

TOPOLOJİK KENAR İŞLEÇLERİ

Elif AYBAR
Doktora Tezi

Fen Bilimleri Enstitüsü
Elektrik-Elektronik Mühendisliği Anabilim Dalı
Aralık – 2003

JÜRİ VE ENSTİTÜ ONAYI

Elif Aybar'ın **Topolojik Kenar İşleçleri** başlıklı **Elektrik-Elektronik Mühendisliği** Anabilim Dalındaki, Doktora tezi ~~19.12.2004~~ tarihinde, aşağıdaki jüri tarafından Anadolu Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin ilgili maddeleri uyarınca değerlendirilerek kabul edilmiştir.

	Adı-Soyadı	İmza
Üye (Tez Danışmanı)	: Prof. Dr. ATİLA BARKANA	
Üye	: Yrd. Doç. Dr. HAKAN G. ŞENEL	
Üye	: Yrd. Doç. Dr. NİHAT ADAR	
Üye	: Yrd. Doç. Dr. ÖMER N. GEREK	
Üye	: Yrd. Doç. Dr. EMİN GERMEN	

Anadolu Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ~~07.01.2004~~.. tarih ve ~~..1/1~~..... sayılı kararıyla onaylanmıştır.

/Enstitü Müdürü
Prof. Dr. Mehmet DZEA
Fen Bilimleri Enstitüsü
M ü d ü r ü

ÖZET

Doktora Tezi

TOPOLOJİK KENAR İŞLEÇLERİ

ELİF AYBAR

Anadolu Üniversitesi
Fen Bilimleri Enstitüsü
Elektrik-Elektronik Mühendisliği Anabilim Dalı

Danışman: Prof.Dr.Atila BARKANA
2003, 78 sayfa

Bu tez çalışmasında, topolojik özellikleri kullanan iki yeni kenar algılama yöntemi geliştirilmiştir. Bağlantı haritası (conn-map) bir gözlem penceresindeki orta pikselin diğerlerine nasıl bağlandığını gösterimidir. Her iki kenar algılama yöntemi, ön aşama olarak, her kayan pencere için bağlantı haritasını hesaplar ve hesaplanan bağlantı bilgisini kullanır. İlk yöntem öz uyarlamalı bir algoritma sonucunda çıktısını üretir. İkinci yöntem ise bağlantı haritası üzerine Sobel kenar işlecini uygular. Literatürdeki çalışmalarda, genellikle, 3x3 büyüklüğünde Sobel işleci kullanılmaktadır, çünkü, daha büyük (örneğin, 5x5, 7x7) işleçler daha kalın kenar çıktısı üretmektedir. Bağlantı haritası orta piksele bağlı olmayan kenarları süzer, bu nedenle, geliştirilen iki yöntem de daha ince kenar çıktısı üretmektedir. Ayrıca, her iki yöntemin, gürültü ortamında performansları incelenmiştir. Özellikle, geleneksel Sobel işlecine göre, ikinci yöntemin daha üstün özellikleri olduğu anlaşılmıştır.

Anahtar Kelimeler: Kenar algılama, Kenar İşleci, Görüntü Topolojisi, Bağlantı Haritası, Sobel İşleci.

ABSTRACT**PhD Dissertation****TOPOLOGIC EDGE OPERATORS****ELİF AYBAR**

**Anadolu University
Graduate School of Sciences
Electrical and Electronics Engineering Program**

**Supervisor: Prof.Dr.Atila BARKANA
2003, 78 pages**

In this thesis, two new edge detection methods that make use of topological properties are proposed. Connectivity map (conn-map) is a quantitative depiction of how the center pixel is connected to others in an observation window. Both edge detection methods compute connectivity map in a sliding window and use the connectivity information. First method use an self adaptive algorithm to produce an output. The second applies Sobel edge operator on the conn-map. For simplicity purposes, Sobel edge operator of size 3x3 has been generally used in edge detection operations. In the literature, in general, larger Sobel operators (e.g. 5x5, 7x7 and more) are not utilized due to the reason that they produce thicker edges since the number of edges included in larger windows increases. Connectivity map filters the edges that are not connected to the center pixel, therefore, both methods yield leaner edges. Performances in noisy image environment are also studied. Especially, the second method is determined to have a superior performance compared to conventional Sobel operator.

Keywords: Edge Detection, Edge Operators, Image Topology, Connectedness Map, Sobel Operator

TEŐEKKÜR

Bu alıőmada bana yardımcı olan sayın Prof. Dr. Atila Barkana ve Yrd. Do. Dr. Hakan G. Őenel'e teőekkür ederim. Ayrıca manevi desteklerinden ve sabırlarından dolayı aileme, arkadaşlarıma ve özellikle ođlumuz Mert'e teőekkürler.

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET	i
ABSTRACT	ii
TEŞEKKÜR	iii
İÇİNDEKİLER.....	iv
ŞEKİLLER DİZİNİ	vi
ÇİZELGELER DİZİNİ.....	x
1. GİRİŞ.....	1
2. KENAR ALGILAMA	4
2.1. Yerel Kenar İşleçleri	5
2.1.1. Roberts Kenar Algılayıcı	6
2.1.2. Prewitt Kenar Algılayıcı	8
2.1.3. Sobel Kenar Algılayıcı	9
2.1.4. Canny Kenar İşleci	14
2.1.5. Marr-Hildreth Kenar Algılayıcı.....	16
2.2. Diğer Kenar Algılayıcılar.....	17
2.2.1. Nesne Temelli Kodlama İçin Dönüşüm Alanı Tekniğini Kullanarak Kenar Algılama	18
2.2.2. Gerçek Zamanlı Kenar Algılayıcı	20
2.2.3. Doğrusal Olmayan Laplas Filtrelemeyi Temel Alan Kenar Algılayıcı.....	22
2.2.4. Kesikli Tekil Evrişim (DSC) Algoritması	24

3. BAĞLILIK	27
3.1. Matematiksel Olarak Bağlılık Tanımı.....	27
3.2. Bulanık Küme	29
3.3. Topoloji.....	30
3.4. Bulanık Topoloji	31
3.5. Bağlantı Haritası	34
4. BAĞLANTI HARİTASININ KENAR ALGILAMADA KULLANILMASI	36
4.1. Kenar Algılama Algoritması-1	36
4.2. Kenar Algılama Algoritması-2.....	46
5. SONUÇ.....	63
5.1. Kenar Algılama Algoritması-1'e Ait Sonuçlar	65
5.1.1. Maksimum Değerler.....	65
5.1.2. Yumuşatma Oranları	66
5.1.3. Tepe Noktası.....	67
5.1.4. Düzgün Alanlardaki Dalgalanma Varyansları	68
5.2. Kenar Algılama Algoritması-2'ye Ait Sonuçlar	69
5.2.1. Maksimum Değerler.....	69
5.2.2. Yumuşatma Oranları	70
5.2.3. Tepe Noktası.....	72
5.2.4. Düzgün Alanlardaki Dalgalanma Varyansları	73
5.2.5. Algoritma Hızları	74
5.3. Gelecekte Yapılması Planlanan Çalışmalar	75
KAYNAKLAR.....	76

ŞEKİLLER DİZİNİ

2.1. Orijinal görüntü ve Roberts kenar algılayıcı uygulandığında elde edilen sonuç görüntüsü	7
2.2. Orijinal görüntü ve Prewitt kenar algılayıcı uygulandığında elde edilen sonuç görüntüsü	9
2.3. Dört temel yönde gradyan vektörleri.....	10
2.4. 3×3 komşulukta yer alan piksellerin yerleşimi.....	10
2.5. Yatay ve düşey yöndeki ağırlık katsayıları	11
2.6. 5×5 komşulukta yer alan piksellerin yerleşimi	11
2.7. Yatay ve düşey yöndeki ağırlık katsayıları	13
2.8. Orijinal görüntü, 3×3 Sobel maske uygulanmış görüntü sonucu, 5×5 Sobel maske uygulanmış görüntü sonucu	14
2.9. Orijinal görüntü ve Canny kenar algılayıcı uygulandığında elde edilen sonuç görüntüsü	16
2.10. Orijinal görüntü ve LoG kenar algılayıcı uygulandığında elde edilen sonuç görüntüsü	17
2.11. Orijinal görüntü ve Suthaharan tarafından oluşturulan yöntemin sonucu	20
2.12. Yarı-Gauss düzgünleştirme maskesi	21
2.13. Mutlak fark maskesi	21
2.14. Orijinal görüntü ve gerçek zamanlı kenar algılama yöntemine ait sonuç görüntüsü	22
2.15. Orijinal görüntü ve doğrusal olmayan Laplas yöntemi ile elde edilen sonuç görüntüsü	24
2.16. Orijinal görüntü ve DSCED algoritması sonucu elde edilen görüntü	25
2.17. Orijinal görüntü ve DSCANED algoritması sonucu elde edilen görüntü	26
3.1. 4 ve 8 komşuluklar	27
3.2. Bir pikselin 4-komşuluğu ve 8-komşuluğu	28

3.3. Baęlılık ve baęlantı Őiddeti.....	33
3.4. Orijinal grnt ve baęlantı haritası oluŐturulmuŐ grntler	35
4.1. Baęlantı haritası kullanılarak geliŐtirilen ynteme ait akıŐ diyagramı	39
4.2. Orijinal grnt, Sobel algılayıcı sonucu, Canny algılayıcı sonucu, LoG algılayıcı sonucu, yeni algılayıcı sonucu.....	40
4.2.(devamı) Orijinal grnt, Sobel algılayıcı sonucu, Canny algılayıcı sonucu, LoG algılayıcı sonucu, yeni algılayıcı sonucu	41
4.2.(devamı) Orijinal grnt, Sobel algılayıcı sonucu, Canny algılayıcı sonucu, LoG algılayıcı sonucu, yeni algılayıcı sonucu	42
4.2.(devamı) Orijinal grnt, Sobel algılayıcı sonucu, Canny algılayıcı sonucu, LoG algılayıcı sonucu, yeni algılayıcı sonucu	43
4.2.(devamı) Orijinal grnt, Sobel algılayıcı sonucu, Canny algılayıcı sonucu, LoG algılayıcı sonucu, yeni algılayıcı sonucu	44
4.2.(devamı) Orijinal grnt, Sobel algılayıcı sonucu, Canny algılayıcı sonucu, LoG algılayıcı sonucu, yeni algılayıcı sonucu	45
4.3. Ynteme ait akıŐ diyagramı.....	47
4.4. 7x7'lik baęlantı haritası oluŐturulan grnt maskesi zerine 5x5'lik Sobel iŐlecinin yerleŐimi.....	48
4.5. Birinci kolon orijinal grnt, ikinci kolon Sobel 3x3 sonucu, nc kolon Sobel 5x5 sonucu, drdnc kolon con7sob5 sonucu, beŐinci kolon orijinal grntnn tersine con7sob5 uygulanması ile elde edilen sonu.....	50
4.5.(devamı) Birinci kolon orijinal grnt, ikinci kolon Sobel 3x3 sonucu, nc kolon Sobel 5x5 sonucu, drdnc kolon con7sob5 sonucu, beŐinci kolon orijinal grntnn tersine con7sob5 uygulanması ile elde edilen sonu.....	51
4.5.(devamı) Birinci kolon orijinal grnt, ikinci kolon Sobel 3x3 sonucu, nc kolon Sobel 5x5 sonucu, drdnc kolon con7sob5 sonucu, beŐinci kolon orijinal grntnn tersine con7sob5 uygulanması ile elde edilen sonu.....	52
4.5.(devamı) Birinci kolon orijinal grnt, ikinci kolon Sobel 3x3 sonucu, nc kolon Sobel 5x5 sonucu, drdnc kolon con7sob5 sonucu, beŐinci	

