. Arka planı çıkarılmış ve ayrırlılığı artırılmış görüntünün (a) logaritmik sıklık göstergesi (b) eşiklenmiş sıklık göstergesi.

Değer kümesinin daha iyi anlaşılması için görüntünün küçük bir kesiti (Şekil 2.4(a) dikdörtgen alan içine alınan kısmı) alınarak dizeysel olarak incelenirse Şekil 2.4(b)'deki dizey elde edilir. Bu dizeyde siyah gözetler 0 değerini alırken, beyaz gözetler 1 değerine karşılık gelmektedir. Bu nesnenin sınırları ise Şekil 2.4(c)'de verilmiştir.



Şekil 2.4. İkil görüntü içerisindeki bir nesne için sınır belirleme adımları. (a) Sayısal görüntü ve dikdörtgen içinde seçilmiş nesne; (b) büyütülmüş ve ayrılaştırılmış görüntü ve sayısal görüntü matrisi; (c) sınırları belirlenmiş sayısal görüntü matrisi (Kesemen ve Güngör, 2011).

Şekil 2.2(f)'de eşikleme sonucunda elde edilen ikil görüntüdeki nesnelere içerisinde boşlukların olduğu görülmektedir. Boşlukları kapatmak için oyuk kapatma işlemi (Soille, 1999) uygulanır (Şekil 2.2(g)). Ön işlemlerden geçirilerek kullanıma hazır duruma getirilmiş görüntü içerisindeki nesnelere tek tek etiketlenir (Şekil 2.2(h)). Etiketleme işlemi her gözetin 4 komşusuna (kuzey, güney, doğu ve batı) bakılarak, her bir nesneye farklı gözet değerleri atanması yoluyla gerçekleştirilir (Kesemen vd, 2010). Etiketlenen her nesne ayrı ayrı ele alınarak gerekli adımlar uygulanır ve nesnenin sınır gözetleri bulunur (Kesemen ve Güngör, 2011).

### 2.3. Yapay Çokgen Oluşturma

Çokgenler, birbirinden farklı, herhangi üçü doğrusal olmayan  $n$  tane noktayı ikişer ikişer birleştiren doğru parçalarının oluşturduğu 2 boyutlu kapalı şekiller olarak tanımlanırlar (Wikipedia, 2011). Matematiksel olarak (6) biçiminde ifade edilir.

$$\Omega = \{p_i | i = 1, 2, \dots, N : p_i \in \mathbb{R}^2\} \quad (6)$$

Burada  $p_i$ 'ler çokgenin köşe noktalarını gösterir  $N$  ise çokgene ait köşe sayısını belirtir.

Sayısal ortamda oluşturulmuş yapay rastgele bir çokgen koordinat düzleminde rastgele belirlenen noktalardan sayısal doğru parçalarının çizdirilmesi esasına dayanır. Köşe noktalarının düzlemde rastgele seçildiği ve bu noktalar kümesi kullanılarak elde edilecek olan sayısal doğru parçalarının birbirini kesmeyecek şekilde oluşturulması gerekmektedir. Bu işlem için daha kolay bir yaklaşım ise, düzlemde bir merkez noktası (C) seçilerek, bu nokta etrafında  $r$  yarıçaplı bir çember kullanılır. Çokgenin çevresel noktalarını bu çember üzerinden sırasıyla rastgele seçmek ve bunları ikişer ikişer birleştirmek sorunu çözecektir. Bu şekilde yapay çokgenler kolaylıkla oluşturulabildiği gibi aynı zamanda parametrelerin de kontrol edilebilirliği söz konusu olmaktadır.

Göreceliliğin en aza indirgenmesi amacıyla, önerilen yöntemleri test etmede kullanılacak çokgenlerin yapay ve rastgele olması gerekmektedir. Bu amaç doğrultusunda aşağıdaki yapay çokgen oluşturma algoritması geliştirilmiştir.

#### 2.3.1. Rastgele Çokgen Oluşturma

Kullanılan yapay çember üzerinden eşit açı aralığında noktalar seçilerek bir düzgün çokgen oluşturulabilir.

$$a_{i+1} = a_i + da, \quad i = 1, 2, \dots, N \quad (7)$$

Burada  $N$  çokgenin köşe sayısıdır.  $a_1$  başlangıç açısıdır ve keyfi olarak belirlenir.  $da$  açısal artım miktarıdır ve  $da = 2\pi/N$  eşitliğiyle hesaplanır. Bu yöntemde köşe noktaları sırasıyla seçildiğinden hem çizimde hem de hesaplamalarda kolaylıklar sağlamaktadır.

Ancak çeşitkenar bir çokgen oluşturmak için, çokgenin açılarını ve kenar uzunluklarının farklı olması gerekir. Bunun için öncelikle açılar eşit artımla değil de  $da = 2\pi i/N + U(-\pi/N, \pi/N)$  eşitliğiyle rastgele biçimde seçilir. Burada,  $U$  fonksiyonu  $(-\pi/N, \pi/N)$  aralığında düzgün dağılımdan rastgele sayı üreticidir. Aynı zamanda, yarıçapın da sabit değil,  $r = U(0, r_0)$  biçiminde rastgele seçilmesi çokgenin rastgeleliğini artıracaktır. Bu yaklaşımlar ışığında üretilen yapay çokgen algoritması Şekil 2.5' te gösterilmektedir.

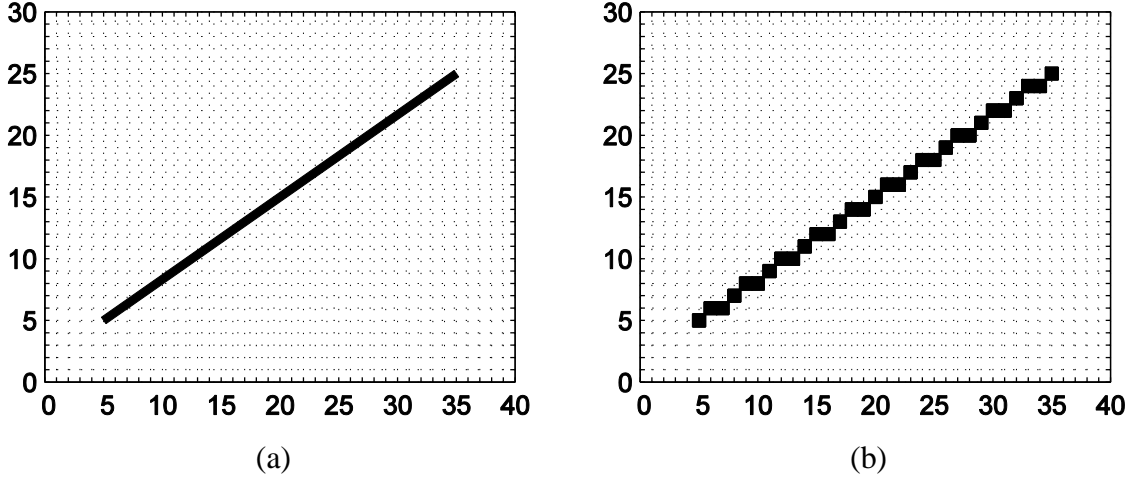
```
function [x,y]=cokgen(N, Cx, Cy, r0, a0)
% N : köşe sayısı
% Cx : çokgenin geçici merkezinin x konumu
% Cy : çokgenin geçici merkezinin y konumu
% r0 : rastgele yarıçapın en yüksek değeri
% a0 : çokgenin başlangıç açısı
%
a=a0;
da=2*pi/N;
for i = 1:N
    r = unifrnd(0, r0);
    de =unifrnd(-pi/N, pi/N);
    x(i) = Cx + r*cos(a);
    y(i) = Cy + r*sin(a);
    a = i*da+de;
end
end
```

Şekil 2.5. Rastgele çokgen oluşturma algoritması.

### 2.3.2. Sayısal Doğru Oluşturma

Noktadan sonra, sürekli uzaylardaki en basit geometrik yapı doğru ve doğru parçasıdır. Düzlemdeki bir doğru, en kısa Öklid uzaklığına sahip nokta çiftlerinin oluşturduğu ardışık sonsuz noktalar kümesinin geometrik yeri olarak tanımlanabilir. Sürekli uzaylardakinin aksine doğrular gözet tabanlı olarak ele alındığında, karmaşık bir davranış sergilerler. Örneğin gözet tabanında herhangi iki nokta verilmiş olsun, bu iki noktadan birden çok sayısal doğru parçası geçebilmektedir. Bu nedenle bir  $N \times N$  görüntüdeki sayısal doğruların olası sayısı henüz kestirilememektedir. Görüntü işleme ve nesne çözümlemesinde ele alınan temel nesnelere, sayısal doğruların özelliklerinin iyi anlaşılması açısından önemlidir, bu sayede şekiller daha doğru temsil edilmekte ve algılanmaktadır.

Sayısal doğrular, sürekli doğruların uzaysal örneklemesinin bir sonucudur. Belirli bir sürekli doğruya ilişkin bir sayısal doğru belirli bir  $\Delta = \Delta x = \Delta y$  aralığında uzaysal örnekleme yapılarak elde edilebilir. Şekil 2.6'da genel bir doğru parçası ve bir  $\Delta$  örnekleme ile gözet tabanlı oluşturulan sayısal doğru parçası gösterilmektedir. Hesaplamalarda kolaylık sağlaması açısından  $\Delta = 1$  olarak kabul edilmektedir (Costa ve Cesar, 2001).



Şekil 2.6. Sürekli doğrudan sayısal doğru parçası oluşturma. (a) sürekli doğru parçası; (b) sürekli doğrunun  $\Delta$  aralığı ile elde edilmiş uzaysal örnekleme.

Sayısal doğrular için kartezyen düzlemde eğim-kesme denklemi (8)'deki gibidir:

$$y = mx + b \quad (8)$$

Burada  $m$  doğrunun eğimi ve  $b$  de doğrunun  $y$ -eksenini kestiği noktayı gösterir. İki noktası  $(x_1, y_1)$  ve  $(x_2, y_2)$  şeklinde belli olan doğrunun denklemini bulmak için gereken  $m$  ve  $b$  değerleri (9) ve (10) eşitlikleri ile elde edilir (Hearn ve Baker, 1997).

$$m = \frac{y_2 - y_1}{x_2 - x_1} \quad (9)$$

$$b = y_1 - mx_1 \quad (10)$$



Sayısal doğru oluşturma algoritmaları (8) eşitliğini temel alır ve (9) ve (10) eşitliklerinde verilen hesaplamalar kullanılır.

$$\begin{aligned} x_{i+1} &= \text{round}(x_i + dx), \quad i = 1, 2, \dots, N - 1 \\ y_{i+1} &= \text{round}(y_i + dy) \end{aligned} \quad (11)$$

Sayısal doğruları çizmek için verilen başlangıç  $(x_1, y_1)$  ve bitiş  $(x_N, y_N)$  noktaları verilen doğru parçasının, aradaki tüm noktaları için (11) eşitliği kullanıldı. Burada,

$$dx = \begin{cases} \text{sign}\left(\frac{x_N - x_1}{y_N - y_1}\right), & \left|\frac{y_N - y_1}{x_N - x_1}\right| < 1 \\ \frac{x_N - x_1}{y_N - y_1}, & \text{diğer durumlarda} \end{cases} \quad (12)$$

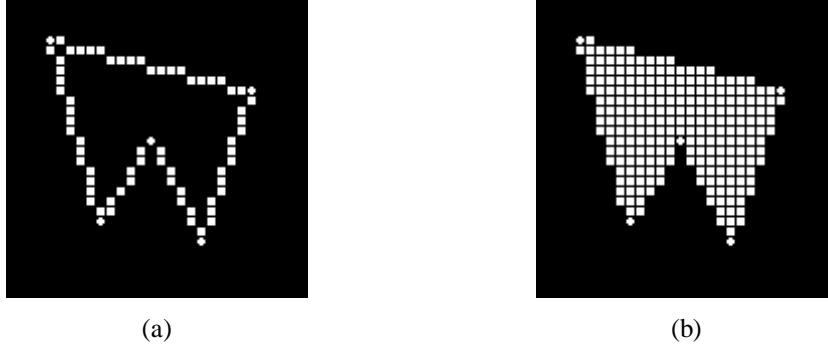
ve

$$dy = \begin{cases} \text{sign}\left(\frac{y_N - y_1}{x_N - x_1}\right), & \left|\frac{x_N - x_1}{y_N - y_1}\right| < 1 \\ \frac{y_N - y_1}{x_N - x_1}, & \text{diğer durumlarda} \end{cases} \quad (13)$$

biçiminde verilmektedir.

### 2.3.3. Sayısal Çokgen Oluşturma

Rastgele çokgen algoritmasıyla elde edilen köşe noktaları, sayısal doğru algoritması yardımıyla birleştirilerek sayısal kapalı bir eğri elde edilir (Şekil 2.7(a)). Bu kapalı eğrinin iç kısmı doldurularak ikil bir görüntüde bir nesne oluşturulur (Şekil 2.7(b)).



Şekil 2.7. Sayısal çokgen oluşturma, (a) sayısal doğrular yardımıyla oluşturulan kapalı eğri; (b) içi doldurulmuş çokgen nesne.

#### 2.4. Baskın Noktaların Belirlenmesinde Maliyet Fonksiyonu ve Uyum Değeri

Optimizasyon probleminde kesin çözüm olmaması veya kesin çözümün imkansız bir sürede hesaplanacağı durumlarda, problemi çözmek için birçok yöntem öne sürülür. Öne sürülen bu yöntemlerin başarısını ölçmek için bir değerlendirme ölçütüne gerek duyulur. Bu konuda yapılan çalışmalarda değerlendirme ölçütü olarak, sınır gözetlerinin kendisine en yakın kuramsal çokgen çizgisine olan dik uzaklıklarının karelerinin toplamı alınır. Bu kabul sayısal bir görüntünün sürekli bir eğriyle değerlendirilmesi temeline dayanmaktadır. Bu konuya daha doğru bir yaklaşım yapmak için sınır gözetlerinin sayısal doğrularla değerlendirilmesi ilkesine dayanan (14)'deki maliyet fonksiyonu geliştirildi.

$$E(I, J) = \sum_x \sum_y |I(u, v) - J(u, v)| \quad (14)$$

Burada  $I(u, v)$  işleme sokulan ikil görüntü (tek nesne içeren),  $J(u, v)$  ise seçilen baskın noktalar ile oluşturulan yapay çokgen nesne görüntüsüdür. Maliyet fonksiyonunun değeri  $E(I, J)$  pozitif tamsayı değeri almaktadır ve sifıra yaklaştıkça baskın noktanın konumunun doğruluk derecesi artmaktadır.  $E(I, J)$ 'nin sıfır olması durumunda nesnenin her kenarının düzgün doğru olduğu ve baskın nokta konumlarının ise doğru yerde olduğu sonucu çıkar.