kolon orijinal görüntünün tersine con7sob5 uygulanması ile elde edilen sonuç.....	53
4.5.(devamı) Birinci kolon orijinal görüntü, ikinci kolon Sobel 3×3 sonucu, üçüncü kolon Sobel 5×5 sonucu, dördüncü kolon con7sob5 sonucu, beşinci kolon orijinal görüntünün tersine con7sob5 uygulanması ile elde edilen sonuç.....	54
4.5.(devamı) Birinci kolon orijinal görüntü, ikinci kolon Sobel 3×3 sonucu, üçüncü kolon Sobel 5×5 sonucu, dördüncü kolon con7sob5 sonucu, beşinci kolon orijinal görüntünün tersine con7sob5 uygulanması ile elde edilen sonuç.....	55
4.5.(devamı) Birinci kolon orijinal görüntü, ikinci kolon Sobel 3×3 sonucu, üçüncü kolon Sobel 5×5 sonucu, dördüncü kolon con7sob5 sonucu, beşinci kolon orijinal görüntünün tersine con7sob5 uygulanması ile elde edilen sonuç.....	56
4.5.(devamı) Birinci kolon orijinal görüntü, ikinci kolon Sobel 3×3 sonucu, üçüncü kolon Sobel 5×5 sonucu, dördüncü kolon con7sob5 sonucu, beşinci kolon orijinal görüntünün tersine con7sob5 uygulanması ile elde edilen sonuç.....	57
4.6. $\sigma=0$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla Sobel 3×3 ile con7sob5 sonuçları	58
4.6.(devamı) $\sigma=10$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla Sobel 3×3 ile con7sob5 sonuçları.....	59
4.6.(devamı) $\sigma=20$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla Sobel 3×3 ile con7sob5 sonuçları.....	59
4.6.(devamı) $\sigma=30$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla Sobel 3×3 ile con7sob5 sonuçları.....	60
4.6.(devamı) $\sigma=40$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla Sobel 3×3 ile con7sob5 sonuçları.....	60
4.6.(devamı) $\sigma=50$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla Sobel 3×3 ile con7sob5 sonuçları.....	61
4.6.(devamı) $\sigma=60$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla	

Sobel 3×3 ile con7sob5 sonuçları.....	61
4.6.(devamı) $\sigma=70$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla	
Sobel 3×3 ile con7sob5 sonuçları.....	62

ÇİZELGELER DİZİNİ

5.1. Gauss gürültü uygulanmış görüntülerde maksimum yoğunluk değerleri	66
5.2. Düzgün gürültü uygulanmış görüntülerde maksimum yoğunluk değerleri.....	66
5.3. Student-t gürültü uygulanmış görüntülerde maksimum yoğunluk değerleri...	66
5.4. Gauss gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin ortalaması	66
5.5. Düzgün gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin ortalaması	67
5.6. Student-t gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin ortalaması	67
5.7. Gauss gürültü uygulanmış görüntülerde tepe noktası ve diğer yoğunluk değerlerinin ortalaması arasındaki fark	67
5.8. Düzgün gürültü uygulanmış görüntülerde tepe noktası ve diğer yoğunluk değerlerinin ortalaması arasındaki fark	67
5.9. Student-t gürültü uygulanmış görüntülerde tepe noktası ve diğer yoğunluk değerlerinin ortalaması arasındaki fark	67
5.10. Gauss gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin varyansı	68
5.11. Düzgün gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin ortalaması	68
5.12. Student-t gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin ortalaması	68
5.13. Gauss gürültü uygulanmış görüntülerde maksimum yoğunluk değerleri.....	69
5.14. Düzgün gürültü uygulanmış görüntülerde maksimum yoğunluk değerleri.....	69
5.15. Student-t gürültü uygulanmış görüntülerde maksimum yoğunluk değerleri.....	69
5.16. Gauss gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin ortalaması	71

5.17. Düzgün gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin ortalaması	71
5.18. Student-t gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin ortalaması	71
5.19. Gauss gürültü uygulanmış görüntülerde tepe noktası ve diğer yoğunluk değerlerinin ortalaması arasındaki fark	72
5.20. Düzgün gürültü uygulanmış görüntülerde tepe noktası ve diğer yoğunluk değerlerinin ortalaması arasındaki fark	72
5.21. Student-t gürültü uygulanmış görüntülerde tepe noktası ve diğer yoğunluk değerlerinin ortalaması arasındaki fark	73
5.22. Gauss gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin varyansı	73
5.23. Düzgün gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin ortalaması	73
5.24. Student-t gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin ortalaması	73
5.25. Değişik boyutlu görüntüler üzerinde çalıştırılan algoritmaların hızları.....	74

1. GİRİŞ

Görüntüleme ve görüntü analizi yöntemlerini içinde barındıran sistemler yaygın bir uygulama alanı bulmaktadır. Görüntü işleme yöntemlerinin ilk geliştirilmeye başlandığı 1970'lerde görüntülerin genellikle çok yer kaplaması, görüntü işlemeye dayalı sistemlerde görüntülerin işlenmesini de zorlaştırmaktaydı. İlk sistemlerde, büyük miktardaki verinin işlenebilmesi için özel donanım cihazları kullanılmaktaydı. 1990'lardan itibaren, kişisel bilgisayar sistemlerinin kapasitelerinin hızlı bir şekilde gelişmesi ve yüksek işlem kapasiteli bilgisayarların ucuzlaması, görüntü işlemeye dayalı sistemlerin yaygınlaşmasını kolaylaştırmıştır [1].

İnsan görme sisteminin, görüntü içindeki nesnelere hızlı bir şekilde tanınması, görüntüdeki süreksizlikleri hızlı bir şekilde algılaması ve işlemesi sayesinde gerçekleşmektedir [2]. Bu nedenle, görüntü işlemeye dayanan sistemlerde görüntüde yer alan süreksizliklerin, diğer bir deyişle kenarların ve köşelerin algılanması işleminin önemli bir işlevi vardır. Kenar algılamının tanıma için gerekli olduğu düşüncesiyle beraber, süreksizliklerin bulunmasına yönelik çeşitli yöntemler geliştirilmiştir. Nesne tanıma sistemleri, genellikle, süreksizlikleri belirlemek ve kenar bilgisini kullanarak nesnelere belirlemek aşamalarında oluşmaktadır. Bununla birlikte, geliştirilen yöntemlerin kenar bulma sürecindeki farklılıkları nedeniyle hızları, doğru kenarları bulabilme yetenekleri, belirsiz kenarları algılayabilme yetenekleri ve işlem hızları oldukça değişken olabilir [1].

Görüntüde yer alan kenarları belirleyip nesne sınırlarını oluşturabilmek için tasarlanmış olan kenar algılama yöntemleri, yoğunluk fonksiyonunun birinci ve ikinci türevinden, Gauss ve Laplas fonksiyonundan ve yüksek geçiren süzgeçlerden yararlanmaktadır [1]. Var olan yöntemlerin birleştirilmesi ve bazı özel durumlar için uyarlanması, çeşitli kenar algılama yöntemlerinin çıkmasında önemli bir rol oynamıştır [3-8]. Görüntünün iki boyutlu bir fonksiyon olarak düşünülerek, fonksiyondaki süreksizliklerin hesaplanması için geliştirilen gradyan (gradient) ve evrişime (convolution) dayanan yöntemler de bulunmaktadır. Özellikle Prewitt, Sobel, Canny, LoG, Marr-Hildreth kenar işlemleri matematiksel

olarak kenarları belirleyen gradyan temelli tekniklerdir. Bu işlemlerin, genellikle kenar bulma işi için yeterli performansı sağladıkları bilinmektedir [9]. Basit olmaları kısa sürede uygulanabilmelerini sağlar. Deterministik özellikleri nedeniyle, verdikleri tepkiler daha kolay yorumlanabilir. Kenar bulma işlemi, genellikle, nesne algılama, nesne tanıma, görüntü iyileştirme ve görüntü sıkıştırma gibi daha üst seviyeli görüntü işlemlerinin ön aşaması olarak kullanıldığından, işlemin kısa sürede gerçekleşmesi önemli bir gerekliliktir [10].

Diğer yandan, basit kenar bulma işlemlerinin, çeşitli durumlara uyarlanmasıyla daha karmaşık yöntemler gerçekleştirilebilir. Literatürde, bu tür yöntemler önerilmiş ve verimli sonuçlar elde edildiği gösterilmiştir [3-8]. Basit ve bileşik kenar algılama yöntemlerine ait araştırma sonuçları Bölüm 2'de verilmektedir.

Bu tez çalışmasında, Prewitt, Sobel, LoG gibi basit kenar işlemleri uygulanmasından önce, bulanık topoloji kuramıyla ilgili bağlantı haritası işlemiyle desteklenmiş bir kenar bulma yöntemi geliştirilmiştir. Görüntü üzerinde gezdirilen gözlem penceresinden elde edilen değerlerin süzgeç katsayılarıyla çarpılarak toplamın elde edilmesinden önce, gözlem penceresi üzerinde bağlantı haritası oluşturulmaktadır. Bağlantı haritası, küçük pencereler için merkezdeki noktanın diğer piksellerle olan bağlantısını gösteren bir işlemidir [11].

Bağlantı haritasının uygulanması, görüntü üzerinde işlemin uygulanmasından önce, görüntünün bölütlenmesi anlamına gelmektedir. Bu aşamada, pencere içinde yer alan ve orta pikselin bağlı olduğu nesneyle ilgili olmayan diğer oluşumlar ortadan kaldırılmaktadır. Kenar bulma işleci bu sonuç üzerine uygulanmaktadır.