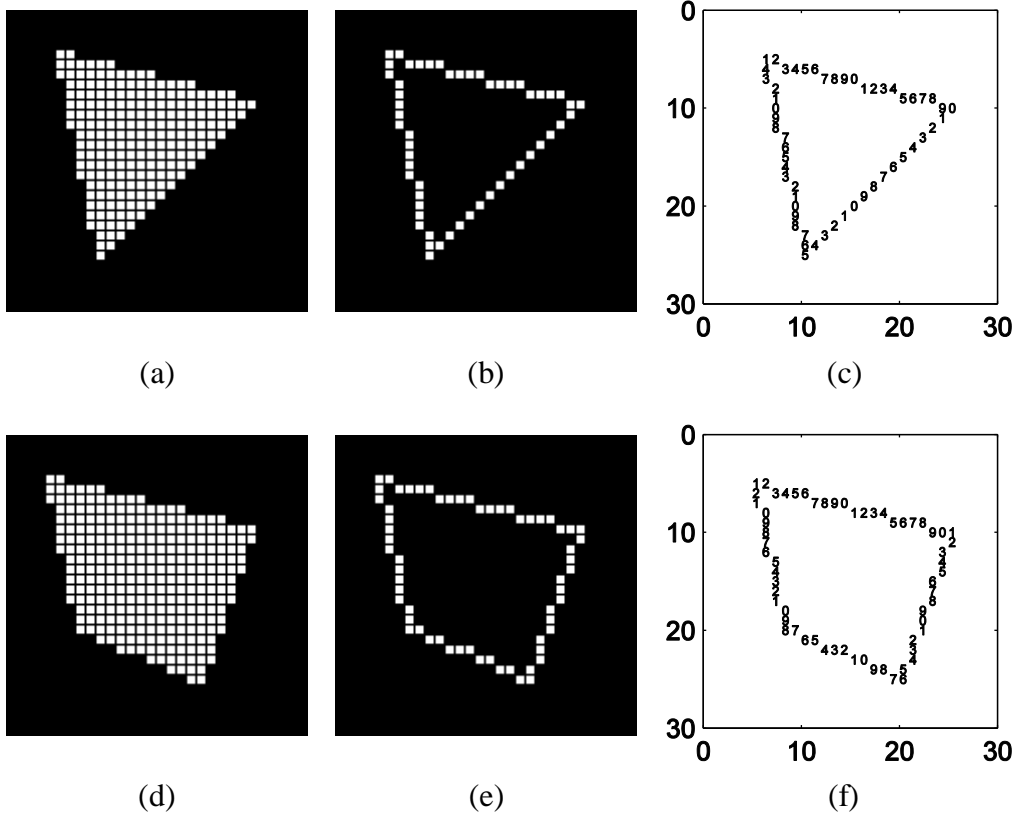
Baskın noktanın belirlenmesinin hata değerini belirleyen maliyet fonksiyonu 0 değeri ile nesnenin gözetksel alan değeri arasında değişen değerler almaktadır. Nesnelerin göreceli karşılaştırılması için bir uyum değerine gerek vardır. Uyum değeri (15) ile verilen eşitlikle hesaplanır.

$$F_{uyum} = 1 - \frac{E(I,J)}{\sum_u \sum_v I(u,v)} \quad (15)$$

Bu uyum değeri, nesnelerin farklı köşe sayılarına, farklı büyüklüklerine ve farklı şekillere sahip olmaları durumunda bile farklı yöntemlerin başarımını karşılaştırma imkanı sağlar.

## 2.5. Yerel Anlamda Baskın Nokta Algılama İçin Önerilen Yöntemler

Bu çalışmada köşe gözeklerinin bulunması için literatürdeki yöntemlerden farklı olarak dört yerel yöntem önerilmiştir. Yöntemlerin işleyişini göstermek için oluşturulan yapay çokgenler Şekil 2.8(a) ve (b)'de verilmektedir. Bu çokgenlerin sınır gözekleri Şekil 2.8(b) ve (e) gibi bulunarak bunları sol-üst köşeden başlanarak etiketlenmiştir (Şekil 2.8(c) ve (f)).



Şekil 2.8. Yapay çokgenin sınır gözeklerinin belirlenmesi, (a) üç köşeli sayısal görüntü; (b) sınır gözekleri; (c) sınır gözeklerinin etiketlenmesi; (d) üç köşeli sayısal görüntü; (e) sınır gözekleri; (f) sınır gözeklerinin etiketlenmesi.

Önerilen yöntemlerin her biri için Şekil 2.8'deki iki yapay çokgen kullanılmıştır. Çokgenlerin tipik özellikleri birinin (yapay üçgen) olası köşe noktasının gözle ve hesaplama ile rahatlıkla belirlenebilmesi, diğerinin (yapay dörtgen) ise bir köşesinin diğer köşeleri kadar belirgin olmaması yani kenar olma karakteri gösterme eğiliminde olmasıdır.

### 2.5.1. Yerel Yöntemler İçin Baskın Noktaların Doğruluk Kriteri

Yerel yöntemler için anlamlılık fonksiyonunun grafiklerin incelenmesinde ve hesaplanan değerlerden yola çıkılarak baskın noktaların belirlenmesinde dikkat edilen nokta, sayısal görüntülerin yapay veya gerçek görüntü olsun ayrık olduğu ve genellikle köşe sayılarının ve koordinatlarının göreceli olduğudur. Yani kimi bakış açısına göre herhangi bir çokgenin sınır gözeteklerinden  $p_n$  noktası köşe gözeği olabileceği gibi bir başkası için  $p_{n+1}$  veya  $p_{n-1}$  noktası köşe gözeği olabilmektedir. Nesnenin baskın noktalarının hangileri olduğu konusunda kesin bir yargıya varılmasının sağlanması gerekir. Bu noktadan hareketle önerilen yöntemler için hesaplanan anlamlılık fonksiyon değerleri tek bir köşe için tek bir gözetek araştırma şeklinde değil de bir gözetekler kümesi araştırma şeklinde değerlendirilmiştir. Bunun için önerilebilecek ölçütlerden biri, köşe olabilecek gözeteklerin oluşturacağı kümenin genel bir eleman sayısını iyi bir şekilde belirlemektir. Sonuç olarak bu çalışmada köşe olabilecek noktaların oluşturacakları küme için keyfi seçilen eleman sayısı 7 ( $x_{n\pm 3}, y_{n\pm 3}$  şeklinde) olarak kararlaştırılmıştır. Bu kararın yöntemlerin başarısının yorumlanmasında göz önünde bulundurulması gerekir. Örneğin Şekil 2.8'teki çokgenler için belirlenen köşe noktaları, üçgen için  $i = 1, i = 20$  ve  $i = 35$  noktaları; dörtgen için ise  $i = 1, i = 21, i = 36$  ve  $i = 48$  noktalarıdır. Dolayısıyla yukarıda da belirtilen şekliyle belirlenen hata aralığı gereği ( $i \pm 3$ ). noktalar da buldukları durumda doğru olarak kabul edilecektir.

### 2.5.2. Kütle-Zemin Oranı Yöntemi

Bu yöntemde  $n \times n$ 'lik bir yapı biriminden yararlanılmıştır. Bu yapı birimi nesnenin sınır gözetekleri üzerinde gezinerek merkez noktası (etkin gözetek) ile çakışacak şekilde etkin gözetek etrafındaki komşuluk gözetekleri hesaplamaya katılmaktadır. Yöntem merkezdeki etkin gözetek komşuluğundaki gözeteklerden nesne olarak etiketlenmiş gözeteklerin sayısı ile zemin olarak etiketlenmiş gözeteklerin sayısının oranlanması esasına dayanır.

Bu oran, kenar gözekleri boyunca 1'e yakın olup küçük sapmalar göstermekte, ancak köşeye yaklaşıldıkça 1'den uzaklaşmakta ve büyük sapmalar göstermektedir. Büyük sapmaların gözleendiği bu durum ise ilgili etkin gözeğin muhtemel bir köşe noktası olduğu anlamına gelmektedir. Aşağıda nesnenin sınırlarına ait her bir gözeğin komşularının nesne değerlerinin toplamı olarak hesaplanan  $s_n$  değerleri,

$$s_n = \sum_{u=x_n-m}^{x_n+m} \sum_{v=y_n-m}^{y_n+m} I(v, u), \quad n = 1, 2, \dots, N \quad (16)$$

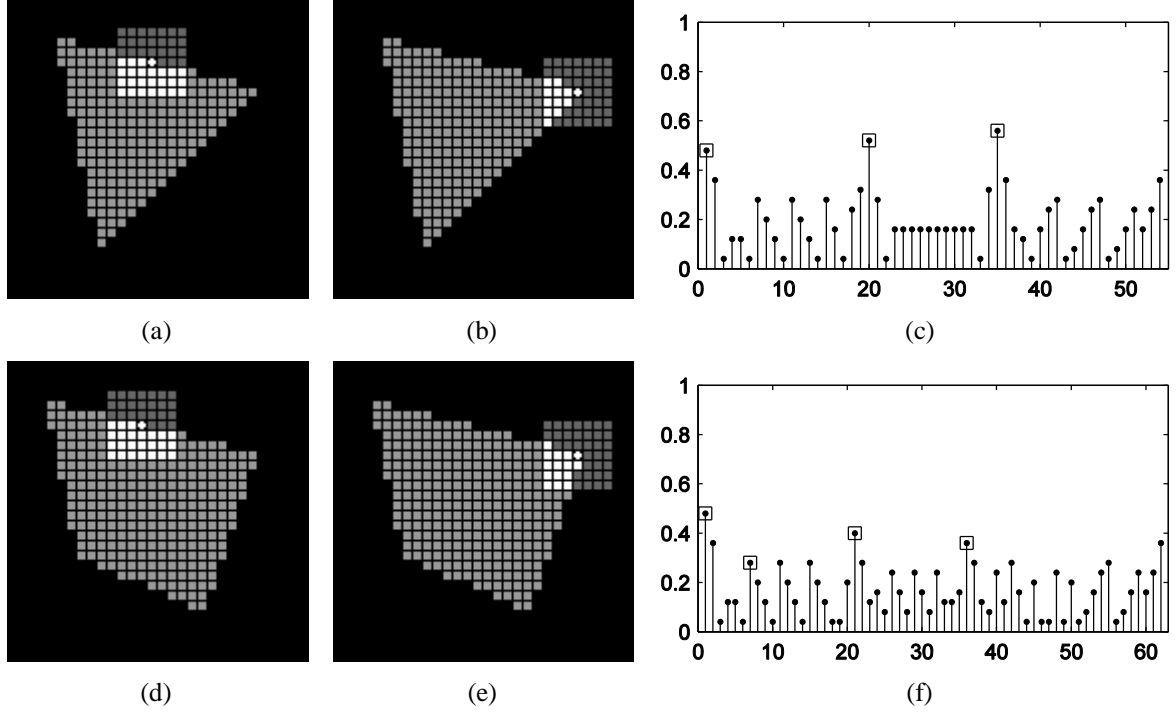
biçiminde verilirken, bu katsayıardan hesaplanan anlamlılık fonksiyonu,

$$e_n = 1 - \frac{2}{M} \begin{cases} M - s_n, & s_n > \frac{M}{2} \\ s_n, & s_n \leq \frac{M}{2} \end{cases} \quad (17)$$

biçiminde verilmektedir. Burada  $M = (2m + 1)^2$  seçilen bölgedeki toplam gözek sayısıdır.  $m$  işleme katılan komşuluk derecesidir ve kullanıcı tarafından keyfi olarak seçilir.  $x_n$  ve  $y_n$  hesaplanan gözeklerin yatay ve düşey eksenlerinin koordinat değerleridir.  $s_n$  seçilen bölgedeki kütleyle ait gözeklerin sayısı,  $e_n$  kütle – zemin oranı yöntemi için hesaplanan sonuç değerlerinin (0,1) arasına çekilmesi için tanımlanan anlamlılık fonksiyonudur.

Şekil 2.9'da üçgen ve dörtgen olmak üzere iki farklı yapay çokgen üzerinde yapılan deneme gösterilmektedir. Şekil 2.9 (a) ve (b)'de yapay üçgenler üzerindeki yapı biriminin iki farklı konumlanması verilmiştir. Sırasıyla birinde etkin gözek kenar gözeği iken diğerinde köşe gözeğidir. Burada yapı birimi olarak seçilen bölgedeki koyu renklendirilmiş noktalar zemin gözeklerini gösterirken, beyaz ile renklendirilmiş noktalar kütle (nesne) gözeklerini göstermektedir. Şekil 2.9(c) ile hesaplanan  $e_n$ ,  $n = 1, 2, \dots, N$  değerleri için grafik verilmiştir. Grafikte 1., 20. ve 34. noktalar anlamlılık fonksiyonun en yüksek olduğu üç noktayı göstermektedir. Bu noktalar aynı zamanda baskın gözek konumlarına karşılık gelmektedir. Burada ilk iki nokta doğru olarak bulunmuş, diğer nokta ise 1 sapma ile doğru noktaya en yakın nokta olarak bulunmuştur.

Yöntemin bu şekil için oldukça başarılı çalıştığı söylenebilir. Benzer şekilde, Şekil 2.9 (d) ve (e)'de yapay dörtgenler üzerinde yapı biriminin iki farklı konumlanması verilmiştir.



Şekil 2.9. Sınır gözeklerinin kütle-zemin oranlaması, (a) yapay üçgenin kenarından seçilmiş bir etkin gözek 3 komşuluğu; (b) yapay üçgenin köşesinden seçilmiş bir etkin gözek 3 komşuluğu; (c) yapay üçgenin tüm sınır gözekleri için oranlama değerleri grafiği. (d) yapay dörtgenin kenarından seçilmiş bir etkin gözek 3 komşuluğu; (e) yapay dörtgenin köşesinden seçilmiş bir etkin gözek 3 komşuluğu; (f) yapay dörtgenin tüm sınır gözekleri için oranlama değerleri grafiği.

Sırasıyla birinde etkin gözek kenar gözeği, diğerinde ise köşe gözeğidir. Burada yapı birimi olarak seçilen bölgedeki zemin ve nesne gözeklerinin renklendirilmesi aynı şekildedir. Şekil 2.9(f) ile de ilgili dörtgen bölgenin sınır gözekleri için hesaplanan  $e_n$ ,  $n = 1, 2, \dots, N$  değerleri için grafik verilmiştir. Bu grafikte farklı olarak 1., 20. ve 36. noktalar üçüncü noktanın 1 birimlik bir sapma göstermesi dışında doğru olarak bulunurken 48. nokta yerine 7. nokta köşe olarak algılanmıştır.

Yerel yöntemlerden biri olan kütle – zemin oranı yöntemi kenarlar arası yumuşak geçişlerin olduğu durumlarda köşe noktalarını algılamakta zorlanmaktadır.

Beklenmeyen değerlerin üretildiği böylesi durumlarda hesapların yanıltıcı olmasının sebebi, iki kenar arasındaki yumuşak geçiş yani kenar doğruları arasındaki açının geniş açılmasıdır ( $\sim 180^\circ$ ).

Bu durumda köşe gözeğine ilişkin kütle – zemin oranı, sayısal doğrulardan oluşturulan çokgenlere ait kenarların doğal olarak gürültülü bir yapıya sahip olmasının da etkisiyle, kenar gözeklerine ilişkin oran değerlerine yakın bulunmaktadır. Buna güzel bir örnek Şekil 2.5'teki dörtgenin 48. sınır gözeğinin köşe noktası olduğu halde kenar noktası karakteri gösterme eğiliminde olması sebebi ile oran değerinin yüksek çıkmasına neden olmaktadır. Bununla birlikte köşelerin belirgin olduğu durumlarda yöntem kabul edilebilir bir hata oranı ile çalışmakta ve iyi sonuçlar vermektedir.

### 2.5.3. Göreceli Uzaklıklar Yöntemi

Bu yöntemde sınır gözekleri boyunca sırayla seçilen her etkin gözekten birbirine zıt yönde  $m$  gözek ötedeki iki gözek ele alınır. Bu iki gözek arasındaki Öklid uzaklığı ölçülür. Eğer ölçülen bu uzaklık, etkin gözek ve diğer gözeklerden her üçü de aynı kenar üzerindeyse (Şekil 2.10(a) ve (d)) yaklaşık  $2m$  kadar olacaktır. Ancak etkin gözek köşe noktasına yaklaştıkça köşenin oluşturduğu açının etkisiyle belirlenen gözekler birbirine yaklaşacak ve dolayısıyla bu uzaklık azalacaktır (Şekil 2.10(b) ve (e)). Uzaklığın en az olduğu, yani belirlenen iki gözeğin birbirine en yakın olduğu nokta ise olası köşe noktası olarak kabul edilecektir (Şekil 2.10(c)). Aşağıda nesnenin sınırlarına ait her bir gözek için hesaplanan göreceli uzaklık,

$$d_n = \sqrt{(x_{n-m} - x_{n+m})^2 + (y_{n-m} - y_{n+m})^2} \quad (18)$$

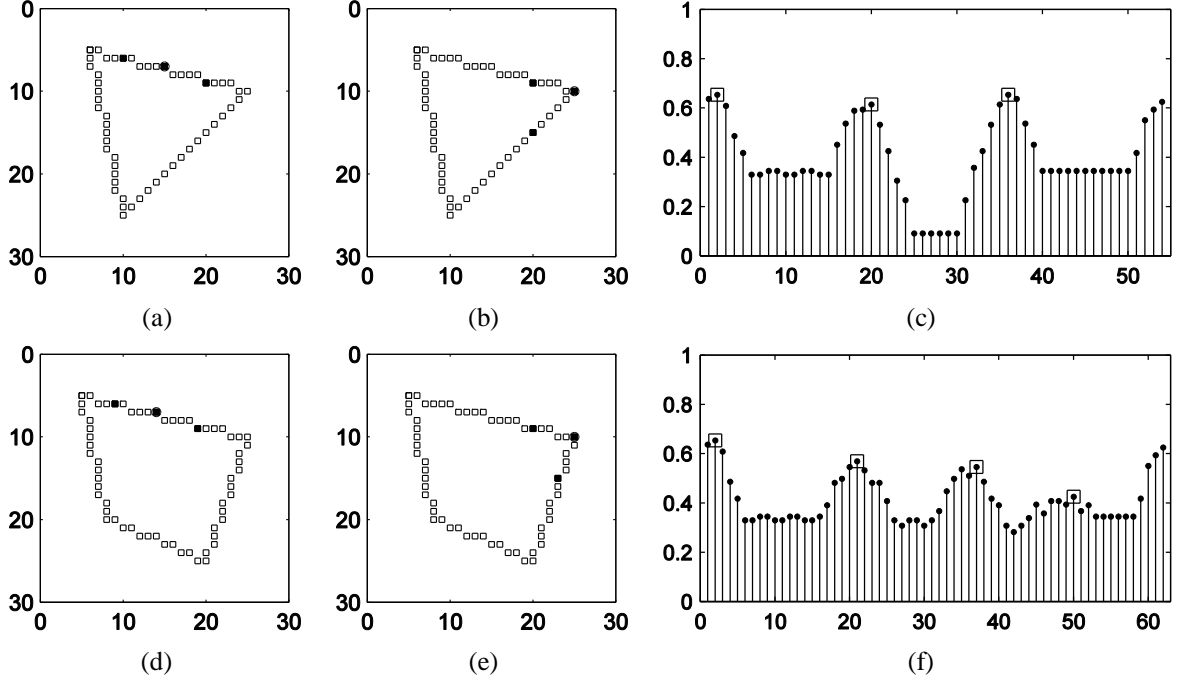
eşitliği ile verilirken, anlamlılık fonksiyonu (19) eşitliği ile verilmiştir.

$$e_n = 1 - \frac{d_n}{\sqrt{2M}} \quad (19)$$

Burada  $x_{n-m}$  etkin gözeğin  $m$  kadar gerisindeki,  $x_{n+m}$  ise  $m$  kadar ilerisindeki gözeğin koordinat değerlerini göstermektedir.  $d_n$  söz konusu koordinatlardaki iki gözek arasındaki uzaklığın değerini vermektedir.  $m$  işleme katılan komşuluk derecesidir.

$e_n$ , göreceli uzaklık yöntemi için anlamlılık fonksiyon değerlerinin (0,1) arasına çekilmesi için tanımlanan düzeltilmiş katsayıdır.

Şekil 2.10'da iki farklı yapay çokgen üzerinde yapılan denemeler gösterilmektedir. Şekil 2.10 (a) ve (b)'de yapay üçgenin sınır noktaları üzerindeki etkin gözüklerin iki farklı konumlanması verilmiştir.



Şekil 2.10. Sınır gözüklerinin göreceli uzaklık ölçütü, (a) yapay üçgenin kenarından seçilmiş bir etkin gözük ve zıt yönlerde  $m$  gözük uzaklığında belirlenmiş iki gözük; (b) yapay üçgenin köşesinden seçilmiş bir etkin gözük ve zıt yönlerde  $m$  gözük uzaklığında iki gözük; (c) yapay üçgene ait tüm etkin sınır gözükleri için göreceli uzaklık değerleri grafiği; (d) yapay dörtgenin kenarından seçilmiş bir etkin gözük ve zıt yönlerde  $m$  gözük uzaklığında iki gözük; (e) yapay dörtgenin köşesinden seçilmiş bir etkin gözük ve zıt yönlerde  $m$  gözük uzaklığında iki gözük; (f) yapay dörtgene ait tüm etkin sınır gözükleri için göreceli uzaklık değerleri grafiği.

Sırasıyla birinde etkin gözük kenar gözeği iken diğerinde köşe gözeğidir. Şekil 2.10(c) ile hesaplanan  $e_n$ ,  $n = 1, 2, \dots, N$  anlamlılık değerleri için grafik verilmiştir.

Burada yatay eksenindeki 1., 20. ve 36. noktalar zıt yönlerde belirlenmiş gözükler arası uzaklıkların en düşük olduğu üç noktayı oluşturmaktadır. Bu noktalar aynı zamanda baskın gözük konumlarına karşılık gelmektedir ve ilk iki nokta doğru şekilde bulunmuş, diğer nokta ise 1 sapma ile doğru noktaya en yakın nokta olarak bulunmuştur. Yöntemin bu şekil için oldukça başarılı çalıştığı söylenebilir.



Benzer şekilde, Şekil 2.10 (d) ve (e)'de yapay dörtgenler üzerindeki etkin gözeklerin iki farklı konumlanması verilmiştir. Sırasıyla birinde etkin gözek kenar gözeği, diğerinde ise köşe gözeğidir. Şekil 2.10 (f) ile de ilgili dörtgenin sınır gözekleri için hesaplanan  $e_n$ ,  $n = 1, 2, \dots, N$  değerleri için grafik verilmiştir. Burada da 1., 21. ve 37.ve 50. noktalar baskın noktalar olarak bulunmuştur. İlk iki nokta en doğru şekilde bulunmuş, diğer noktalar ise sırasıyla 1 ve 2 sapma ile doğru noktalara çok yakın noktalar olarak bulunmuştur. Yöntemin kütle – zemin oranı yönteminin aksine bu şekil için oldukça başarılı çalıştığı söylenebilir.

Yerel yöntemlerden biri olan göreceli uzaklık yöntemi kenarlar arası yumuşak geçişlerde dahi köşe noktalarını algılamakta sorun oluşturmamaktadır. Çünkü iki kenar arasındaki yumuşak geçiş olması durumunda dahi köşe gözeğine ilişkin göreceli uzaklık değeri, etkin gözeğin kenar gözeklerinden biri olduğu durumdakinden yeterli derecede farklı (belirlenebilecek kadar) olarak hesaplanmaktadır. Fakat özgün sayısal görüntülerdeki kenar doğrularının çok fazla gürültülü olduğu durumlarda hata oranı artabilmektedir. Bununla birlikte kenarların temiz olduğu durumlarda yöntem kabul edilebilir bir hata oranı ile çalışmakta ve iyi sonuçlar vermektedir.

#### 2.5.4. Kesişim Açısı Yöntemi

Kesişim Açısı yönteminde açı değerlerinden yararlanılır. Diğer yöntemlerde olduğu gibi sınır gözeklerinin her biri için bir açı değeri hesaplanır. Etkin gözeğin zıt yönlerindeki  $m$  kadar gözeğin konumlarının oluşturduğu iki veri kümesinden hesaplanan iki bağlantım (regression) doğrusunun kesişim açısı baskın noktanın belirlenmesinde bir ölçüt olarak alınır. Eğer etkin gözek kenarlık gözeği ise açı  $\theta \approx 0^\circ$  (veya  $\theta < a$ ) derece civarında (Şekil 2.11(a) ve (d)), köşe noktası civarındaki bir gözek ise  $a \leq \theta \leq 90^\circ$  aralığında olacaktır (Şekil 2.11(b) ve (e)). Bu iki bağlantım doğrusunun kesişim açısı (20)'de verilir.

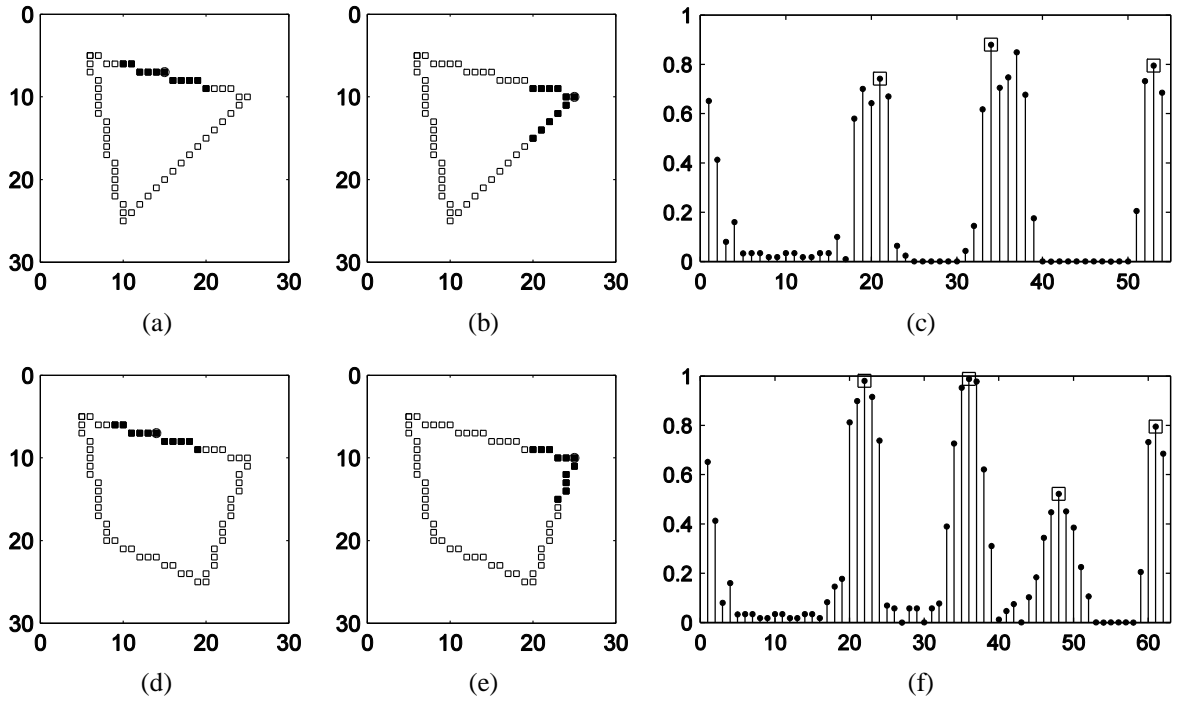
$$\theta_n = \text{atan} \left( \frac{m_{n(+)} - m_{n(-)}}{1 + m_{n(+)} m_{n(-)}} \right) \quad (20)$$

Bu açının anlamlılık ölçüsü (21) ile verilen eşitlik ile bulunur.

$$e_n = \frac{2}{\pi} |\theta_n| \quad (21)$$

Burada  $m_{n^{(+)}}$  ve  $m_{n^{(-)}}$  sırasıyla birinci ve ikinci bağlantım doğrusunun eğimleridir. Yukarıda anılan  $a$  değeri anlamlılık düzeyi olarak kabul edilir. (20) eşitliği kullanılarak her etkin sınır gözeği için kesişim açısı hesaplanır, istenen sayıda köşe için en yüksek  $e_n$  değerine ilişkin sınır gözekleri köşe gözekleri olarak belirlenir (Şekil 2.11(c)).