Yeni geliştirilen yöntemi açıklayabilmek için Bölüm 3'te bağıllık ve bulanık bağıllık kavramları ile topoloji ve bağlantı haritası tanımları yer almaktadır. Dördüncü bölümde, yeni yöntemin algoritması verildikten sonra, belirlenen özellikleri sunulmaktadır. Sonuçların sunulması, 60 görüntü üstünde yapılan sınaama sürecinden elde edilen sonuçların, diğer yöntemlerden elde edilenlerle karşılaştırılmasıyla yapılmaktadır. Ayrıca, çeşitli görüntü ortamlarındaki performansları da bulunmuş ve sonuçlar bölümüne eklenmiştir.

Yeni yöntem sonuçlarından yola çıkarak, ilk kenar algılama yöntemlerinden biri olan Sobel kenar algılama operatörünün daha önce 3×3 boyutunda olan maske boyutu 5×5 olacak şekilde genişletilerek bu maskelerin bağlantı haritası oluşturulan görüntülere uygulanması gerçekleştirilmiştir. Bu çalışmada oluşturulan algoritmalar, çeşitli süzgeç çekirdek genişlikleri kullanılarak 60 görüntü üzerinde sınanmıştır. Testler sırasında görüntülere çeşitli varyanslarda Gauss, düzgün ve students-t dağılımlarında gürültüler ilave edilmiş ve gürültü ortamında süzgecin performansı ölçülmüştür. Bu algoritma sonucunda elde edilen sonuçlar ve diğer yöntemlerle olan farklılıkları Bölüm 4.2'de sunulmaktadır.

Bölüm 4.1'de, bu çalışmada oluşturulmuş olan diğer bir yönteme ait sonuçlar da yer almaktadır. Ayrıca Bölüm 5'te standart Sobel işleci (yatay ve dikey yönlerde 3×3 'lük maskenin kullanıldığı), genişletilmiş Sobel işleci (yatay ve dikey yönlerde 5×5 'lik maskenin kullanıldığı), 7×7 'lik maske kullanılarak oluşturulan bağlantı haritası görüntüsü üzerine 5×5 'lik Sobel işlecinin uygulandığı yöntem ve 9×9 'lik maske kullanılarak oluşturulan bağlantı haritası görüntüsü üzerine 5×5 'lik Sobel işlecinin uygulandığı yöntemin hızları açısından karşılaştırıldığı sonuçlar yer almaktadır.

Geliştirilen algoritma, bulanık topolojinin düşük seviyeli görüntü işlemlerine olan ilk uygulamalarından biridir. Genellikle, bulanık topoloji'nin kullanıldığı alan bölütlemedir (segmentation). Geliştirilen yöntemin, temelinde Sobel işleci olması rağmen, Sobel'in sağladığı sonuçlardan farklı özelliklere sahiptir. Özellikle gürültülü görüntülerde daha başarılı sonuçlar sağlamakta ve bazı kenarlar için daha iyi sonuçlar vermektedir.

2. KENAR ALGILAMA

Kenarlar, düzgün bölgeler arasında piksel değerlerindeki ani değişikliklerin olduğu yerlerdir. Kenar algılamanın temel amacı görüntüde süreksizliklerin, diğer bir deyişle ani deęişimlerin yer aldığı bölgeleri ortaya çıkarmaktır [12]. Literatürde kullanılan bir kenar tanımı Kitchen ve Rosenfeld [13] tarafından, “her bölgenin kendi içinde homojen fakat bazı yönlerden birbirlerinden farklı olduğu komşu iki bölge arasındaki sınır” olarak yapılmıştır [1]. Bu tanıma dayanarak kenar algılamadaki amacın aslında aydınlık kenarları algılamak olduğu söylenebilir.

İnsanın görme sistemi, herhangi bir görüntü üzerinde ilk olarak nesnenin yer aldığı görüntüdeki süreksizlikleri (kenarları) algılama eğilimindedir, çünkü, durağan yerler genellikle bilgi içermez. Görüntü iyileştirme (image enhancement), bölütleme (segmentation) gibi görüntü işleme görevlerinde kenarların belirlenmesi (süreksizlikler) büyük önem taşır. Genellikle, bu gibi yüksek seviyeli görüntü işlemlerinde ön aşama olarak kenar bulma işlemleri uygulanmaktadır [14].

Bir görüntüde geometrik ve optik birçok farklı özellik kenarların oluşmasına neden olur. Kenarların belirmesindeki geometrik özellikler; i) nesne sınırları, ii) yüzey yönelim veya renk veya dokusundaki süreksizlikler, iii) iki nesnenin üst üste görünmesi nedeniyle ortaya çıkan doku ve renk farklılıklarıdır [6]. Kenarları oluşturan optik özellikler de; i) ışığın direk yansımaları, ii) aynı nesnenin bir bölümünden veya diğer nesnelere nedeniyle oluşan gölgeler, iii) aynı nesnenin bir bölümünden veya diğer nesnelere kaynaklanan ara yansımalar, iv) doku veya renk deęişimleridir [10].

Genel olarak kenar algılamada kullanılan yöntemler, uzun bir süre teorik olarak deęişiklik geçirmemiştir. Genellikle, birbirine yakın pikseller arasındaki deęer farklarını kullanırlar. Birkaç yöntemin birleştirilmesi ile oluşturulmuş [16] yöntemlerin önerilmesiyle birlikte, genel olarak temel kenar algılama algoritmaları olan Prewitt [17], Sobel [18], Canny [19], Marr ve Hildreth [20] gibi yöntemlerin en yaygın kullanılan algoritmalar olduğunu söylemek mümkündür [21-23].

Genellikle kenar algılama algoritmalarının amacı, belirgin bir şekilde görüntü üzerinde gözüken adım (step) kenarları bulmaktır. Aslında, bir kenar bulma algoritmasının görüntü üzerinde belirebilecek her türlü süreksizliği bulabilmesi gerekmektedir. Süreksizlik türleri üç farklı grupta ele alınabilir: i) adım (step) kenarlar, ii) hat (line) kenarlar, iii) kesişim (junction) kenarlar [10].

Kenar algılama algoritmaları, görüntüde yer alan kenarı bulabilmek için temel işlemler olan düzgünleştirme, türev alma ve etiketleme görevlerini gerçekleştirmeye çalışır. Bu basamaklar bütün algoritmalarda aynı görünmesine rağmen, kullandıkları düzgünleştirme filtreleri, türev işlemleri (operator), etiketleme süreçleri, amaçları, hesaplama karmaşıklıkları ve gerektirdiği matematiksel işlemler bakımından birbirleriyle farklılıklar gösterirler. Bu bölümde, yöntemlerdeki farklılıkları vurgulamak için, temel kenar işlemleri ve ardından da bileşik yöntemler konusunda bilgi verilecektir.

2.1. Yerel Kenar İşlemleri

Bir görüntü, iki değişkenli (x ve y), kesikli bir fonksiyon olarak tanımlanabilir. Kenar algılama sürecinde, bu fonksiyonda meydana gelen ani değişimler algılanır. Bu nedenle, kenarların bulunabilmesi için görüntü fonksiyonu üzerinde birinci ve ikinci türev kullanılabilir. Temel kenar işlemlerinden bazıları, aslında, türev bulan evrişim (convolution) çekirdekleridir [17-20].

Türev bulunurken gradyandan yararlanılmaktadır. Gradyan, iki boyutlu bir fonksiyonda türevle birlikte süreksizliğin yönü hakkında bilgi verebilir. Gradyanın büyüklüğüne bakılarak kenar olup olmadığı ve gradyanın açısına bakılarak kenar yönelim bilgisi elde edilir. $f(x,y)$ sürekli görüntü fonksiyonu olmak üzere, görüntü yoğunluğunun gradyan vektörü şöyle bulunabilir:

$$G[f(x, y)] = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (2.1)$$

$$= \left[\begin{array}{l} \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x, y) - f(x, y)}{\Delta x} \\ \lim_{\Delta y \rightarrow 0} \frac{f(x, y + \Delta y) - f(x, y)}{\Delta y} \end{array} \right] \quad (2.2)$$

gradyan vektörünün büyüklüğü ve yönü,

$$G = \sqrt{G_x^2 + G_y^2} \quad , \quad \theta = \tan^{-1} \frac{G_y}{G_x} \quad (2.3)$$

olarak tanımlanmaktadır. İki piksel arasındaki uzaklık en fazla bir birim olabileceğinden, $\Delta x = \Delta y = 1$ alındığında, kesikli bir görüntü için gradyan vektörü,

$$\begin{bmatrix} G_x \\ G_y \end{bmatrix} \approx \begin{bmatrix} f[i, j+1] - f[i, j] \\ f[i, j] - f[i+1, j] \end{bmatrix} \quad (2.4)$$

olarak bulunur ve buna karşılık gelen evrişim (convolution) maskeleri de,

$$G_x = \begin{bmatrix} -1 & 1 \end{bmatrix} \quad , \quad G_y = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \quad (2.5)$$

olarak hesaplanır.

2.1.1. Roberts Kenar Algılayıcı

Roberts kenar algılayıcısı, orijinal görüntü üzerinde 2×2 'lik bir maske dolaştırarak her noktada “yaklaşık” gradyan vektörünü hesaplar. Hesaplanan gradyan vektörünün büyüklükleri, görüntü büyüklüğündeki bir piksel matrisi üzerine konulursa, orijinal görüntünün kenarlarını gösteren bir görüntü elde edilebilir.

Gradyan, orijinal görüntü üzerinde yatay ve düşey yönlerde iki farklı maskenin evrişimi ile bulunur. Bu maskeler birinci türeve ait fark denklemleri kullanılarak elde edilmektedir. $f(x,y)$ sürekli görüntü fonksiyonu olmak üzere, görüntü yoğunluğunun gradyan vektörü [24],

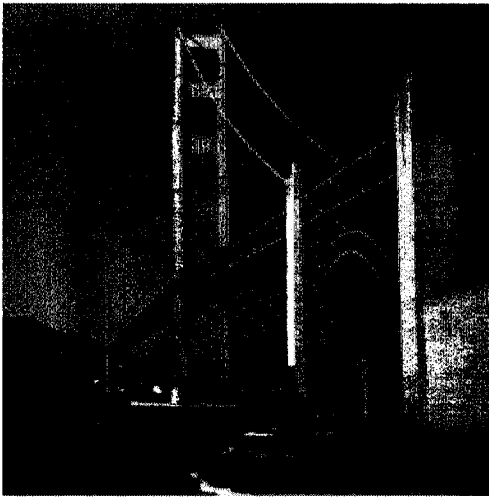
$$G|f(x, y)| = \left\{ |f(x, y) - f(x+1, y+1)|^2 + |f(x+1, y) - f(x, y+1)|^2 \right\}^{1/2} \quad (2.6)$$

olarak elde edilir. Bu eşitlikte yer alan terim katsayılarının 2×2 'lik maskeye yerleştirilmesi ile Roberts [25] kenar işleçleri oluşturulur. Yatay ve düşey yönlü evrişim maskeleri,

$$G_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \text{ ve } G_y = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \quad (2.7)$$

olarak elde edilir.