Şekil 2.11’de kesişim açısının uygulaması olarak iki farklı yapay çokgen üzerinde yapılan denemeler gösterilmektedir. Şekil 2.11 (a) ve (b)’de yapay üçgenin sınır noktaları üzerindeki etkin gözeklerin iki farklı konumlanması verilmiştir. Sırasıyla birinde etkin gözük kenar gözeği iken diğesinde köşe gözeğidir. Şekil 2.11(c) ile hesaplanan  $e_n$ ,  $n = 1, 2, \dots, N$  değerleri için anlamlılık grafiği verilmiştir.



Şekil 2.11. Etkin gözekten geçen iki bağlantım doğrusu arasındaki açının belirlenmesi, (a) yapay üçgene ait kenardan seçilmiş etkin gözeğin zıt yönlerinde  $m$  tane gözeğin oluşturduğu iki değerler kümesi; (b) yapay üçgene ait köşeden seçilmiş etkin gözeğin zıt yönlerinde  $m$  tane gözeğin oluşturduğu iki değer kümesi; (c) yapay üçgenin tüm etkin gözükleri için bağlantım doğrularının kesişim açıları; (d) yapay dörtgene ait kenardan seçilmiş etkin gözeğin zıt yönlerinde  $m$  tane gözeğin oluşturduğu iki değerler kümesi; (e) yapay dörtgene ait köşeden seçilmiş etkin gözeğin zıt yönlerinde  $m$  tane gözeğin oluşturduğu iki değer kümesi; (f) yapay dörtgenin tüm etkin gözükleri için bağlantım doğrularının kesişim açıları.

Burada 53., 21. ve 34. noktaları bu noktalardaki gözeklerin iki yanındaki kesişen doğrular arasındaki açılar hesaplanan diğer açı değerleri arasında en düşük değeri veren üç noktadır. Bu noktalar aynı zamanda baskın gözek konumlarına karşılık gelmektedir ve ilk nokta  $-2$  sapma ile doğru sayılabilecek bir nokta, diğer noktalar ise  $+1$  sapma ile doğru noktaya en yakın noktalar olarak bulunmuştur. Kesişim açıları yönteminin de diğer iki yöntem gibi bu şekil için oldukça başarılı çalıştığı söylenebilir.

Benzer şekilde, Şekil 2.11 (d) ve (e)'de yapay dörtgen üzerindeki etkin gözeklerin iki farklı konumlanması verilmiştir. Sırasıyla birinde etkin gözek kenar gözeği, diğerinde ise köşe gözeğidir. Şekil 2.11 (f) ile de ilgili dörtgenin sınır gözekleri için hesaplanan  $e_n$ ,  $n = 1, 2, \dots, N$  değerleri için anlamlılık grafiği verilmiştir. Burada da 61., 22., 36. ve 48. noktalar baskın noktalar olarak bulunmuştur. İlk nokta  $-2$ 'lik bir sapma ile doğru sayılabilecek bir nokta, ikinci nokta  $+1$  birimlik bir sapma ile doğruya en yakın nokta, diğer noktalar ise sapmasız tam doğru noktalar olarak bulunmuştur. Kesişim açısı yönteminin de bu şekil için oldukça başarılı çalıştığı söylenebilir.

Yerel yöntemlerden biri olan kesişim açısı yöntemi köşelerin belirgin olduğu yapay üçgende doğal olarak başarılıdır; bununla birlikte yöntem ardışık kenarlara ilişkin doğruların geniş açı yaptığı durumlarda dahi köşe noktalarını algılamakta sorun oluşturmamaktadır. Çünkü yöntem hesaplamada kullanılan eşitliğin tanımı gereği sürekli doğrular için açı değerleri üretmektedir. Dolayısıyla sayısal görüntülerdeki gürültülerden en az etkilenen yöntemlerden biridir. Ne var ki yöntem özgün sayısal görüntülerde yapay çokgenlerin aksine kenar doğusu sayılabilecek sayısal doğruların, görüntünün geneline oranla çok kısa olduğu durumlarda tarife uyan bir kenarın tam ortasındaki bir gözeği de köşe noktası olarak algılayabilme eğilimindedir. Ancak kenarların çok kısa olmayıp uygun uzunluklarda olduğu durumlarda yöntem kabul edilebilir bir hata oranı ile çalışmakta ve iyi sonuçlar vermektedir.

### 2.5.5. Belirtme Katsayısı Yöntemi

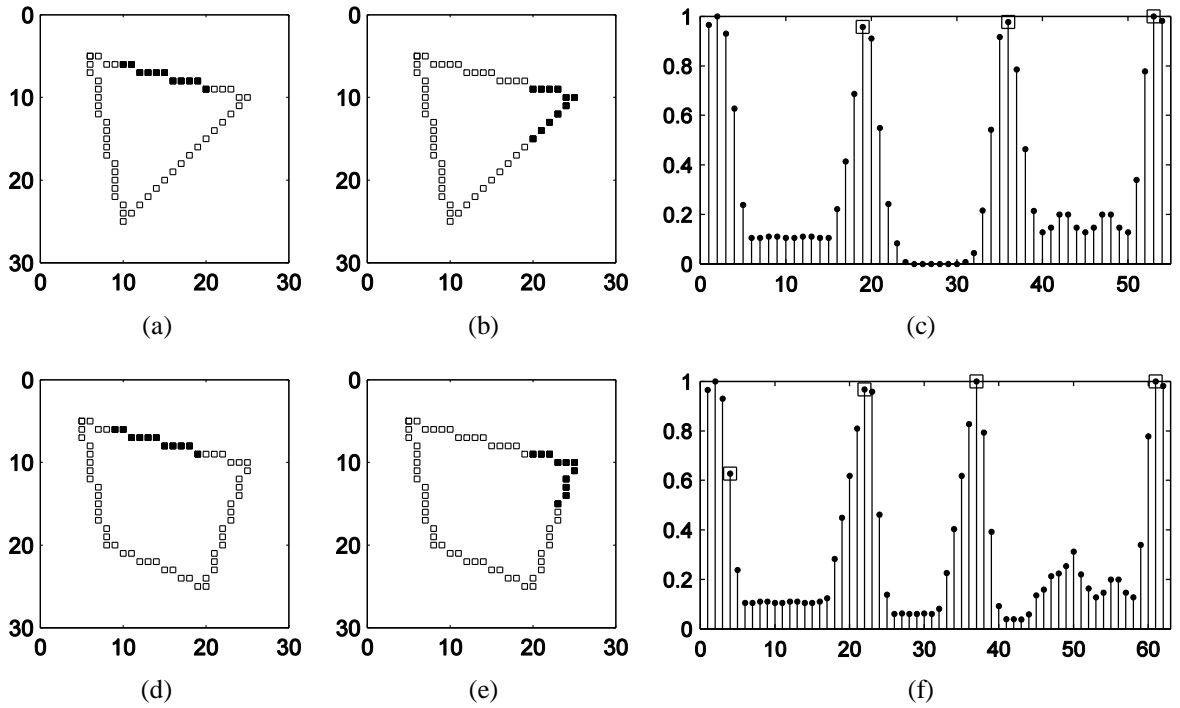
Kesişim açısı yöntemindeki iki bağlanım doğrusu yerine tek bağlanım doğrusu seçilir. Seçilen bu noktalardan belirtme katsayısı hesaplanır. İlgili gözeklerin doğru etrafında sapmaları düşükse düz bir kenar olduğu (Şekil 2.12(a), (d)), sapmaları yüksekse, yani köşelere yaklaşırsa (Şekil 2.12(b), (e)) baskın noktaya yaklaşıldığı söylenebilir (Şekil 2.12(c), (f)). Belirtme katsayısı (22)'de verilmiştir.

$$r_n^2 = \frac{(\sum_{k=n-m}^{n+m} (x_k - \bar{x}_n)(y_k - \bar{y}_n))^2}{\sum_{k=n-m}^{n+m} (x_k - \bar{x}_n)^2 \sum_{k=n-m}^{n+m} (y_k - \bar{y}_n)^2} \quad (22)$$

Yöntemin başarısını ölçmek için belirtme katsayısından yararlanarak,

$$e_n = 1 - r_n^2$$

eşitliği elde edilir. Bu eşitlikteki  $e_n$  anlamlılık fonksiyonu olarak adlandırılır.



Şekil 2.12. Etkin göze merkezli  $2m$  sayıda gözeğin oluşturduğu kümenin belirtme katsayıları, (a) yapay üçgene ait kenardan seçilmiş etkin gözeğin zıt yönlerindeki  $m$  tane gözeğin oluşturduğu değerler kümesi; (b) yapay üçgene ait köşeden seçilmiş etkin gözeğin zıt yönlerindeki  $m$  tane gözeğin oluşturduğu değer kümesi; (c) yapay üçgenin tüm etkin gözleri için bağlantım doğrularından elde edilen  $e_n$  değerlerinin grafiği; (d) yapay dörtgene ait kenardan seçilmiş etkin gözeğin zıt yönlerindeki  $m$  tane gözeğin oluşturduğu değerler kümesi; (e) yapay dörtgene ait köşeden seçilmiş etkin gözeğin zıt yönlerindeki  $m$  tane gözeğin oluşturduğu değer kümesi; (f) yapay dörtgenin tüm etkin gözleri için bağlantım doğrularından elde edilen  $e_n$  değerlerinin grafiği.

Şekil 2.12’de kesişim açısının uygulaması olarak iki farklı yapay çokgen üzerinde yapılan denemeler gösterilmektedir.

Şekil 2.12 (a) ve (b)'de yapay üçgenin sınır noktaları üzerindeki etkin gözleklerin iki farklı konumlanması verilmiştir. Sırasıyla birinde etkin gözlek kenar gözeği iken diğerinde köşe gözeğidir. Şekil 2.12(c) ile hesaplanan  $e_n$ ,  $n = 1, 2, \dots, N$  değerleri için anlamlılık grafiği verilmiştir. Burada 53., 19. ve 36. noktaları bu noktalardaki gözleklerin iki yanından seçilen noktalardan geçen doğrulardan hesaplanan bağlanım fonksiyonuna ilişkin anlamlılık değerleri arasında en düşük değeri veren üç noktadır. Bu noktalar aynı zamanda baskın gözlek konumlarına karşılık gelmektedir ve ilk nokta  $-2$  sapma ile doğru sayılabilecek bir nokta, diğer noktalar ise  $-1$  sapma ile doğru noktaya en yakın noktalar olarak bulunmuştur. Belirtme katsayısı yönteminin de diğer iki yöntem gibi bu şekil için oldukça başarılı çalıştığı söylenebilir.

Benzer şekilde, Şekil 2.12 (d) ve (e)'de yapay dörtgen üzerindeki etkin gözleklerin iki farklı konumlanması verilmiştir. Sırasıyla birinde etkin gözlek kenar gözeği, diğerinde ise köşe gözeğidir. Şekil 2.12 (f) ile de ilgili dörtgenin sınır gözlekleri için hesaplanan  $e_n$ ,  $n = 1, 2, \dots, N$  değerleri için anlamlılık grafiği verilmiştir. Burada da 4., 21., 37. ve 61. noktalar baskın noktalar olarak bulunmuştur. İlk nokta  $+3$ 'lük bir sapma ile doğru sayılabilecek bir nokta, ikinci nokta sapmasız tam doğru bir nokta, üçüncü nokta  $+1$  birimlik bir sapma ile doğruya en yakın nokta olarak bulunmuştur. Diğer nokta ise ilk köşe gözeği civarındaki bir nokta olarak bulunmuştur. Bu durum istenmeyen bir durumdur. Hatalı belirlemenin olası nedeni kenarlar arasındaki yumuşak geçiş olarak öngörülebilir. Yani algoritma köşe bölgesi keskin olmadığından dolayı söz konusu bölgeyi kenar gibi algılamıştır. Belirtme katsayısı yönteminin bu gibi şekiller için yeterince başarılı çalıştığı söylenemez.

## 2.6. Genel Anlamda Baskın Nokta Algılama Yöntemleri

Yerel anlamda baskın nokta algılaması, etkin gözlek ile komşuları arasındaki ilişkinin derecesine göre belirlenen bir yöntemdir. Ancak bu yöntemler genel anlamda bakıldığında nesneye çokgensel yaklaşımda amaç fonksiyonunun iyiliğini vermekte yetersiz kalmaktadır.

### 2.6.1. Kesin Çözüm

Bir nesnenin istenen baskın nokta sayısına ( $K$ ) göre sınır gözeklerinin tüm kombinasyonları denenerek en küçük hatayı verecek çokgen yaklaşımında kesin çözüm kullanılabilir. Şayet nesne küçük ve istenen köşe sayısı da az ise kesin çözüm en iyi sonucu verecektir. Ancak nesnenin sınır gözeklerinin sayısı çok ve istenen baskın nokta sayısı fazla ise çözümün bulunma zamanı üstel olarak artacaktır. Kesin çözümün neredeyse imkansız olduğu sonsuz çözüm uzayında sezgisel algoritmalar kullanılmaktadır.