Roberts kenar algılama işleçleri 2×2 boyutunda olmaları nedeniyle hesaplanan gradyan değerleri, yaklaşık değerlerdir. Tek sayıda (3×3 , 5×5) boyuta sahip kenar işleçlerinden farklı olarak çift sayıda boyuta sahip bir işleç olması nedeniyle hesaplanan gradyan değeri maskenin merkezi yerine $f(x,y)$ ile belirtilen noktaya yerleştirilir. Bu nedenle yaklaşık gradyan değeri orijinal görüntüye yerleştirilmiş olur. Yatay ve düşey yönlerde görüntü üzerinde 2×2 boyutlu bu maskelerin evrişimi ile elde edilen kenarların yer aldığı görüntü aslında kenarların bir ölçüde ötelenmiş halinin yer aldığı görüntülerdir. Öteleme miktarı, süreksizliğin yönüyle ilgili olduğundan, ötelemeyi ortadan kaldırmak için sonradan bir işlemin yapılması zordur.



Şekil 2.1: Orijinal görüntü ve Roberts kenar algılayıcı uygulandığında elde edilen sonuç görüntüsü

2.1.2. Prewitt Kenar Algılayıcı

Prewitt kenar algılayıcısı, orijinal görüntü üzerinde her noktada gradyan vektörünü hesaplar. Hesaplanan gradyan vektörünün büyüklüğü kenar iyileştirmesi yapılmış görüntüyü oluşturur. Kenarın yönü de gradyan açısı ile bulunmaktadır [17].

Gradyan, orijinal görüntü üzerinde yatay ve düşey yönlerde iki farklı maskenin evrişimiyle bulunur. Kullanılan maskeler birinci türe ait fark denklemleri kullanılarak elde edilir. $f(x)$ fonksiyonunun birinci türevi, Taylor serisi açılımı kullanılarak elde edilmektedir. $f(x)$ fonksiyonunun $x = a$ komşuluğunda Taylor açılımı,

$$f(x) = f(a) + f'(a)(x-a) + \frac{f''(a)(x-a)^2}{2!} + \frac{f'''(a)(x-a)^3}{3!} + \dots \quad (2.8)$$

olarak ifade edilir. Bu eşitlikten yararlanılarak $f(x + \Delta x)$ 'in x etrafındaki açılımı,

$$f(x + \Delta x) = f(x) + \Delta x f'(x) + \frac{(\Delta x)^2}{2!} f''(x) + \dots \quad (2.9)$$

olarak gösterilebilir. Bu eşitliğin tekrar düzenlenmesiyle,

$$\frac{f(x + \Delta x) - f(x)}{\Delta x} = f'(x) + \frac{\Delta x}{2!} f''(x) + \dots \quad (2.10)$$

eşitliği elde edilebilir. Benzer şekilde $f(x - \Delta x)$ 'in x etrafında açılımı,

$$f(x - \Delta x) = f(x) - \Delta x f'(x) + \frac{(\Delta x)^2}{2!} f''(x) - \dots \quad (2.11)$$

ifadesini vermektedir. $f(x - \Delta x)$ ve $f(x + \Delta x)$ açılımları birbirinden çıkarıldığında,

$$f(x + \Delta x) - f(x - \Delta x) = 2\Delta x f'(x) + \frac{2(\Delta x)^3}{3!} f'''(x) + \dots \quad (2.12)$$

$$\frac{f(x + \Delta x) - f(x - \Delta x)}{2\Delta x} = f'(x) + \frac{(\Delta x)^2}{3!} f'''(x) + \dots \quad (2.13)$$

denklemleri bulunur. Bu eşitlikten yararlanılarak birinci türev nümerik olarak aşağıdaki şekilde elde edilebilir [26].

$$\frac{\partial f}{\partial x} = \frac{f(x + \Delta x, y) - f(x, y)}{\Delta x} = \frac{f(x + \Delta x, y) - f(x - \Delta x, y)}{2\Delta x} \quad (2.14)$$

$$\frac{\partial f}{\partial y} = \frac{f(x, y + \Delta y) - f(x, y)}{\Delta y} = \frac{f(x, y + \Delta y) - f(x, y - \Delta y)}{2\Delta y} \quad (2.15)$$

Yukarıdaki iki eşitlik, x ve y eksenleri yönünde gradyan büyüklükleridir. Bu ifadede, pikseller arasındaki fark olarak, $\Delta x = \Delta y = 1$ alındığında türevde yer alan katsayıların 3×3 'lük bir evrişim çekirdeği oluşturacak şekilde genişletilmesi ile Prewitt evrişim maskeleri,

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \text{ ve } G_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad (2.16)$$

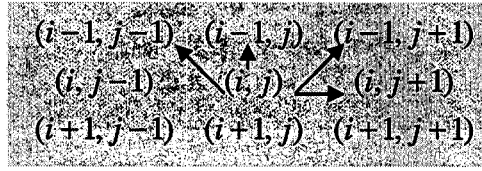
elde edilir [26].



Şekil 2.2: Orijinal görüntü ve Prewitt kenar algılayıcı uygulandığında elde edilen sonuç görüntüsü

2.1.4. Sobel Kenar Algılayıcı

Sobel kenar algılama işlecinin geliştirilmesindeki amaç, Robert işleci [25] gibi yaygın kullanılan işleçlere göre yönden daha bağımsız gradyan değerlendirmesi ile verimli bir hesaplama yapabilmektir [27]. Sayısal bir görüntü, bir fonksiyon olarak değerlendirildiğinde, bir nokta üzerindeki gradyan değerinin, 3×3 komşulukta mümkün olan dört merkezi yönde (Şekil 2.3) elde edilebilir gradyan değerlerinin vektör toplamları şeklinde oluşturulması düşüncesine dayanır. Bu yaklaşımla Sobel kenar işleci oluşturulmuştur.



Şekil 2.3: Dört temel yönde gradyan vektörleri

Bu gradyan değerlerinin vektör toplamları, gradyan ölçümlerinin yönü (directions of measurement) üzerinde ortalama değerin bulunmasını sağlar. Eğer yoğunluk fonksiyonu gerçekten düzlemsel ise, o noktanın etrafındaki bütün yakın komşuluklardaki tüm gradyanlar aynı değere sahip olur.

3×3 'lük komşuluk için merkez noktanın gradyan değeri, dik vektör çiftlerinin vektör toplamları olarak bulunur (Denklem 2.18). Burada kullanılan gradyan vektörleri, (i, j) elemanı merkezde olmak üzere Şekil 2.3'de verilmektedir. Her dik vektör, yönsel türev değerinin türev yönünü belirleyen birim vektör ile çarpımı ile elde edilir.

a	b	c
d	e	f
g	h	i

Şekil 2.4: 3×3 komşulukta yer alan piksellerin yerleşimi

Dört gradyan değerinin vektörel toplamı, sekiz yönsel türev vektörünün vektörel toplamıyla aynıdır. Bir nokta ve onun sekiz komşuluğunun yoğunluk değerleri, Şekil 2.4'te gösterilmektedir. Verilen komşuluklar için; yf , yoğunluk farkı, k , komşular arası uzaklık olmak üzere, g , yönsel türev değer vektörünün büyüklüğü

$$|g| = yf / k \quad (2.17)$$

olarak verilir. Birbirine karşılık gelecek şekilde komşu noktalar (a, i) , (b, h) , (c, g) ve (f, d) olarak gruplanırsa, $R = \sqrt{2}$ için (köşegenlerdeki piksellerin merkez noktadan uzaklığı) gradyan değerlerinin vektörel toplamı,

$$G = \frac{(c-g)}{R} \cdot \frac{[1,1]}{R} + \frac{(a-i)}{R} \cdot \frac{[-1,1]}{R} + (b-h) \cdot [0,1] + (f-d) \cdot [1,0] \quad (2.18)$$

olarak ifade edilir. Bu işlemin sonucunda,

$$G = [(c-g-a+i)/2 + f-d, (c-g+a-i)/2 + b-h] \quad (2.19)$$

olarak sonuç vektörü elde edilmektedir. Buradaki 1/2 çarpanından kurtulmak için (tamsayı işlemi yapabilmek için) vektör ikiyle çarpılır ve sonuç olarak,

$$G' = 2.G = [(c-g-a+i) + 2.(f-d), (c-g+a-i) + 2.(b-h)] \quad (2.20)$$

vektörü elde edilir. Bu vektör, yoğunluk ağırlıklarının toplamı olarak Şekil 2.5'te gösterilen düşey ve yatay yönlerde ki ağırlık fonksiyonları kullanılarak gösterilir.

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

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

Şekil 2.5: Yatay ve düşey yöndeki ağırlık katsayıları

Bu algoritma, T önceden belirlenmiş bir eşik seviyesi olmak üzere,

$$|G'|^2 > T. \quad (2.21)$$

şartını sağlayan noktaları kenar noktası olarak belirleyen bir kenar noktası algılayıcısı olarak kullanılır [28].

Sobel'in 3×3 komşuluğundaki noktalar için hesaplanmış olan gradyan tanımı 5×5 komşuluk için de uygulanabilir. Bu durumda dört yerine değişik konumlardaki oniki gradyan değerinin hesaplaması gerekir. Bu değerlerin vektörel toplamı ile gradyan değeri elde edilir. 5×5 'lik komşulukta, bir nokta ve onun yirmidört komşuluğunun yoğunluk değerleri Şekil 2.6'te gösterilmektedir.

a	b	c	d	e
f	g	h	i	j
k	l	m	n	o
p	r	s	t	u
v	w	x	y	z

Şekil 2.6: 5×5 komşulukta yer alan piksellerin yerleşimi

Tüm yönlerdeki gradyan değerlerinin vektörel toplamı, $R_1 = \sqrt{2}$, $R_2 = \sqrt{5}$, $R_3 = \sqrt{8}$ ve $R_4 = 2$ olmak üzere,

$$\begin{aligned}
 G = & \frac{(i-r)}{R_1} \cdot \frac{[1,1]}{R_1} + (h-s) \cdot [0,1] + \frac{(g-t)}{R_1} \frac{[-1,1]}{R_1} + (n-l) \cdot [1,0] + \frac{(o-k)}{R_4} \cdot \frac{[2,0]}{R_4} \\
 & + \frac{(e-v)}{R_3} \cdot \frac{[2,2]}{R_3} + \frac{(c-x)}{R_4} \cdot \frac{[0,2]}{R_4} + \frac{(a-z)}{R_3} \cdot \frac{[-2,2]}{R_3} + \frac{(j-p)}{R_2} \cdot \frac{[2,1]}{R_2} \\
 & + \frac{(d-w)}{R_2} \cdot \frac{[1,2]}{R_2} + \frac{(b-y)}{R_2} \cdot \frac{[-1,2]}{R_2} + \frac{(f-u)}{R_2} \cdot \frac{[-2,1]}{R_2} \quad (2.22)
 \end{aligned}$$

eşitliği ile ifade edilebilir.

Sonuç vektörü,

$$\begin{aligned}
 G = & [(n-l) + \frac{1}{2} \cdot (i-r-g+t+o-k) + \frac{1}{4} \cdot (e-v-a+z) + \frac{1}{5} \cdot (d-w-b+y) \\
 & + \frac{2}{5} \cdot (j-p-f+u), (h-s) + \frac{1}{2} \cdot (i-r+g-t) + \frac{1}{4} \cdot (e-v+a-z) \\
 & + \frac{1}{5} \cdot (j-p+f-u) + \frac{2}{5} \cdot (d-w+b-y)] \quad (2.23)
 \end{aligned}$$

olarak bulunur. Bu denklemdeki kesirli terimlerden kurtulmak için vektör 20 ile çarpılırsa,

$$\begin{aligned}
 G' = & 20.G \\
 = & [20(n-l) + 10(i-r-g+t+o-k) + 5(e-v-a+z) + 4(d-w-b+y) \\
 & + 8(j-p-f+u), 20(h-s) + 10(i-r+g-t) + 5 \cdot (e-v+a-z) \\
 & + 4(j-p+f-u) + 8(d-w+b-y)] \quad (2.24)
 \end{aligned}$$

formülü elde edilir.

Bu eşitlikteki katsayılar kullanılarak, elde edilen yatay ve düşey yönlerdeki maske katsayıları Şekil 2.7'de verilmektedir.

-5	-4	0	4	5
-8	-10	0	10	8
-10	-20	0	20	10
-8	-10	0	10	8
-5	-4	0	4	5

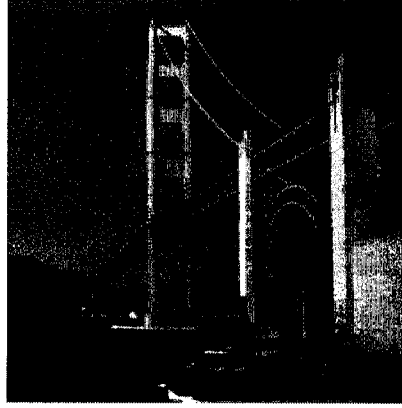
5	8	10	8	5
4	10	20	10	4
0	0	0	0	0
-4	-10	-20	-10	-4
-5	-8	-10	-8	-5

Şekil 2.7: Yatay ve düşey yöndeki ağırlık katsayıları

Sobel işlecine ait yatay ve düşey yönlü maskeler, görüntü üzerinde ayrıca gezdirilerek sonuç görüntüsü elde edilir. Görüntüde sol üst köşeden başlayarak işleç maskesi görüntü üzerine konular. Her piksel değerine karşılık gelen maske katsayısıyla, piksel değeri çarpılarak, tüm sonuçlar toplanır ve işlecin tepkisi bulunur. Orijinal görüntüde yatay ve düşey yönlü maskelerin aynı yerlerdeki tepkilerinin kareleri toplanır ve karekökü alınarak, gerçek gradyan değeri elde edilir. Gradyan değerleri, orijinal görüntü büyüklüğündeki bir matrisse yerleştirildiğinde, orijinal görüntünün kenarlarını gösteren bir görüntü bulunmuş olur (Şekil 2.8).

Bir türevsel kenar algılama işleci olan, Sobel işlecinin boyutunun ve katsayılarının sabit olması nedeniyle, görüntünün özelliklerine göre uyarlanamaz. Ayrıca gürültüye karşı oldukça hassastır. Gürültü hassasiyeti, evrişim işleminden önce bir düzgünleştirme işlemi gerektirir. Düzgünleştirme işlemiyle, kenarlar bozulmadan düzgün alanlardaki gürültü ortadan kaldırılmaya çalışılır [10]. Bu aşamada meydana gelebilecek bilgi kayıpları nedeniyle, görüntüdeki kenarların bazılarının bulunmasında sorunlar yaşanabilir [10].

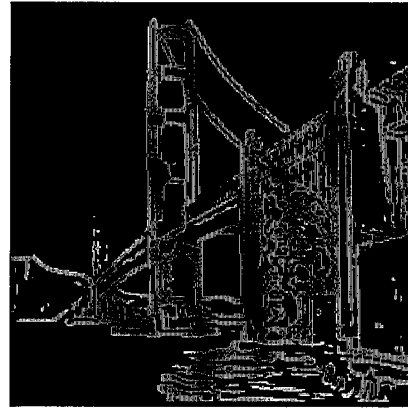
Sobel işleci kullanıldığında elde edilen gradyan matrisi üzerinde bir eşikleme (thresholding) ve ardından bir inceltme işlemi gerekmektedir [10]. Bu algılayıcıya ait yatay ve düşey yönlü maskelerin boyutu büyüdükçe bulunan kenarlar kalınlaşır. Maskenin boyutu büyüdükçe, maske içine giren kenar sayısı da artmaktadır. Bu nedenle, hesaplanan yönlü türevlerin etkisi fazlalaşır ve daha fazla kenar algılanır. Kenarların kalınlaşması, kenarın yöneliminin değiştiği noktalar ve köşeler gibi ufak detayların kaybolmasına neden olabilmektedir. Buna rağmen işlem kolaylığı nedeniyle, yaygın bir şekilde kullanım alanı bulmaktadır [22].



(a)



(b)



(c)

Şekil 2.8: (a) Orijinal görüntü, (b) 3×3 'lük Sobel maske uygulanmış görüntü sonucu, (c) 5×5 'lik Sobel maske uygulanmış görüntü sonucu

2.1.4. Canny Kenar İşleci

Canny [29] optimal kenar bulucunun elde edilebilmesi için gerekli koşulları belirlemeye çalışmıştır. Optimal kenar algılayıcının sahip olması gereken özellikleri şöyle belirtir:

- 1) Kenarların var olduğu yerleri büyük olasılıkla belirlemeli ve kenar olmayan yerleri düşük olasılıklarla belirleyebilmelidir;
- 2) Kenarın gerçek yerleşimine en yakın olacak şekilde işaretlenen kenar piksellerini yerleştirebilmelidir;
- 3) Bir kenara sadece bir kez tepki vermelidir.

Canny iki boyutlu görüntüler için iki farklı türevsel işleci; gradyan boyunca yönlü ikinci türevi,

$$\frac{\partial^2}{\partial n^2} \quad (2.25)$$

ve türevsel işleci (Gauss) birbirinin yerine seçmeli olarak kullanır [23].

Gauss filtrelenmiş görüntü;

$$f_s(i, j) = G(i, j, \sigma) * f(i, j) \quad (2.26)$$

olmak üzere, gradyan hesaplandığında

$$\nabla f_s = \left[\frac{\partial f_s}{\partial x}, \frac{\partial f_s}{\partial y} \right]^t \quad (2.27)$$

elde edilir. Gradyan fonksiyonunun hesaplanmasından sonra, gradyanın büyüklüğü ve yönü;

$$m(i, j) = \sqrt{\left(\frac{\partial f_s}{\partial x} \right)^2 + \left(\frac{\partial f_s}{\partial y} \right)^2}, \quad \theta(i, j) = \arctan\left(\frac{\partial f_s / \partial y}{\partial f_s / \partial x} \right) \quad (2.28)$$

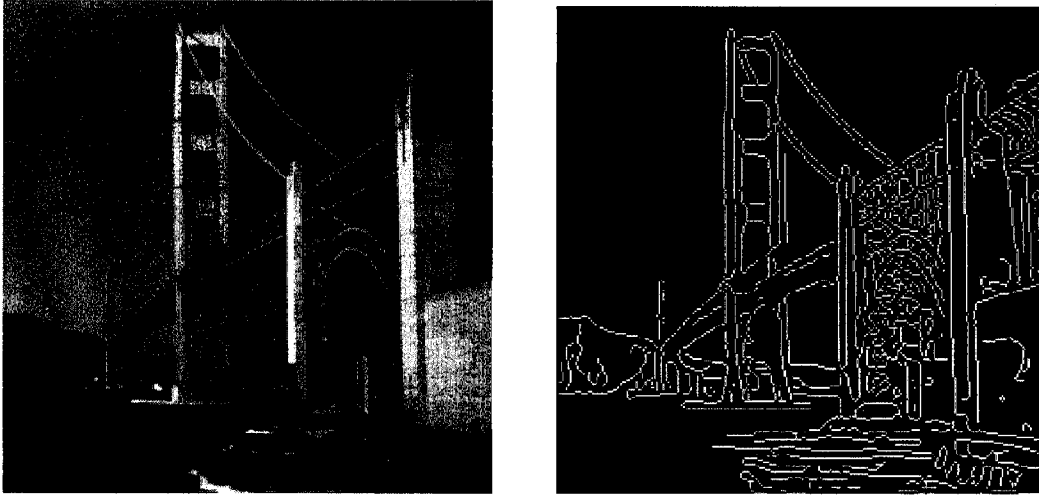
formülleriyle belirlenir [9,12].

Gradyan büyüklüğündeki yamaçlar, maksimum olmayan işaretlenmiş bölgelerin bastırılması ile inceltildikten sonra yanlış kenar bölütlerinin azaltılması için elde edilen görüntüye eşik seviyeleme uygulanır [9].

Canny işleci tek değildir. Kenar şekline (adım kenar, rampa kenar gibi) göre Canny kenar algılama değişim gösterir. Adım kenarlar için kullanılan optimal kenar algılama işleci, LoG [20] işlece benzer [21]. Ayrıca, Canny kenar algılayıcı, kenar düzensizliklerini ortadan kaldırmak için histerez (hysteresis) ile uyarlamalı eşikleme tekniğini kullanır. Canny uygulanmış görüntüye örnek, Şekil 2.9'da verilmektedir.