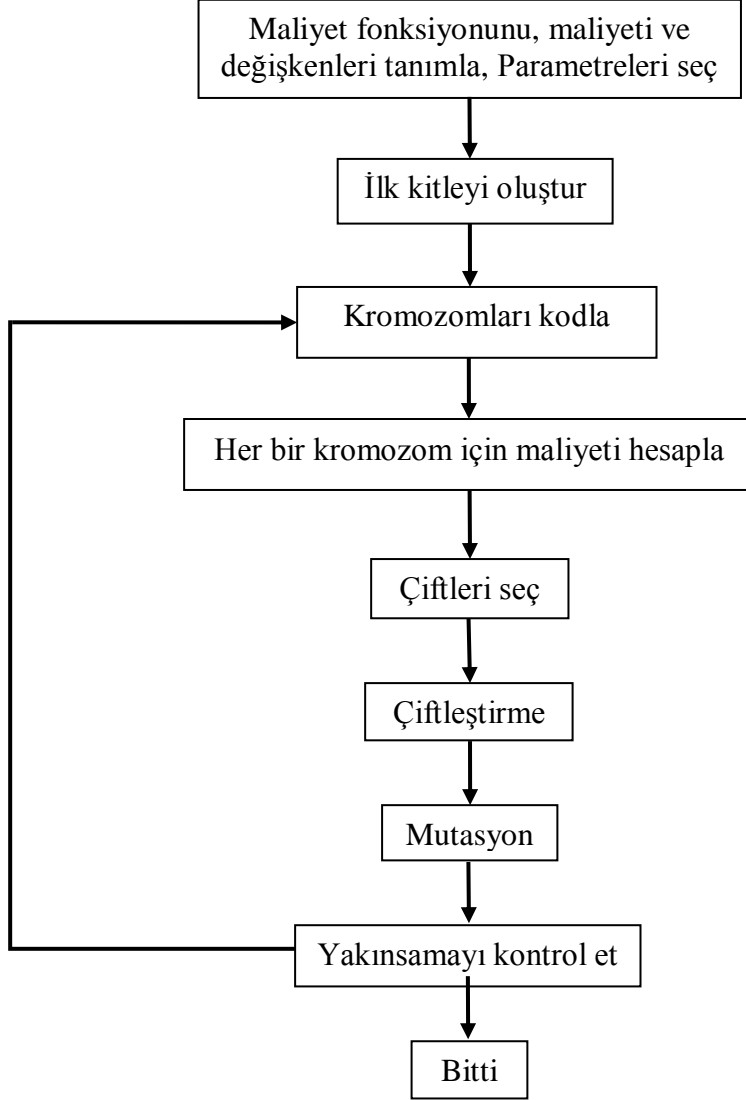
### 2.6.2. Baskın Nokta Algılanmasında Genetik Algoritma

Genetik algoritma (GA) genetiğin doğal seçim konusunun temel prensiplerine dayanan bir optimizasyon ve arama tekniğidir. Bu yöntemi ilk olarak 1975'te Michigan Üniversitesinden John Holland tarafından geliştirildi ve önerildi (Holland, 1975). GA belirli seçim kuralları altında, bir kitleyi oluşturan bireylerin, ulaşılmak istenen durum için uyumu büyüktmesini veya maliyeti küçüktmesini sağlar. Genetik algoritmaların klasik optimizasyon yöntemlerine göre sayısız üstünlüğü vardır. Bu üstünlükleri sayesinde diğer klasik yöntemlere göre açık farkla çok iyi sonuçlar üretmektedir. Bu durum her türlü problemin genetik algoritmalar ile tam sonuca ulaşabileceği anlamına gelmemekle birlikte yöntemin genellikle iyi sonuçlar verdiği bilinmektedir.

Genetik algoritmalar da değişkenlerin ve maliyet fonksiyonunun tanımlanmasıyla, diğer optimizasyon algoritmaları gibi başlar. Yakınsama sürecine ilişkin testlerin yapılmasıyla da diğer tekniklere benzer şekilde sonlanır. Ne var ki algoritma, işleyişi açısından oldukça farklıdır. Genetik algoritmanın adımlarının gösterildiği akış şeması Şekil 2.13'de verilmiştir. Genel anlamda maliyet fonksiyonu değişken uzayında tanımlı, tepeler ve çukurlar içeren bir yüzey alanıdır. Çukurları (yerel minimum noktalarını) bulmak isteyen optimizasyon algoritması minimum maliyeti, tersi şekilde de maksimum kazancı araştıracaktır.

Bir maliyet fonksiyonu girdi değerleri kümesindeki her kromozom için bir maliyet değeri üretir. Maliyet fonksiyonu matematiksel bir fonksiyon, bir deney ya da bir oyun olabilir. Amaç girdi değişkenleri için uygun değerleri bularak çıkış değerlerini arzu edilen şekilde değiştirmektir.

Uyumluluk değeri genetik algoritma literatüründe genellikle amaç fonksiyonunun türünü belirtmek için kullanılır. Bu çalışma için kullanılacak maliyet uyumluluk değeri 2.4. bölümde verilmiştir.



Şekil 2.13. İkili genetik algoritmanın akış şeması.

Baskın nokta algılamasında kullanılan genetik algoritma aşağıdaki biçimde gerçekleşmektedir:

**Adım 1:** Girdilerin belirlenmesi

İkili bir görüntüde ele alınan bir nesnenin sınır gözetleri  $(p_i(x_i, y_i) | i = 1, 2, \dots, N)$  bulunarak genetik algoritmaya girdi olarak verilir. Diğer bir girdi ise istenilen baskın nokta sayısıdır (K).

**Adım 2:** Maliyet fonksiyonu ve Uyumluluk ölçütünün tanımlanması

Maliyet fonksiyonu ve uyumluluk ölçütü başlık 2.4'de verilmiştir.

**Adım 3:** Başlangıç Kitesinin Oluşturulması

Başlangıç kitesinin oluşturulması için öncelikle baskın nokta probleminin kromozom yapısının tanımlanması gerekir. Kromozom,  $b_{p_1}, b_{p_2}, \dots, b_{p_N}$  şeklinde  $N$  değişkenden oluşmaktadır. Dolayısıyla bu problem  $N$ -boyutlu bir optimizasyon problemidir ve  $N$  birimlik satır vektörü şeklinde yazılır,

$$Kromozom = [b_{p_1}, b_{p_2}, \dots, b_{p_N}]. \quad (23)$$

Buradaki her  $p_i$  değeri sınır gözeleklerini göstermektedir.  $b_{p_i} \in \{0,1\}$  değeri ise her  $p_i$  sınır gözeğinin kromozom değerini vermektedir.  $b_{p_i}$  değerinin 0 olması o sınır gözeğin kenar olduğunu, 1 olması durumunda ise o gözeğin baskın nokta olduğunu gösterir. Örneğin, sınır gözek sayısı  $N = 10$  olan bir nesnenin 2., 6. ve 8. sınır gözeği baskın nokta ise kromozom dizilimi,

$$Kromozom = [0 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0] \quad (24)$$

biçiminde verilir. Bu gösterim tarzına ikil gösterim denir ve genetik algoritmalarda sıklıkla tercih edilen bir gösterimdir. Buradaki her 0 veya 1 değerine *gen* adı verilir. Kromozom dizisindeki 1'lerin sayısının istenen baskın nokta sayısına eşit olması gerekir. Toplamda kromozom dizisinin boyu  $N$  olmalıdır.  $N$  tane sınır gözeği için  $N$  tane değerden oluşan bir kromozom tanımlanır ve bu dizinin içinde rastgele konumlanmış  $K$  tane 1 değeri bulunur. Bu şekilde  $M$  tane farklı kromozom rastgele oluşturularak başlangıç kitlesi elde edilir.

**Adım 4:** Seçim

Çiftleşme havuzuna girecek kromozomların seçilimini gerçekleştirmek için öncelikle  $M$  kromozomun her biri için maliyet fonksiyon değerinin hesaplanması gerekir. Maliyet fonksiyonun hesaplanması için her kromozomun baskın noktalarının oluşturduğu yapay bir çokgen ile ilgilenilen çokgen arasındaki farkın mutlak değerlerinin toplamını almak gerekir. Her kromozom için elde edilen maliyet değeri temel alınarak, maliyet değerine göre kromozom küçükten büyüğe sıraya dizilir. Tüm kitlenin ilk %50 kromozomu çiftleşme havuzuna atılır.



**Adım 5: Çaprazlama**

Çiftleşme havuzundaki her kromozom kendisine rastgele bir eş seçer. Her çiftin çaprazlama sonucunda iki yavrusu oluşur. Çaprazlama işlemi, ilk olarak rastgele bir çaprazlama noktası seçilmesiyle başlar. Bu nokta,

$$s = U[2, N - 1] \quad (25)$$

eşitliğiyle verilir. Burada  $s$ ,  $[2, N - 1]$  aralığında düzgün dağılıma sahip rastgele bir tamsayıdır. Çiftlerden birinci kromozomun  $s$  çaprazlama noktasına kadar olan ilk kısmı birinci yavruya olduğu gibi geçer. Birinci yavru geri kalan genlerini ikinci kromozomun çaprazlama noktasından sonraki kısmından alır. Aynı şekilde, ikinci kromozomun birinci kısmı ikinci yavruya aktarılırken, birinci kromozomun ikinci kısmı da ikinci yavrunun ikinci kısmına aktarılarak ikinci yavru oluşturulur. Her kromozom çifti için bu işlemler tekrarlanır. Bu işlemin sonucunda havuzda bulunan kromozom sayısı kadar yeni kromozom oluşmuş olur. Ancak yeni oluşan kromozomların baskın noktayı temsil eden 1 genleri istenen baskın noktadan fazla veya eksik çıkabilir. Bu durumda bir onarma işlemi gerçekleşir. Onarım için yeni oluşan yavru çiftler arasında ihtiyaç duyulan kadar sayıda rastgele gen alış verişi yapılarak 1 genlerinin sayıları her iki yavru için eşitlenmiş olur.

**Adım 6: Mutasyon**

GA yöntemi çoğu zaman yerel minimumlara takılıp kalır. Yerel minimumdan kurtulmak için yeni oluşan kromozomların %5'ine mutasyon işlemi uygulanır. Mutasyon işlemi rastgele seçilen bir kromozomun farklı değerli herhangi iki geninin yer değiştirmesiyle gerçekleştirilir.

**Adım 7: Yakınsama Denetimi**

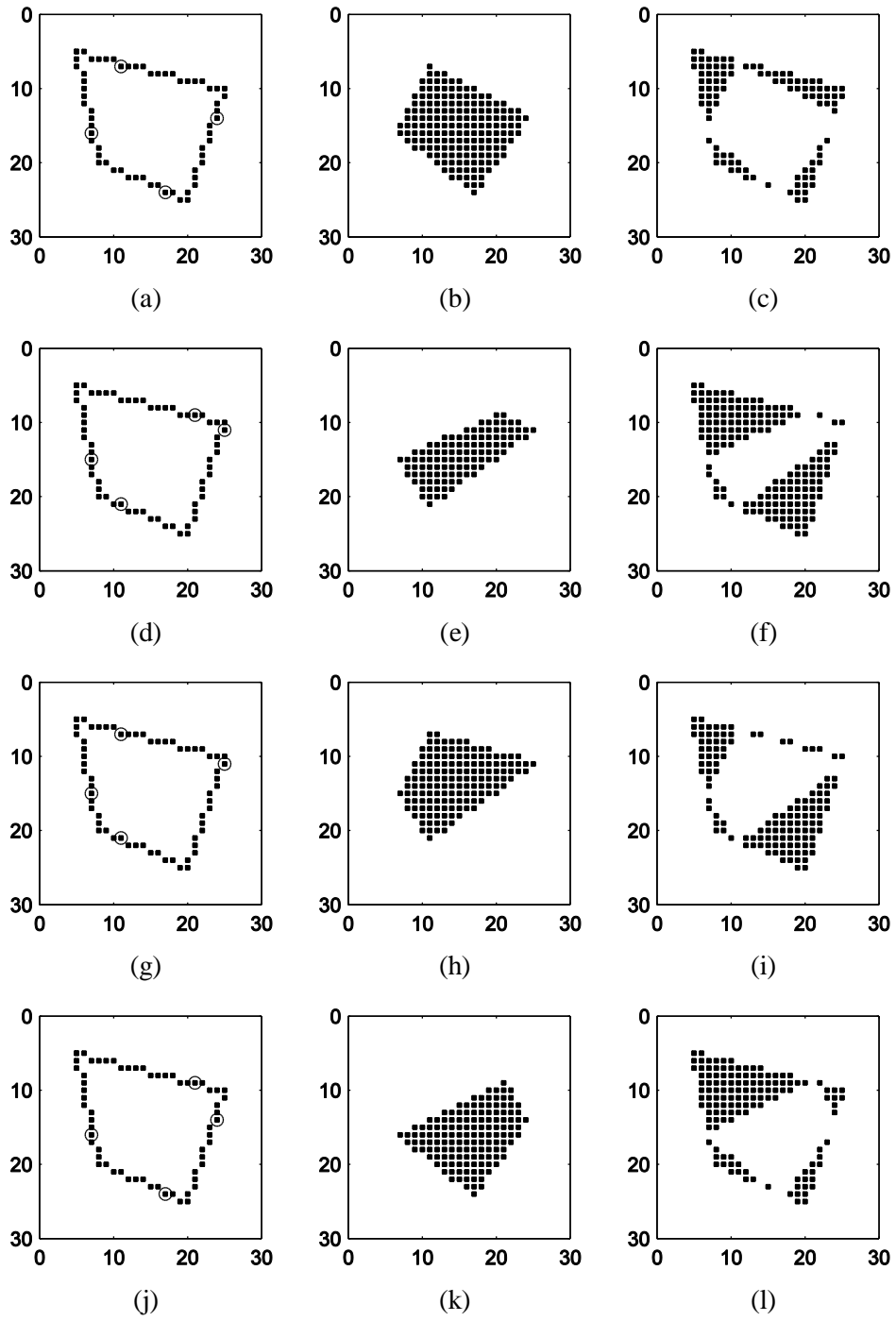
Yeni nesil kitlenin içindeki herhangi bir kromozomun maliyet değeri sıfırsa veya nesil dönemi istenen döneme eşitse işlem sonlandırılır. Eğer değilse Adım 4'e giderek yeni nesil dönemini başlatılır.