Canny tarafından oluşturulan algoritmada bir pikselin değeri, komşuluğunda yer alan piksellerin değerinden büyük ise nokta kenar olarak işaretlenir. Gürültülü görüntülerde bu durum yanlış kenar algılama sorunu doğurabilir [21].

Canny algılayıcısı zayıf kenarları bulmak konusunda oldukça hassastır. Ancak bu özellik, algoritmayı, belirsiz değişimlerin olduğu yoğunluklarda (düzgünce gölgelenmiş nesnelere, bulanıklaşmış sınırlar gibi) yani kararsız ve parazitli kenarların algılanmasında oldukça sorunlu yapar [21].



Şekil 2.9: Orijinal görüntü ve Canny kenar algılayıcı uygulandığında elde edilen sonuç görüntüsü.

2.1.5. Marr-Hildreth Kenar Algılayıcı

Marr ve Hildreth [20] , 1979 da Marr ve Poggio tarafından yapılan çalışmayı genişleterek 1980 yılında bir kenar algılama işleci oluşturmuşlardır [30]. Bu kenar algılayıcı, Gauss filtreleme ve ardından Laplas uygulanması prensibine dayanır. Bu kenar algılayıcı Laplacian of Gaussian (LoG) olarak da bilinir. Filtrenin frekans tepkisinin şeklinden dolayı “Meksika Şapkası” işleci olarak da bilinmektedir.

Gauss filtrelemeyle, görüntüde yer alan yüksek frekans etkileri ortadan kaldırıldıktan sonra Laplas uygulanması ile sıfır geçişlerinin bulunması gerçekleştirilir. Laplacian of Gaussian (LoG) işleci,

$$h(x, y) = \nabla^2 [g(x, y) * f(x, y)] = [\nabla^2 g(x, y)] * f(x, y) \quad (2.29)$$

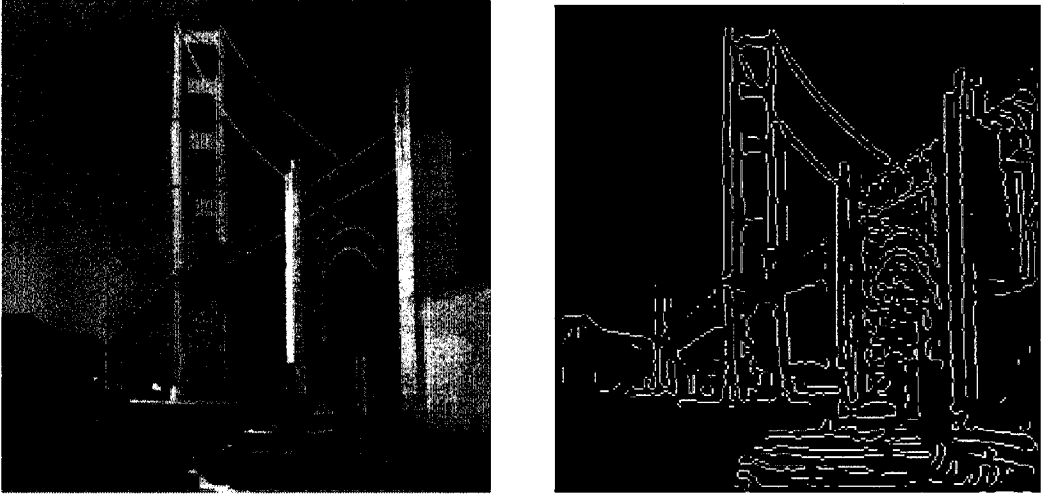
$$\nabla^2 g(x, y) = \left(\frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} \right) e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (2.30)$$

olarak ifade edilebilir.

LoG filtre, yapısında yer alan Gauss fonksiyonunun görüntüde yarattığı bulanıklaşma (blurring) nedeniyle, görüntülerde gerçek yerleşimini değiştirmiş kenarları algılar. Bunun sonucunda orijinal görüntüde olan fakat filtrenmiş

görüntüde gözlenemeyen silikleşmiş kenarlar LoG filtrelerin en büyük sorunudur [21]. Bunun nedeni LoG filtreler gerçekleştirirken, gerçek kenarlar aslında sadece yerel maksimumlarda yer almasına rağmen, görüntü fonksiyonunun birinci türevine ait yerel maksimum ve minimumların gözlenmesidir.

LoG filtre ile elde edilmiş “Golden Gate” görüntüsü kenarları Şekil 2.10’da gösterilmektedir. Diğer kenar işleçleriyle kıyaslandığında, görüntüdeki kenarların ve bazı detayların yok olduğu görülmektedir.



Şekil 2.10: Orijinal görüntü ve LoG kenar algılayıcı uygulandığında elde edilen sonuç görüntüsü.

2.2. Diğer Kenar Algılayıcılar

Bölüm 2.1’de bahsedilen temel işlemler haline gelmiş kenar algılayıcılar yanında, farklı kenar algılama algoritmaları da türetilmiştir. Bazı temel kenar algılama yöntemlerini, örneğin Sobel, Prewitt, Canny [17-19] gibi işlemler merkez alınarak, daha dayanıklı kenar algılamanın nasıl gerçekleştirilebileceği üzerinde çalışılmıştır [3-8]. Bu kenar algılama yöntemlerinin çok sayıda olması nedeniyle, sadece önemli görülen yöntemler bu bölümde açıklanacaktır.

2.2.1. Nesne-Temelli Kodlama için Dönüşüm Alanı Tekniğini Kullanarak Kenar Algılama

Prewitt, Sobel ve Canny gibi kenar algılama yöntemleri evrişime dayanırlar ve durağan görüntülerdeki kenarları algılamak için uygundur. Bu teknikler kenarların istatistiksel davranışlarını kullanırlar. Bu nedenle nesne-temelli kodlama içeren video kodlama sistemlerinde görüntünün kodlanması ve sıkıştırılması gibi durumlardan meydana gelen sorunların bulunduğu görüntülerdeki kenarları bulamazlar.

Suthaharan, nesnenin anlamlı bölümlerini belirlemek için dönüşüm katsayılarındaki kontrast ve frekans bilgilerini kullanmıştır [3]. Bu kenar algılama yöntemiyle, kesikli kosinüs dönüşüm (DCT) katsayıları (denklem 2.31) ile birlikte katsayıların insan görme sistemi (HVS) ağırlıklarının (denklem 2.32), kenar kontrast bilgisi ve algısal özelliklerini temel alan, görüntü ve kenar detaylarını algılamak için dönüşüm alanı tekniği ileri sürülmektedir. HVS ağırlıklarının amacı, kontrast seviyelerini temel alan kenarları ayırmak ve insan algılaması için görünür hale getirmektir.

İlk olarak DCT katsayıları, özellikle frekans duyarlılığı ve kenar kontrast bilgisi gibi algısal özelliklere bağlı olarak birkaç gruba ayrılır. Ayrılan bu gruplar HVS'den bağımsız olarak düşük frekans katsayılı kenar detaylarını yüksek frekanslı katsayılardan ayırmak için kullanılır. Bu şekilde, insanın görme sistemi ağırlıkları (HVS) kullanılarak, algısal olarak en belirgin kenar ve görüntü ayrıntıları vurgulanarak görüntü oluşturulur.

Suthaharan tarafından yapılan bu çalışmada önerilen algoritma aşağıdaki şekilde özetlenebilir. X ve Y , I görüntüsünde yer alan $n \times n$ piksele sahip görüntü bloklarını, a , X 'in DCT katsayılarını ve $x(i, j)$, (i, j) 'nin yoğunluk değerini göstermek üzere,

$$a = \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^n x(i, j) \quad (2.31)$$

olarak, X görüntü bloğunun uzamsal (spatial) yoğunluk değerleri ile DCT katsayıları gösterilmiştir. U , X 'in AC yani görünemeyen kenar ve görüntü

detaylarına ait katsayıları ve sıfır DCT katsayılarını, T , $n \times n$ 'lik birimcil (unitary) dönüşüm matrisini göstermek üzere,

$$X = T * U * T + \frac{a}{n} \begin{bmatrix} 1 & \dots & 1 \\ \dots & \dots & \dots \\ 1 & \dots & 1 \end{bmatrix}_{n \times n} \quad (2.32)$$

eşitliği elde edilir [24].

Frekans spektrumunda, alçak veya orta değerli bileşenlerde kenar bilgisi diğer görüntü bilgileri ile karıştığından kenarları algılamak güçleşebilir. Bu nedenle, kenar bilgisini genellikle yüksek frekanslı katsayılardan elde etmek daha iyi sonuçlar verir. Önerilen yöntemle, alçak, orta ve yüksek frekans katsayılarına uygun HVS katsayıları uygulanarak bu katsayılar arasındaki nümerik farklar alçaktan yükseğe doğru üstel olarak genişletilir ve kenar detayları diğer görüntü bilgilerinden ayrılır. Böylelikle saklı kenarlar daha belirgin hale getirilir [3].

Önerilen diğer bir yöntem olan, geliştirilmiş HVS modeli, a, b, c HVS eğrisinin şeklini saptamak için sabitler ve f ışınal frekans olmak üzere,

$$\tilde{H}(f) = (a + b \times f) \times e^{(c \times f)} \quad (2.33)$$

olarak verilir. Burada yer alan HVS modeli, gerçekteki modelin tersidir ve bu nedenle IHVS (ters HVS) olarak adlandırılmıştır.

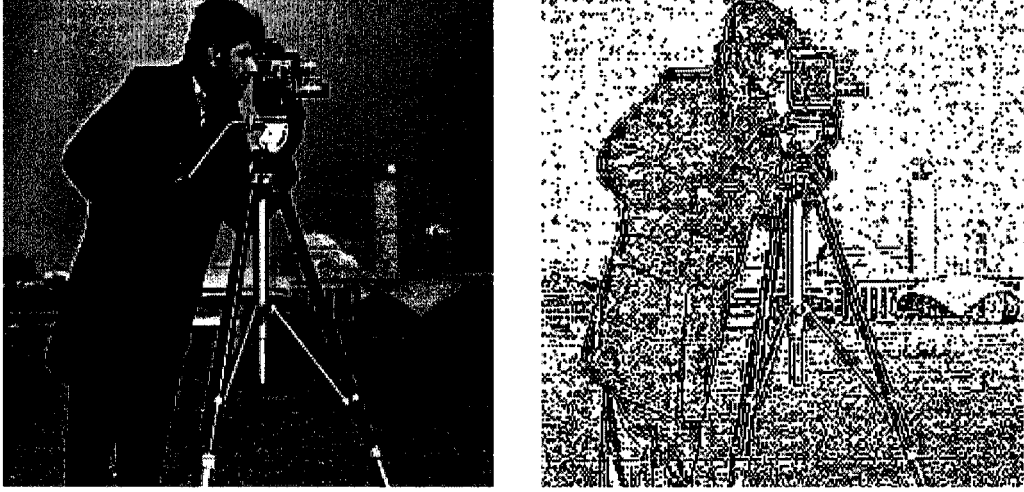
Bu uygulamada alçak frekans katsayıları ve yüksek frekans katsayıları arasındaki bağıl kenar etkilerini temel alan kenar bilgileri algılanır. Böylece denklem 2.34'te tanımlandığı gibi X 'in dönüşüm katsayılarının değiştirilmesi ile yeni görüntü bloğu X_1 tanımlanır.

$$X_1 = T * (\tilde{H} * U) * T + \frac{\alpha \cdot a}{n} \begin{bmatrix} 1 & \dots & 1 \\ \dots & \dots & \dots \\ 1 & \dots & 1 \end{bmatrix}_{n \times n} \quad (2.34)$$

α , X_1 'in DC seviyesini ayarlayarak X_1 bloğunun yoğunluk ortalamasına göre değişik seviyelerde görüntü ve kenar detaylarını elde etmek için kullanılan bir katsayıdır. α 'nın büyük olması algılanan kenarların sayısının azalmasına neden olur. Diğer yandan yüksek kontrastlı kenar detayları korunur.

Suthaharan [3] geliştirdiği algoritmanın, bölütleme için gerekli kenar detayları yanında görsel olarak anlamlı kenarlar belirlediğini ve genellikle görüntü

ve video işleme uygulamaları için elverişli kenar detayları da sağladığını ileri sürmektedir.



Şekil 2.11: Orijinal görüntü ve Suthaharan tarafından oluşturulan yöntemin sonucu [3]

2.2.2. Gerçek-Zamanlı Kenar Algılayıcı

Çeşitli kenar algılama işlerinde çarpma, bölme ve karekök gibi büyük zaman gerektiren yüksek-ranklı operasyonlar kullanılmaktadır [4]. Gerçek zamanlı uygulamalarda, genellikle, zaman kısıtlaması olduğu için, algoritmalarda zaman harcayan aşamaların kullanılmaması yararlı olmaktadır. Gerçek-Zamanlı Kenar Algılayıcı algoritmasının çalışma ilkesi güçsüz kenarlar bulmak ve her kenarda tek piksellik bir büyüklük elde etmek olarak verilebilir.

Çalışmada tasarlanan tam fark maskesi algoritması (absolute difference mask - ADM), yaygın olarak kullanılan algoritmalarla karşılaştırıldığında yüksek-ranklı operasyonlar gerektirmemesine rağmen oldukça iyi sonuçlar üretebilen bir kenar bulucu algoritmasıdır. Yöntemin donanım olarak sayısal VLSI (very large scale integration)'larda gerçeklemeleri yapılmıştır.

Algoritma gürültülü veya gürültüsüz görüntülerde üç adımda kenarları bulur: i) Gürültü etkilerini azaltmak için yarı-Gauss filtre (Şekil 2.12) uygulanır; ii) her piksel için kenar şiddeti ve yönü bulunur; iii) sonuç kenar haritası

oluşturulur. Tek piksel genişliğinde, inceltme için kullanılan süreçlere gerek duyulmadan kenarlar üretilir ve yerleştirilir.

$$\frac{1}{16} \times$$

0.25	0.50	0.50	0.5	0.25
0.50	0.75	1.00	0.75	0.50
0.50	1.00	2.00	1.00	0.50
0.50	0.75	1.00	0.75	0.50
0.25	0.50	0.50	0.50	0.25

Şekil 2.12: Yarı-Gauss düzgünleştirme maskesi

Yarı-Gauss düzgünleştirme maskesinde, W_p piksel ağırlıklarını ve d_p , maskenin merkezinden p pikselinin uzaklığını göstermek üzere, piksel ağırlıkları,

$$W_p = \frac{1}{d_p} \quad (2.35)$$

formülüyle hesaplanır. Kenar şiddetini hesaplamak için kullanılan mutlak-fark maskesi Şekil 2.13'te verilmektedir. $p_{i,j}$ 'nin kenar şiddeti dört farklı yönde hesaplanıp en büyük şiddet pikselin kenar şiddeti olarak atanır. Bu dört yön: negatif köşegen yönü ($yön=1$), dikey yönü ($yön=2$), yatay yönü ($yön=3$), pozitif köşegen yönü ($yön=4$).

1		3		5
Pdu(2)		Vu(2)		Ndu(2)
	7	8	9	
	Pdu(1)	Vu(1)	Ndu(1)	
11	12	13	14	15
Hl(2)	Hl(1)	P(i,j)	Hr(1)	Hr(2)
	17	18	19	
	Ndl(1)	Vl(1)	Pdl(1)	
21		23		25
Ndl(2)		Vl(2)		Pdl(2)

Şekil 2.13: Mutlak-fark maskesi

Kenar şiddetini ve yönünü bulmak için mutlak-fark maskesi aşağıdaki adımlarla oluşturulur.

1. $p_{(i,j)}$ 'nin mutlak farkları bulmak için girdilerin hazırlanması:

$$V_u = V_u(1) + V_u(2), \quad V_l = V_l(1) + V_l(2) \quad (2.36)$$

$$H_r = H_r(1) + H_r(2), \quad H_l = H_l(1) + H_l(2) \quad (2.37)$$

$$Pd_u = Pd_u(1) + Pd_u(2), \quad Pd_l = Pd_l(1) + Pd_l(2) \quad (2.38)$$

$$Nd_u = Nd_u(1) + Nd_u(2), \quad Nd_l = Nd_l(1) + Nd_l(2) \quad (2.39)$$

2. $p_{(i,j)}$ için tüm mutlak farkların hesaplanması:

$$V = |V_u - V_l|, \quad H = |H_r - H_l| \quad (2.40)$$

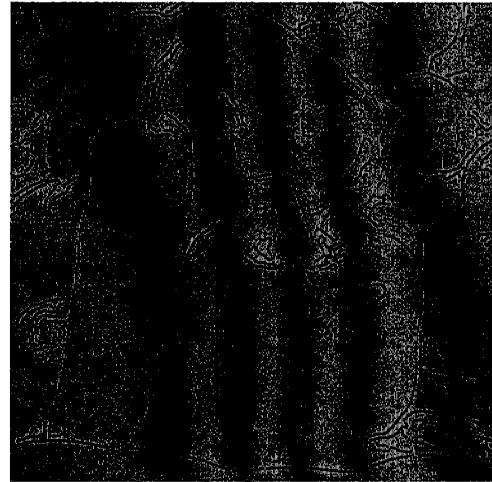
$$Pd = |Pd_u - Pd_l|, \quad Nd = |Nd_u - Nd_l| \quad (2.41)$$

3. Kenar şiddeti ve yönünün belirlenmesi:

$$S_e = \max\{V, H, Pd, Nd\} / 2 \quad (2.40)$$

$$yön_e = yön(\min\{V, H, Pd, Nd\}) \quad (2.41)$$

Diğer kenar algılama algoritmaları sonuçta elde edilen görüntü üzerinde ek olarak bir inceltme işlemine gerek duymalarına rağmen, söz konusu yöntem kenar algılama, yerleştirme ve inceltme işlemlerini birlikte gerçekleştirebilmektedir.



Şekil 2.14: Orijinal görüntü ve gerçek zamanlı kenar algılama yöntemine [4] ait sonuç görüntüsü

2.2.3. Doğrusal Olmayan Laplas Filtrelemeyi Temel Alan Kenar Algılayıcı

Vliet ve diğerleri [5] tarafından önerilen bu kenar algılama tekniği doğrusal olmayan Laplas işleç ile Marr-Hildreth kenar algılayıcı modelinin birleştirilmesi ile oluşturulmuştur.

Doğrusal olmayan Laplas filtre, minimum ve maksimum filtreler kullanılarak gerçekleştirilir. Geleneksel Laplas filtre her yatay ve düşey eksen boyunca ikinci türevi hesaplamasına rağmen doğrusal olmayan Laplas filtre bu yönelimleri, maksimum ve minimum yönünde gradyanları kullanarak, yerel gradyan yönlerine göre değiştirir [6]. Doğrusal olmayan Laplas işleç sonucunda görüntüdeki mümkün olan kenar şiddetleri yansıtılır. Bu algoritma ile algılanan kenarlar gerçek ve hayali (phantom) olmak üzere ikiye ayrılır.

Bu yöntemde, ilk olarak görüntüdeki yüksek frekans etkileri alçak geçiren filtre uygulanarak azaltılır. Daha sonra mümkün olan kenarları yerleştirebilmek için analitik olmayan ikinci türev filtre ile ikinci türevdeki sıfır geçişleri hesaplanır [5].

Gradyan yönündeki ikinci türev (SDGD), yerel maksimum ve minimum filtrelerin orijinal görüntüye uygulanması ile oluşturulur. Burada orj orijinal görüntüyü, N maske boyutunu göstermektedir.

$$SDGD(orj) = (\min_N(orj) - orj) + (\min_N(orj) - orj) \quad (2.42)$$

Maksimum filtre, çıktı görüntüde karanlık bölgelerin kapladıkları alanı azaltırken, minimum filtre bu bölgeleri genişletir. Filtrelenmiş görüntülerden orijinal görüntünün çıkarılmasıyla, minimum ve maksimum yönlerde gradyanı gösteren sonuç elde edilir.

Analitik olmayan Laplas filtrenin orijinal görüntüye uygulanmasından sonra, 8-komşulukta sıfır geçiş algılayıcısı, nesne sınırlarının mümkün olan tüm yerleşimlerini yansıtan bir görüntü yaratmak için kullanılır. Bu kenar şiddeti görüntüsü, bulanık ve minimum işleçleri temel alan (BAM filtresi) [7] kenar şiddet algılayıcıları kullanılarak yaratılır. n , maske boyutunu ve “.” mantıksal ve işlemini göstermektedir.

$$BAM(orj) = \min(\max_n(orj) - orj \cdot orj - \min_n(orj)) \quad (2.43)$$

Doğrusal olmayan işlecin hesaplama süresi maske boyutundan bağımsız olduğundan bu işlecin, kenar algılama için daha büyük maskeler gerektiren görüntüler için daha kullanışlı olduğu ileri sürülmektedir. Ancak filtre boyutunun artmasıyla algılanan kenarların azaldığı gözlenir. Büyük filtrelerin kullanılmasının güçlü kenarların bulunmasını sağladığı söylenmektedir.