Yukarıdaki adımların tamamlanmasıyla problemin çözümü gerçekleşmiş olur. Şekil 2.14'te yerel yöntemlerin genellikle başarılı olamadığı yapay dörtgen şekil üzerinde genetik algoritmanın uygulanışı gösterilmiştir. Yerel yöntemlerin aksine GA amaç fonksiyonuna en iyi değeri vermeyi amaçladığından baskın nokta olarak en doğru noktaları

bulana kadar araştırır. Böylece köşelerin keskin ya da yumuşak geçişli olması baskın noktaların belirlenmesi için engel oluşturmayacaktır. Zira oluşturulan kitleler içinde nesne sınırlarının hemen hemen tüm noktalarına şans verilir. Bu şekilde iterasyonların gerçekleşmesi sonucunda beklenen en az hata oranına ulaşılması ise kaçınılmazdır. Fakat belirtmek gerekir ki kromozomların rastgele oluşturulması sebebiyle ve çaprazlama noktasının da rastgele olarak seçilmesinden dolayı her zaman az da olsa hata yapma ve hiç beklenmeyen sonuçlarla karşılaşma olasılığı söz konusudur.

Şekil 2.14(a)'da ilk oluşturulan kitlede maliyet fonksiyonuna en iyi değeri veren kromozoma ilişkin baskın noktalar, (b)'de bu baskın noktalardan elde edilen  $J$  görüntüsü ve (c)'de ise bu çözüme ilişkin maliyet değeri gözeysel alan olarak görülmektedir. Şekil 2.14(d)'de ilk oluşturulan kitlede maliyet fonksiyonuna ikinci en iyi değeri veren kromozoma ilişkin baskın noktalar, (e)'de bu baskın noktalardan elde edilen  $J$  görüntüsü ve (f)'de ise bu çözüme ilişkin maliyet değeri gözeysel alan olarak görülmektedir. Şekil 2.14(g)'da ilk oluşturulan kitlede maliyet fonksiyonuna en iyi değeri veren ilk iki kromozomun çiftleştirilmesi sonucu oluşan iki yavrudan birine ilişkin baskın noktalar, (h)'de yavrunun baskın noktalarına ilişkin  $J$  görüntüsü ve (i)'de ise bu çözüme ilişkin maliyet değeri gözeysel alan olarak görülmektedir. Şekil 2.14(j)'de ilk oluşturulan kitlede maliyet fonksiyonuna en iyi değeri veren ilk iki kromozomun çiftleştirilmesi sonucu oluşan iki yavrudan diğerine ilişkin baskın noktalar, (k)'de yavrunun baskın noktalarına ilişkin  $J$  görüntüsü ve (l)'de ise bu çözüme ilişkin maliyet değeri gözeysel alan olarak görülmektedir.

Bu şekille gösterilen çözümler GA'nın ilk iterasyonu sonucu elde edilen çözümlerdir. İterasyonların tekrarlanmasıyla (c), (f), (i), (l) ile gösterilen maliyet değeri azalacaktır.



Şekil 2.14. Yapay dörtgen nesnesine Genetik Algoritma uygulaması, (a) birinci ebeveyn çözümü; (b) birinci ebeveyn çözümüne ilişkin  $J$  görüntüsü; (c) maliyet fonksiyonunun değeri; (d) ikinci ebeveyn çözümü; (e) ikinci ebeveyn çözümüne ilişkin  $J$  görüntüsü; (f) maliyet fonksiyonunun değeri; (g) birinci yavru çözümü; (h) birinci yavru çözümüne ilişkin  $J$  görüntüsü; (i) maliyet fonksiyonunun değeri; (j) ikinci yavru çözümü; (k) ikinci yavru çözümüne ilişkin  $J$  görüntüsü; (l) maliyet fonksiyonunun değeri.

### 2.6.3. Baskın Nokta Algılanmasında Tabu Arama Algoritması

Tabu Arama (TA) algoritması genel olarak basit ve karmaşık tamsayılı programlama problemlerinin çözümünde kullanılan bir stratejiler bütünüdür. Doğrusal programlama algoritmaları ve yerel optimalliğin sınırlarını araştıran sezgisel algoritmalar gibi farklı yaklaşımlara da uyarlanabilen bir yöntemdir (Glover, 1989). TA algoritması genel olarak aşağıdaki adımlardan oluşur:

#### Adım 1: Başlangıç Durumu Oluşturma

N tane sınır gözeğinden sadece K tanesi baskın nokta olmaktadır.  $\binom{N}{K}$  tane farklı çözümün içerisinde, rastgele bir çözüm seçilir. Seçilen bu çözümün başlık 2.4'de verilen maliyet fonksiyonuna göre maliyet değeri hesaplanır. Bu çözüm aynı zamanda *Yasak Listesine* kayıt edilir.

#### Adım 2: Komşuların Araştırılması

Etkin çözüm, N tane sınır gözeğinde farklı konumlara yerleşmiş K tane baskın noktanın oluşturduğu bir çözümdür. Bu çözümde her noktanın önünde ve arkasındaki nokta o noktanın bir komşusu olmaktadır. Aynı şekilde, tüm baskın noktaların önünde ve gerisindeki noktalar bir komşu çözümü olarak düşünülürse toplam  $3^K - 1$  tane komşu çözüm bulunur. Her çözümün maliyet değeri hesaplanır.

#### Adım 3: Yeni Etkin Çözümün Belirlenmesi

Komşu çözümlerin maliyet değerlerine göre en iyi çözümü veren komşu seçilerek etkin çözüm ile karşılaştırılır. En iyi çözüm etkin çözümden iyi ise yeni etkin çözüm bu çözüm olacaktır. Unutulmaması gereken diğer bir konu ise bulunan en iyi her çözümün yasak listesiyle karşılaştırılarak aynı çözümün listede olup olmadığının araştırılmasıdır. Şayet en iyi çözüm yasak listesinde bulunursa, komşular içerisindeki bir sonraki en iyi çözüm etkin çözüm olarak değerlendirilmeye alınır. Seçilen en iyi çözüm ile etkin çözüm farklı ise Adım 2'ye gidilerek işlemlere devam edilir.

#### Adım 4: Yakınsama Denetimi

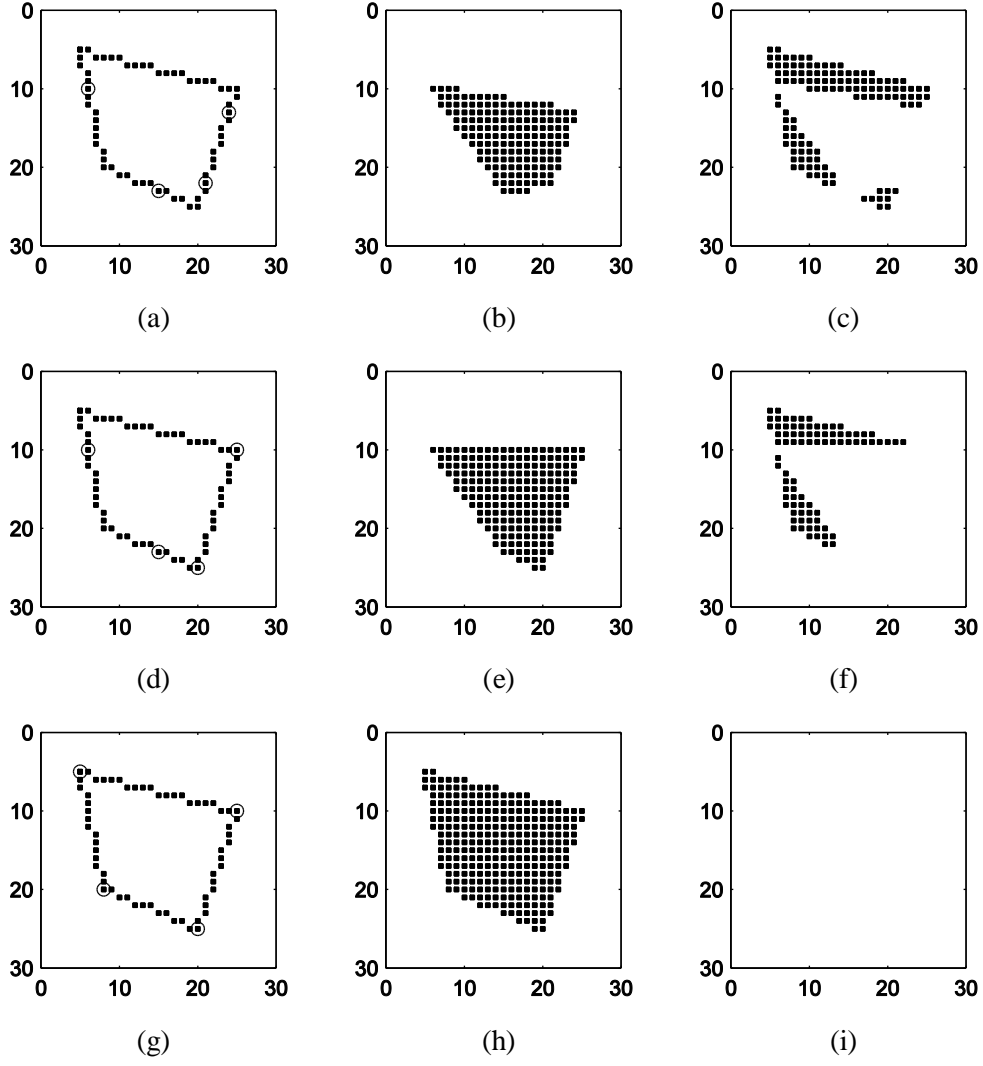
Komşu çözümler arasındaki en iyi çözüm etkin çözümden iyi değilse etkin çözüm yerel veya genel amaca ulaşmış demektir. Denemelerin sayısı tamamlandığında veya maliyet olduğunda tüm işlemler sonlandırılır ve yasak listesinde bulunan en iyi çözüm, baskın nokta çözümü olarak kabul edilir.

Eğer deneme sayısı istenen deneme sayısına eşit değilse bir sonraki denemeye geçmek için Adım 1'e gidip farklı çözümler araştırılır.

Yukarıdaki adımların tamamlanmasıyla problemin çözümü gerçekleşmiş olur. Şekil 2.15'te yerel yöntemlerin genellikle başarılı olamadığı yapay dörtgen şekil üzerinde Tabu Arama algoritmasının uygulanışı gösterilmiştir. Yerel yöntemlerin aksine Tabu Arama algoritması da GA'da olduğu gibi amaç fonksiyonuna en iyi değeri vermeyi amaçladığından baskın nokta olarak en doğru noktaları bulana kadar nesnenin sınır noktalarını araştırır. Böylece köşelerin keskin ya da yumuşak geçişli olması baskın noktaların belirlenmesi için engel oluşturmayacaktır. Burada da çözüm adımları boyunca nesne sınırlarının hemen hemen tüm noktalarının baskın nokta olup olamayacağı araştırılır. Bu şekilde tabu algoritmasında da denemelerin gerçekleşmesi sonucunda beklenen en az hata oranına ulaşılması kaçınılmazdır. Tabu Arama algoritması kesin çözüme yüksek başarı ile yaklaşan oldukça hızlı çözüm yapan bir algoritmadır. Sadece birkaç denemede çözüm sürecini tamamlayabilmekte ve çözüm tamamlandığında ise neredeyse 0 hata ile diğer tüm yöntemlerden çok daha iyi sonuçlar üretmektedir.

Şekil 2.15(a)'da Tabu Arama algoritmasının ilk çözümüne ilişkin baskın noktalar, (b)'de bu baskın noktalardan elde edilen  $J$  görüntüsü ve (c)'de ise bulunan çözüme ilişkin maliyet değeri gözeysel alan olarak görülmektedir. Şekil 2.15(d)'de algoritmanın ikinci denemesine ilişkin baskın noktalar, (e)'de bu baskın noktalardan elde edilen  $J$  görüntüsü ve (f)'de ise bulunan çözüme ilişkin maliyet değeri gözeysel alan olarak görülmektedir. Şekil 2.15(g)'de algoritmanın dördüncü denemesine ilişkin baskın noktalar, (h)'de bu baskın noktalardan elde edilen  $J$  görüntüsü ve (i)'de ise kesin çözümün gerçekleştirilmiş olduğu görülmektedir.

Bu şekille gösterilen çözümler Tabu Arama algoritmasının başarısını açık bir şekilde ortaya koymaktadır. Denemelerin tekrarlanmasıyla (c), (f), (i) ile gösterilen hata değeri azalarak çok hızlı bir şekilde 0'a inmektedir.



Şekil 2.15. Yapay dörtgen nesnesine Tabu Arama algoritmasının uygulaması, (a) başlangıç çözüm; (b) başlangıç çözümden elde edilen  $J$  görüntüsü; (c) başlangıç çözüme ilişkin maliyet değeri; (d) 2. denemedeki çözüm; (e) 2. denemedeki çözümden elde edilen  $J$  görüntüsü; (f) 2. denemedeki çözüme ilişkin maliyet değeri; (g) 4. denemedeki çözüm; (h) 4. denemedeki çözümden elde edilen  $J$  görüntüsü; (i) 4. denemedeki çözüme ilişkin maliyet değeri.

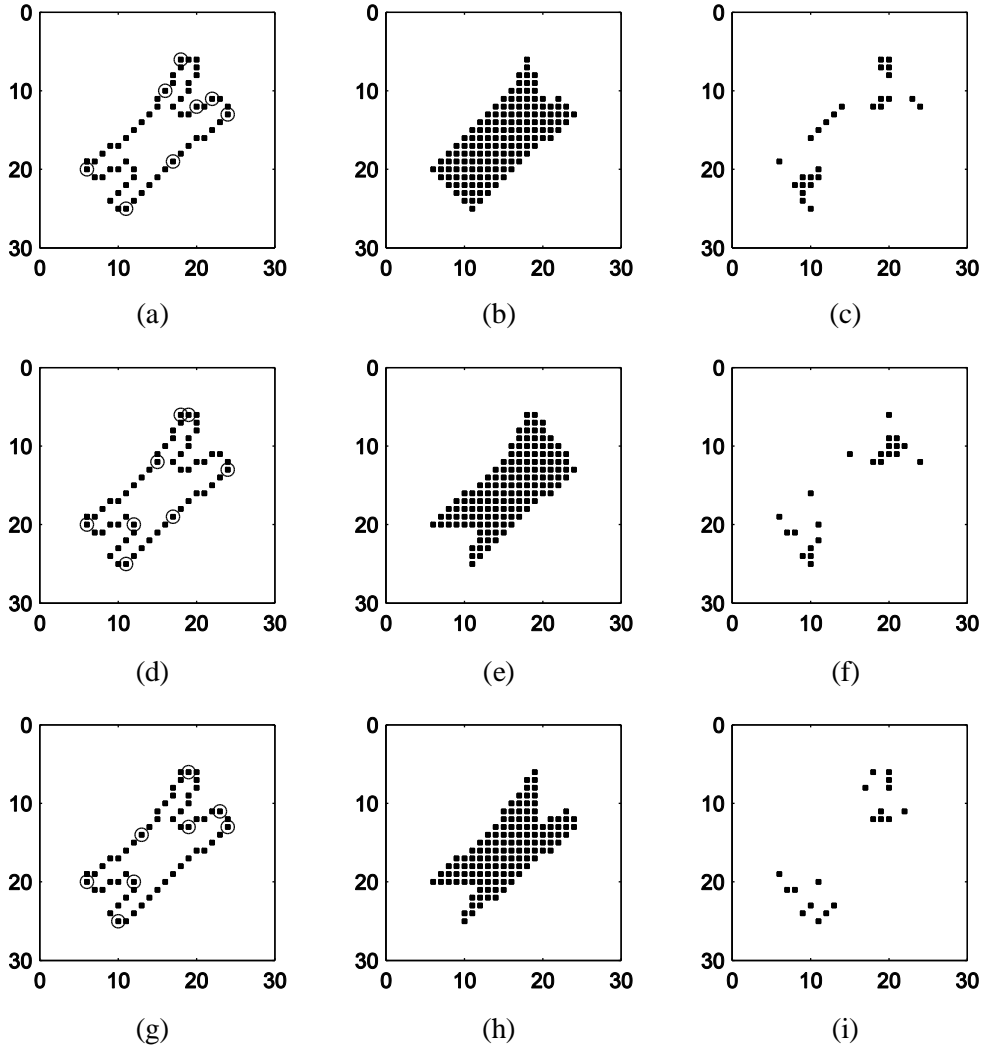
### 3. BULGULAR VE SONUÇLAR

Bu bölümde yerel ve genel yöntemlerin tümünün başarıları sırasıyla üçgen dörtgen ve beşgen olmak üzere 1000'er adet rastgele yapay çokgen üzerinde yapılan benzetimler ile sınanmış ve benzetim sonuçları Tablo 3.1 ile verilmiştir. Bununla birlikte yöntemler hem yerel ve genel olarak kendi içlerinde, hem de tamamı birbirleri ile hata oranı ve algoritmalarının çalışma süreleri göz önünde bulundurularak irdelenmiştir. Ayrıca Şekil 3.1 ve Şekil 3.2 ile gösterilen literatürde sıklıkla kullanılan kromozom (chromosome) şekli kullanılarak GA ve Tabu Arama algoritmalarının uygulanması verilmiştir.

Literatürde bilinen bir şeklin kullanılması bu çalışmadaki yaklaşımların literatürdeki diğer yaklaşımlarla karşılaştırılmasına olanak sağlamaktadır. Bu çalışmada amaç fonksiyonunun tanımlanmasındaki farklılıktan dolayı başarı değerlerinin bire bir karşılaştırılması şeklinde bir kıyaslama yapılamasa da bulunan baskın noktaların konumları referans alınarak bi tür karşılaştırmanın yapılabilmesi mümkündür.

Bu şekil için her iki yöntemde de baskın nokta sayısı  $K = 8$  olarak seçilmiştir. Başarı ve hata oranı baskın nokta sayısı arttıkça değişecektir. Zira  $N$  sınır noktasının oluşturduğu bir kapalı eğrinin en iyi şekilde  $N$  sayıda köşe ile temsil edilebileceği bilinen bir durumdur.

Şekil 3.1'de kromozom nesnesine GA'nın uygulanması verilmiştir. (a)'da 10. iterasyondaki çözüme ilişkin baskın noktalar, (b)'de bulunan baskın noktaların oluşturduğu  $J$  görüntüsü ve (c)'de ise 10. iterasyondaki çözüme ilişkin maliyet değeri gözeysel alan olarak görülmektedir. Şekil 3.1(d)'de 15. İterasyon sonucunda elde edilen çözüme ilişkin baskın noktalar, (e)'de bulunan baskın noktaların oluşturduğu  $J$  görüntüsü ve (f)'de ise 15. iterasyondaki çözüme ilişkin maliyet değeri gözeysel alan olarak görülmektedir. Son olarak Şekil 3.1(g)'de 20. İterasyon sonucunda elde edilen çözüme ilişkin baskın noktalar, (h)'de bulunan baskın noktaların oluşturduğu  $J$  görüntüsü ve (i)'de ise 20. iterasyondaki çözüme ilişkin maliyet değeri gözeysel alan olarak görülmektedir. Bu şekille ilgili algoritmanın  $K = 8$  için oldukça başarılı olduğu, mümkün olan hemen hemen en az nokta sayısı ile temsil edilebilmiş olmasından söylenebilir. Daha önce de belirtildiği gibi nokta sayısı artırıldıkça başarı da o oranda artacaktır. (c), (f), (i)'den görüldüğü gibi hata giderek azalmakta ve son iterasyonda algoritma en iyi sonucu üretmektedir. Sonuç olarak en küçük maliyetin 19 birim (gözek) kare olduğu görülür.

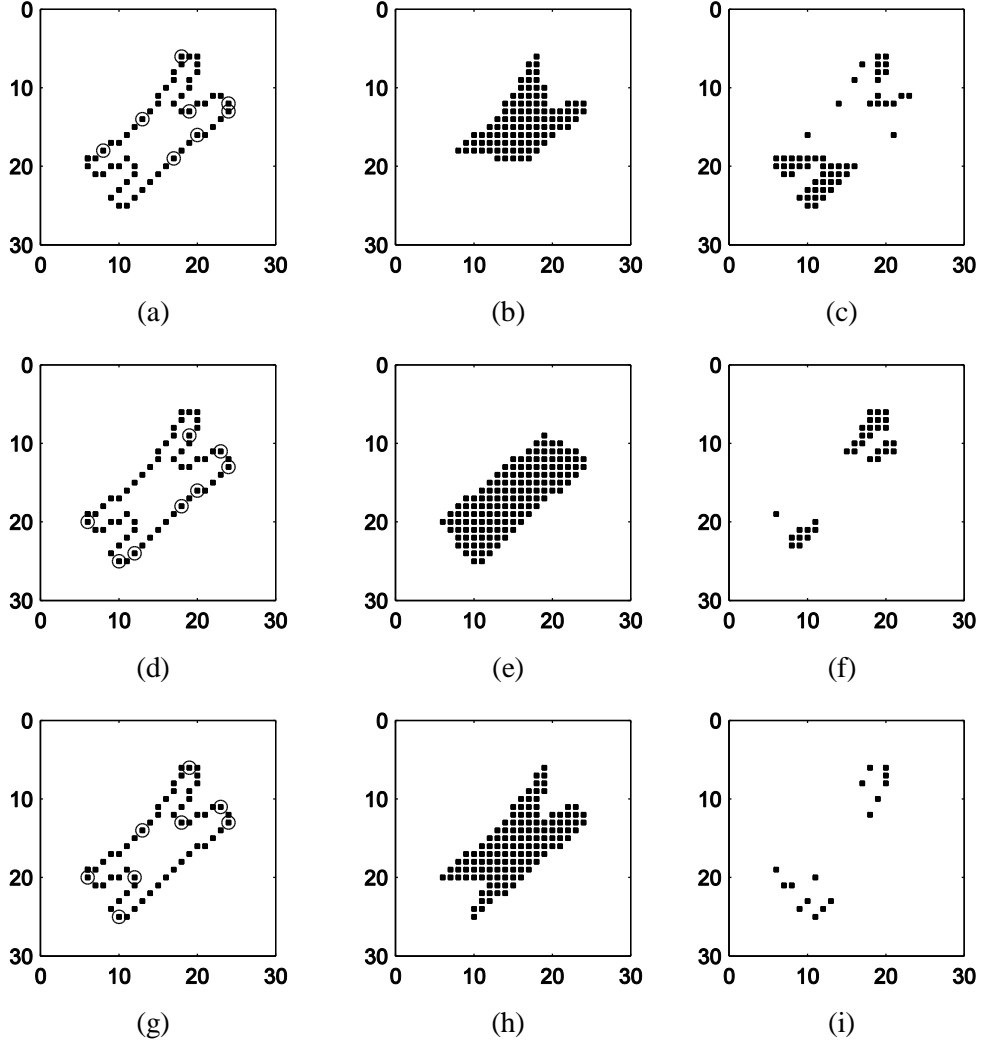


Şekil 3.1. Kromozom nesnesine genetik algoritma uygulaması, (a) 10. iterasyondaki çözüm; (b) 10. iterasyona ilişkin  $J$  görüntüsü; (c) 10. iterasyona ilişkin maliyet değeri; (d) 15. iterasyondaki çözüm; (e) 15. iterasyona ilişkin  $J$  görüntüsü; (e) 15. iterasyona ilişkin maliyet değeri; (g) 20. iterasyondaki çözüm; (h) 20. iterasyona ilişkin  $J$  görüntüsü; (i) 20. iterasyona ilişkin maliyet değeri.

Şekil 3.2’de Tabu Arama algoritmasının kromozom şekline uygulanması verilmiştir. (a)’da tabu arama algoritmasının 10. denemedeki çözümüne ilişkin baskın noktalar, (b)’de 10. Denemedeki baskın noktalardan elde edilen  $J$  görüntüsü ve (c)’de ise 10. Denemedeki çözüme ilişkin maliyet değeri gözeysel alan olarak görülmektedir. Şekil 3.2(d)’de algoritmanın 20. denemedeki çözümüne ilişkin baskın noktalar, (e)’de 20. denemedeki baskın noktalardan elde edilen  $J$  görüntüsü ve (f)’de ise 20. Denemedeki çözüme ilişkin maliyet değeri gözeysel alan olarak görülmektedir.



Son olarak Şekil 2.11(g)'de algoritmanın 12. denemesine ilişkin baskın noktalar, (h)'de 12. denemedeki baskın noktalardan elde edilen  $J$  görüntüsü ve (i)'de ise en iyi çözüm olan 12. çözüme ilişkin maliyet değeri görülmektedir.



Şekil 3.2. Kromozom nesnesine Tabu Arama algoritma uygulaması, (a) 10. denemedeki çözüm; (b) 10. denemedeki baskın noktalardan elde edilen  $J$  görüntüsü; (c) 10. denemedeki çözüme ilişkin maliyet değeri; (d) 20. denemedeki çözüm; (e) 20. Denemedeki baskın noktalardan elde edilen  $J$  görüntüsü; (e) 20. Denemedeki çözüme ilişkin maliyet değeri; (g) En iyi çözüm olan 12. Denemedeki çözüm; (h) 12. Denemedeki baskın noktalardan elde edilen  $J$  görüntüsü; (i) en iyi çözüm olan 12. Denemedeki çözüme ilişkin maliyet değeri.

Bu şekille gösterilen çözümler de Tabu Arama algoritmasının başarısını göstermektedir.  $K = 8$  için denemelerin tekrarlanmasıyla (c), (f), (i) ile gösterilen hata değerinin hızlı bir şekilde azaldığı görülmektedir.

Sonuç olarak en küçük maliyet değeri bu şekil için 16 birim (gözek) karedir. Yukarıdaki grafiklerden GA'nın çözümlerine ilişkin maliyet değeri ve iterasyon sayısı ile bir karşılaştırması yapılacak olursa Tabu Arama algoritmasının GA'ya göre daha başarılı olduğu söylenebilir. Ayrıca Tabu Arama algoritmasının ve diğer tüm algoritmaların çalışma hızları için ölçümler Tablo 3.1 ile de gösterilmiştir. Bu konudaki net bilgiye ulaşmak için tablodaki benzetim sonuçları incelenmelidir.

Tablo 3.1'de yerel yöntemler ve genel yöntemlerin tümünün yöntemlerin ortalama hesaplama süresi ve oluşan ortalama hata değerleri temel alınarak 1000'er adet rastgele yapay üçgen dörtgen ve beşgen ile yapılan benzetimlerinin sonuçları verilmiştir. Burada ortalama hesaplama süresi her bir çözüm için, toplam sürenin 1000'e bölünmesiyle elde edilmiştir. Ortalama hata değeri ise hatalar toplamının toplam alana oranı ile elde edilmiştir.

Tablo 3.1. Baskın nokta algılama ve araştırma yöntemlerinin başarı değerleri

	Yapay Üçgenler		Yapay Dörtgenler		Yapay Beşgenler	
Ortalama Alan	255,793		275,922		286,464	
Ortalama Çevre	63,434		58,351		60,931	
	Ortalama Hesaplama Süresi(sn)	Ortalama Uyum hatası	Ortalama Hesaplama Süresi(sn)	Ortalama Uyum hatası	Ortalama Hesaplama Süresi(sn)	Ortalama Uyum hatası
Kütle-Zemin Oranı	0,0022	0,0025	0,0022	**0,2606	0,0022	**0,2406
Göreceli Uzaklık	*0,0020	0,0805	*0,0020	0,1632	*0,0020	0,1670
Kesişim Açısı	0,1075	**0,2118	0,0992	0,1395	0,1031	0,1102
Belirtme Katsayısı	0,0082	0,0779	0,0077	0,0855	0,0079	0,1515
Tabu Arama Algoritması	0,1976	*0,0000	0,3359	*7,26E-06	0,9172	*0,0004
Genetik Algoritma	**3,1133	0,00027	**4,9071	0,0021	**6,9788	0,0046

(\*) ve (\*\*) ile belirtilen değerler ait oldukları sütunların sırasıyla en küçük ve en büyük değerleridir.

Tablo 3.1 incelendiğinde benzetim sonuçlarına göre kullanılan yapay üçgenler için ortalama alanın yaklaşık 256, dörtgenler için yaklaşık 276 ve beşgenler için yaklaşık 286 birim kare olduğu görülmektedir.

Yapay üçgenler için Kütle-Zemin oranı ortalama 0,0022 sn'lik bir süre ile çözüm yapmış ve %0,00025'lik bir hata ile yerel yöntemler arasında en iyi çalışan algoritmadır. Kütle-Zemin Oranı yöntemi en az sürede en doğru çözümü vermiştir. Göreceli Uzaklık yöntemi ortalama 0,0020 sn'lik bir sürede çözüm yapmış ve ortalama yaklaşık %0,008'lik bir hata ile yapay üçgenler için kütle-zemin oranından sonra en iyi yöntemdir denilebilir. Bu iki yöntemi ortalama 0,0082 sn'lik bir sürede çözüm yaparak ortalama olarak yaklaşık %0,0078 oranında hata ile Belirtme Katsayısı yöntemi izlemektedir. Bu durumda süre ve hata ölçütünün her ikisi de göz önünde bulundurulursa yerel yöntemler arasında yapay üçgen görüntüsü için en az başarılı yöntemin Kesişim Açısı yöntemi olduğu söylenebilir. Yöntemin çözüm süresi ortalama 0,1075 sn ile diğer yerel yöntemlere göre oldukça fazladır. Bununla birlikte ortalama %0,02118 oranında hata ile en çok hata veren yöntemdir.

Yapay dörtgenleri incelenirse burada üçgenlerden farklı olarak Belirtme Katsayısı yöntemi ortalama 0,0077 sn'lik bir süre ile ortalama %0,00855 oranında hata ile çözüm yapmıştır ve hatasının azlığı göz önünde bulundurulursa yapay dörtgenler için en iyi çalışan yöntemdir denilebilir. Ortalama 0,0992 sn'lik çözüm süresi ve ortalama yaklaşık %0,01395 oranında hata değeri ile Kesişim Açısı yöntemi ile birlikte ortalama 0,0020sn'lik çözüm süresi ve ortalama yaklaşık %0,016320 oranında hata değeri ile Göreceli Uzaklık yönteminin ikinci iyi yöntem olduğu söylenebilir. Kütle-Zemin Oranı yönteminin ise üçgenlerde gösterdiği başarıyı dörtgenlerde gösterebildiği söylenememektedir. Bu yöntem ortalama 0,0022 sn'lik bir sürede çözüm yaparak ortalama olarak yaklaşık %0,02606 oranında hata ile dört yerel yöntem arasında en yüksek hata oranına sahip yöntemdir.

Yapay beşgen görüntülerde yerel yöntemler arasında ortalama 0,1031 sn'lik işlem süresi ve ortalama yaklaşık %0,01102'lik hata değeri ile hata oranları göz önünde bulundurulduğu zaman Kesişim Açısı yönteminin en başarılı yöntem olduğu söylenebilir. Belirtme Katsayısı yöntemi burada ortalama 0,0079 sn çözüm süresi ve ortalama yaklaşık %0,01515 hata oranı ile ikinci en iyi yöntemdir. Göreceli Uzaklık yöntemi ise ortalama 0,0020 sn çözüm süresi ve ortalama yaklaşık %0,01670 oranında bir hata payına sahiptir.

Yapay beşgenlerde de yerel yöntemler arasında en az başarılı yöntem Kesişim Açısı yöntemidir.

Benzetim sonuçlarına göre kendi içlerinde karşılaştırıldıklarında yerel yöntemler arasında hiçbir çokgen türü için en başarılı ya da en az başarılı olmayan en vasat yöntemin Göreceli Uzaklıklar yöntemi olduğu sonucu çıkarılabilir. Dolayısıyla bu yöntem için yerel yöntemler arasında standart bir yöntemdir denilebilir.

Yapay üçgenler için Tabu Arama yönteminin ortalama 0,1976 sn çözüm süresi ve ortalama %0,0000 gibi bir hata oranıyla neredeyse hatasız çözüm yapmış olduğu görülmektedir. Genetik Algoritma yönteminin ise işlem süresi ortalama 3,1133 sn olmakla birlikte Tabu Aramadan oldukça fazladır. Hata oranı ise ortalama yaklaşık %0,000027'dir. Yani yapay üçgenler için GA hem süre yönünden hem de hata oranı dikkate alındığında Tabu Arama yöntemine yaklaşamamıştır.

Yapay dörtgenler için Tabu Arama yönteminin işlem süresi ortalama 0,3359 sn'dir. Hata oranı ise ortalama %7,26E – 07 olan çok küçük bir hata oranıyla hatasız çözüm yapmış olduğu görülmektedir. Genetik algoritma yönteminin ise hata oranı ortalama yaklaşık %0,00021 olarak gözlenirken işlem süresi ortalama 4,9071 sn'dir. Dörtgen şekiller için de benzer şekilde Tabu Arama yöntemine göre oldukça yavaştır. Yani GA hem süre yönünden hem de hata oranı dikkate alındığında burada da Tabu arama yöntemine yaklaşamamıştır.

Yapay beşgenler için Tabu Arama yönteminin işlem süresi ortalama 0,9172 sn'dir. Hata oranı ise ortalama %00004 olup oldukça başarılı olduğu söylenebilir. Genetik Algoritma yönteminde hata oranı ortalama %0,00046'dır ve işlem süresi ortalama 6,9788 sn'dir. Beşgen şekiller için de aynı şekilde Tabu Arama yöntemine göre oldukça yavaştır. Ayrıca nesnenin köşe sayısı arttıkça Genetik Algoritma yönteminin işlem süresinin de arttığı gözlenmektedir. Yani GA hem süre yönünden hem de hata oranı dikkate alındığında burada da Tabu Arama yöntemine yaklaşamamıştır.

Genel olarak tüm yöntemler hakkında yorum yapmak gerekirse, yerel yöntemlerin nesnenin sınır gözeleklerinin sayı kadar işlem yapmalarından dolayısıyla da iterasyon olmaksızın çalıştıklarından dolayı genel yöntemlere göre çok daha hızlı çalıştıkları görülmektedir. Bu durumun tüm nesnelere için geçerli olduğu Tablo 3.1'den açıktır. Ancak bir yöntemin başarılı olup olmadığı hakkında karar verilirken genellikle (arada aşırı süre farkları yok ise) hata oranının düşüklüğü göz önünde bulundurulur. Bu nedenle hızlı çözüm tek başına bir anlam ifade edememektedir. Dolayısıyla benzetim sonuçları

incelendiğinde ürettiđi çok iyi sonuçlardan ötürü genel yöntemlerin yerel yöntemlere göre tercih edilmesi kaçınılmazdır. Bununla birlikte çok iyi sonuçlar üreten genel yöntemler arasında Tabu Arama yöntemi için deneme sonuçları hemen hemen kusursuzdur. Tabu yönteminin sadece çokgenlerde değil, daha karmaşık şekillerde bile baskın noktaları çok başarılı çalıştığı Şekil 3.2'den görülebilmektedir.

#### **4. KAYNAKLAR**

Ansari, N. and Huang, K.W., 1991. Non-parametric Dominant Point Detection, Pattern Recognition, 24, 849-862

Arce, G. R., Paredes J. L. and Mullan J., 2000. Nonlinear Filtering For Image Analysis and Enhancement, in Handbook of Image and Video Processing. Academic Press, San Diego.

Attneave, F., 1954. Some Informational Aspects of Visual Perception, Psychological Review, 61, 3, 183.

Burger, W. and Burge, M.J., 2009. Principles of Digital Image Processing Fundamental Techniques, Springer-Verlag Press Company. London, UK.

Burger, W. and Burge, M.J., 2009. Principles of Digital Image Processing Core Algorithms, Springer-Verlag Press Company. London, UK.

Buzuloiu V., Ciuc M. and Rangayyan R. M ., 2001. Adaptive-Neighborhood Histogram Equalization of Color Images. J. Electron. Imaging, 10,2,445-459.

Castleman, K. R., 1996. Digital Image Processing, Prentice-Hall Publishing Company, NewJersey, USA.

Costa L. F. and Cesar R. M., 2001. Shape Analysis and Classification, CRC Press, Boca Raton, FL.

Davies E. R., 1988. On the Noise Suppression and Image Enhancement Characteristics of the Median, Truncated Median and Mode Filters, Pattern Recognition Lett. 7, 87-97.

Davies, E., 2003. Machine Vision: Theory, Algorithms and Practicalities, Academic Press.

Fisher, R., Perkins, S., Walker, A. and Wolfart, E., Thresholding. <http://homepages.inf.ed.ac.uk/rbf/HIPR2/threshld.htm>. 26 Ağustos 2011.

Fu, K. S., 1982. Syntactic Pattern Recognition and Applications, Processing, Prentice-Hall Publishing Company, NewJersey, USA.

Glover, F., 1989. Tabu Search-Part I, ORSA Journal on Computing, 1, 3, 190-206.

Gonzalez, R.C. and Woods, R.E., 2002. Digital Image Processing, Prentice-Hall Publishing Company, NewJersey, USA.

Gonzalez, R.C., Woods, R.E. and Eddins, S.L., 2004. Digital Image Processing with Matlab, Prentice-Hall Publishing Company, NewJersey, USA.

Haupt, R.L. and Haupt, S.E., 2004. Practical Genetic Algorithms, Wiley-Interscience Publishing Company, New Jersey, USA.

Hearn, D. and Baker, M.P., 1997. Computer Graphics C Version, Second Edition, Prentice-Hall Publishing Company, New Jersey, USA.

Holland, J.H., Adaptation In Natural and Artificial Systems. University of Michigan Press, Michigan, 1975.

<http://en.wikipedia.org/wiki/Polygon>, 02 Aralık 2011.

Huang T. S., 1979. A Fast Two-Dimensional Median Filtering Algorithm. IEEE Trans. ASSP 27, 13-18.

Jahne, B., 2002. Digital Image Processing, Springer Publishing Company, Heidelberg, Germany.

Kesemen, O., 2006-2011. Sayısal Görüntü İşleme Ders Notları, İstatistik ve Bilgisayar Bilimleri Bölümü, Karadeniz Teknik Üniversitesi, Trabzon.

Kesemen, O., Aktaş, N. ve Solak, Y.N., 2010, Sayısal Görüntülerdeki Taneciklerin Topaklık Ölçüsünün Belirlenmesi, 9. Matematik Sempozyumu, 20-22 Ekim, Trabzon.

Kesemen, O. ve Güngör, Ç., 2011. Sayısal Görüntülerdeki Nesnelerin Baskın Noktalarının Algılanmasında İstatistiksel Yaklaşımlar, Uluslararası 7. İstatistik Kongresi, Antalya, Türkiye.

Kesemen, O., Karakaya, G. ve Gürbüz, F., 2010. Sayısal Görüntülerde Taneciklerin Bulanık Yönelimlerinin Belirlenmesi, İstatistik Günleri Sempozyumu, 28-30 Haziran, Ankara.

Kurozumi, Y. and Davis, W., 1981. Polygonal Approximation by The Minimax Method, Comput. Graphics Image Process., 19, 248–264

Leu, J.G. and Chen, L., 1988. Polygonal Approximation of 2-D Shapes Through Boundary Merging, Pattern Recogn. Let, 7, 231-238

Marji, M. and Siy, P., 2004. A New Algorithm For Dominant Points Detection and Polygonization of Digital Curves, The Journal of Pattern Recognition Society Press, 37, 2113-2130.

Marshall, S., 1989. Review of Shape Coding Techniques, Image and Vision Computing, 7, 4, 281–294.

Martin, D., Fowlkes, C., and Malik, J. 2004. Learning to Detect Natural Image Boundaries Using Local Brightness, Color and Texture Cues, IEEE Transactions on Pattern Analysis and Machine Intelligence, 26, 5, 530–549.

Otsu, N. 1979. A Threshold Selection Method From Gray-level Histogram, IEEE Trans. Systems Man Cybern., 9, 1, 62–66.

- Pavlidis, T., 1977. Polygonal Approximation by Newton's method, IEEE Trans. Compute., 26, 800-807
- Pavlidis, T. and Horowitz, S.L., 1974. Segmentation of Plane Curves, IEEE Transactions on Computers, 860-870.
- Pratt, W.K., 2007. Digital Image Processing, Wiley-Interscience Publishing Company, New Jersey, USA.
- Ramer, U., 1972. An Iterative Procedure for the Polygonal Approximation of Planar Curves, Computer Graphics and Image Processing, 244-256.
- Ray, B.K. and Ray, K.S., 1991. A New Approach to Polygonal Approximation, Pattern Recogn. Let., 12, 229-234.
- Rosenfeld, A., 1969. Picture Processing by Computer, Academic Press, New York, USA.
- Searle, JR., 1996. Minds, Brains and Science, Harvard University Press, Cambridge.
- Soille, P., 1999. Morphological Image Analysis: Principles and Applications, Springer-Verlag Press Company.
- Szeliski, R., 2011. Computer Vision Algorithms and Applications, Springer-Verlag Press Company.
- Teh, C.H. and Chin, R.T., 1989. On the Detection of Dominant Points on Digital Curves, IEEE Trans. Pattern Analysis and Machine Intelligence., 8, 859-872.
- Thai, R., 2003. Fingerprint Image Enhancement and Minutiae Extraction, Honours project, The University of Western Australia.
- Wu, J.S. and Leou, J.J., 1993. New Polygonal Approximation Schemes For Object Shape Representation. Pattern Recognition Let., 26, 471-484.



## ÖZGEÇMİŞ

Çiğdem GÜNGÖR, 11 Haziran 1986 tarihinde İstanbul'da doğdu. İlköğrenimini Güngören Ali Fuat Cebesoy İlköğretim Okulu'nda, ortaöğrenimini ise Fatih Vatan Anadolu Lisesi'nde tamamladı. 2005-2006 öğretim yılında Karadeniz Teknik Üniversitesi, Fen Edebiyat Fakültesi, İstatistik ve Bilgisayar Bilimleri Bölümü'ne yerleşti. 2009 yılında bu bölümden mezun oldu. Aynı yıl Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstatistik ve Bilgisayar Bilimleri Anabilim dalında tezli yüksek lisans programına başladı. Ekim 2010 – Eylül 2011 tarihleri arasında Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstatistik ve Bilgisayar Bilimleri Anabilim Dalında araştırma görevlisi olarak çalıştı. Eylül 2011 tarihinde Yüksek Öğretim Kurulunun Öğretim Üyesi Yetiştirme Programı kapsamında İstanbul Medeniyet Üniversitesi, Fen Fakültesi, İstatistik Bölümü'ne araştırma görevlisi olarak atandı ve halen bu görevine devam etmektedir. Ulusal ve uluslararası birçok sempozyum ve kongrede yayınlanmış bildirileri bulunmakta ve iyi derecede İngilizce bilmektedir.