İşleç çözünürlüğü ve yerleşimi değiştiğinde Marr-Hildreth işleci analitik tanımlamadan yeni örneklerin alınmasını gerektirirken, önerilen yöntemde sadece filtrenin parametrelerini değiştirmek yeterli olmaktadır. Bu nedenle doğrusal olmayan Laplas kenar algılayıcı çoklu ölçekli çözünürlüklerde ideal işleç olarak uygulanabilmektedir.



Şekil 2.15: Orijinal görüntü ve doğrusal olmayan Laplas yöntemi elde edilen sonuç görüntüsü [5]

2.2.4. Kesikli Tekil Evrişim (DSC) Algoritması

Hou ve Wei [8] tarafından, çok ölçekli kenarları algılamak amacıyla oluşturulan yeni kenar algılama algoritması iki sınıfa ayrılmıştır: i) DSC kenar bulucu (DSCED), ii) Gürültüye karşı DSC (DSCANED). DSCED, görüntüdeki küçük detayları bulabilirken DSCANED gürültüye karşı daha dayanıklıdır. Bu iki sınıfın birleşmesi ile oluşmuş olan DSC kenar algılayıcı, çok ölçekli kenarlar için verimli ve güvenilir bir çözüm sağlar.

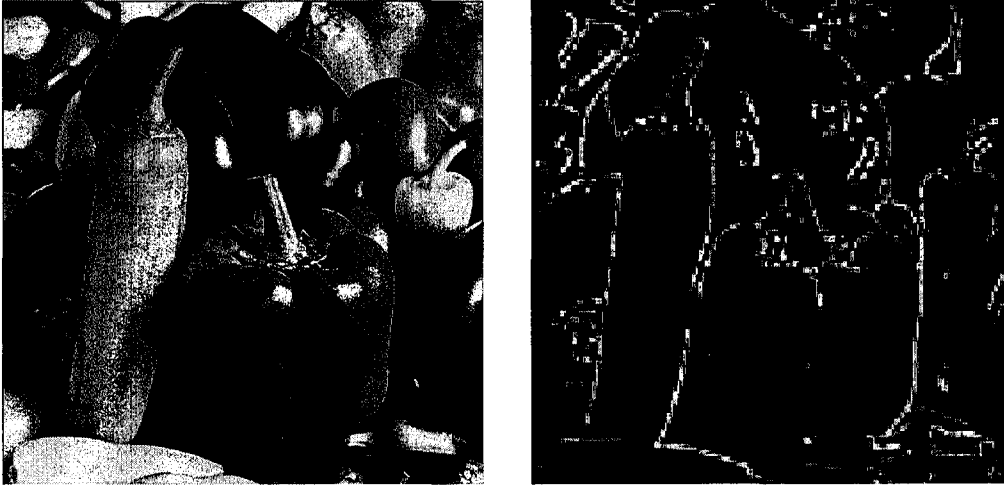
Matematiksel olarak dağılım kuramı ve dalgacık (wavelet) analizine dayanan DSC algoritmasının, uygulama olarak özel bir tipi olan fark tipi eğriler kullanılarak bu kenar algılama yöntemi oluşturulur. Dalgacık fonksiyonu integrali sıfır olan bir fonksiyondur. İlk olarak fonksiyon gösterimi olarak oldukça artıklı (redundant) bir dönüşümdür. İkinci olarak fonksiyon kenarda (keskin değişimlerde) yer alan süreksizliğin derecesini göstermek için kullanılır.

Gürültüsüz görüntüler için n 'inci dereceden ince-ölçekli (fine-scale) DSC kenar algılayıcı (DSCEDⁿ), I sayısal görüntü, $\delta_{\sigma,\alpha}^{(n)}(x_m - x_k)$, n 'inci dereceden DSC çekirdeğinin türevi, W hesaplama band genişliği ve $n=1,2,\dots$ olmak üzere,

$$DSCED^n(x_i, y_i) = \left| \sum_{k=-W_n}^{W_n} \delta_{\sigma_n, \alpha_n}^{(n)}(x_i - x_k) I(x_k, y_j) \right| + \left| \sum_{l=-W_n}^{W_n} \delta_{\sigma_n, \alpha_n}^{(n)}(y_j - y_l) I(x_i, y_l) \right| \quad (2.44)$$

olarak verilir. Bu kenar algılayıcı tüm ölçeklerde, gürültüye karşı duyarlı olmasına rağmen ince ayrıntıları çıkarma kabiliyetine sahiptir.

Kaba(büyük) ölçekli (coarse-scale) kenar algılayıcıların gürültü altında çalışmalarına rağmen ince detaylara karşı duyarlı olmadığı bilinmektedir [8]. İnce ayrıntıların belirlenmesi için çok sayıda ölçekte çalışmak gereklidir. İnce ayrıntılar, yüksek çözünürlüklü ifadelerde görülürken, belirgin detaylar düşük çözünürlüklü görüntülerde belirirler.



Şekil 2.16: Orijinal görüntü ve DSCED algoritması sonucu elde edilen görüntü [8]

Çok ölçekli kenarları elde etmenin bir yolu çok çözünürlüklü dalgacık analizini kullanmaktır. Bir başka yol ise filtreleme ve kenar algılamının birleşimi ile oluşturulmuş kenar algılayıcılar kullanmaktır. Söz konusu çalışmada gürültüye karşı kenar algılayıcı (DSCANEDⁿ) veya n 'inci dereceden kaba DSC kenar algılayıcı önerilmiştir. ($n=1,2,\dots$)

$$\begin{aligned}
DSCANED^n(x_i, y_j) = & \left| \sum_{k=-W_n}^{W_n} \sum_{l=-W_0}^{W_0} \delta_{\sigma_n, \alpha_n}^{(n)}(x_i - x_k) \delta_{\sigma_0, \alpha_0}^0(y_j - y_l) I(x_k, y_l) \right| \\
& + \left| \sum_{k=-W_0}^{W_0} \sum_{l=-W_n}^{W_n} \delta_{\sigma_0, \alpha_0}^{(0)}(x_i - x_k) \delta_{\sigma_n, \alpha_n}^{(n)}(y_j - y_l) I(x_k, y_l) \right| \quad (2.45)
\end{aligned}$$



Şekil 2.17: Orijinal görüntü ve DSCANED algoritması sonucu elde edilen görüntü [8]

Birçok kenar algılama yöntemi özelliklerin elde edilmesinden hemen sonra kenar çevritlerini (contour) inceltmek için, son aşamada eşik seviyeleme teknikleri kullanır [8]. Söz konusu çalışmada, kenar algılama işleminde eşik seviyeleme işlemi son aşamada uygulanır.

DSC ve Canny kenar algılayıcıların karşılaştırılması yapıldığında yeni kenar bulucunun Gauss beyaz gürültü altında daha verimli sonuçlar verdiği yazarlar tarafından iddia edilmektedir. Bu sonuca, Canny algoritmasından farklı olarak, her iki algoritmada da yer alan, filtrenin zaman-frekans yerleşimini etkileyen σ (standart sapma) parametresine ek olarak, DSC kenar algılama algoritmasında α parametresinin kullanılması neden olmaktadır. Bu parametre frekans seçiminde değişiklik elde edilebilmesini sağlamaktadır.

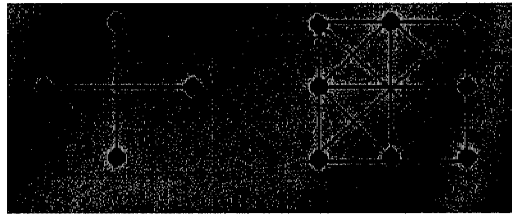
3. BAĞLILIK

Piksellerin birbirileri ile olan bağılıkları görüntünün anlaşılabilmesi açısından önemlidir. Pikseller arası bağılık kavramı, özellikle bölütleme işlemlerinde yaygın olarak kullanılmaktadır [31]. Nesneyi oluşturan piksellerin birbirileriyle olan ilişkileri, bağılık (connectivity) kavramının önemli bir uygulama alanı bulmasını sağlamaktadır. Bağılık kullanan yöntemler daha çok nesneye ait bir pikseli seçip bu piksele bağlı olan noktaları bulmaya çalışan bölgesel büyütme (region growing) yöntemleridir [31]. Bunun yanında bu çalışmadaki gibi, düşük seviyedeki görüntü işlemlerinde pikseller arasındaki bağlantıyı kullanan [11] yöntemler de bulunmaktadır.

Piksellerin bağılıkları incelenirken seçilen pikselin dört yanında yer alan 4-komşu piksele veya tüm çevresini saran 8-komşu pikseline bakılması gerekir (Şekil 3.1). Komşulukların tanımı ve kullanım potansiyeli ilk olarak 1970'lerde Rosenfeld [32] tarafından gösterilmiştir. Rosenfeld'in yaptığı bu tanımlamayla bir pikselin çevresindeki pikseller ile 4-bağılılığı veya 8-bağılılığı görüntü işlemede yaygın kullanılan bir tanım olmuştur.

3.1. Matematiksel Olarak Bağılık Tanımı

Bir pikselin yatay ve düşeyindeki tüm komşuları, 4-komşular olarak adlandırılırken, bunlara ek olarak köşegenlerde bulunan bütün pikselleri 8-komşular (ya da 8-komşuluk) olarak adlandırılır (Şekil 3.1). Bağılık kavramı görüntü işlemede kullanılacaksa, yapılacak işlem için hangi komşuluğun gerekli olduğunun belirlenmesi gerekir. Daha az işlem gücü gerektirdiğinden, genellikle 4-komşuluk tanımı kullanılmaktadır.



Şekil 3.1: 4 ve 8 komşuluklar

P ve Q görüntü üzerinde iki nokta olmak üzere bu iki noktanın bağıllığı incelendiğinde; Σ orijinal görüntüyü gösterirken, S, Σ 'nin bir alt kümesi olduğunda, P ve Q'nun birbirine bağlı olabilmesi için P ve Q'yu birleştiren yolun S'in içinde bulunan noktalardan oluşması gerekir.

P noktası kendine sıfır uzunluğunda bir yolla bağlıdır. Eğer P, Q'ya bağlı ise Q'da P'ye bağlıdır. Eğer P, Q'ya ve Q'da R'ye bağlı ise P noktası da R'ye bağlıdır. Bu ilişki S'yi, S'in birbirine bağlı bileşenleri olarak düşünülebilecek denklik sınıflarına ayırmıştır [33].

İkilik (binary) görüntülerde bağıllığı incelemek 0 ve 1'lerden oluşan bir görüntü olduğu için, çok seviyeli görüntülere göre daha kolaydır. İkilik görüntüde yer alan bir pikselin 4 ve 8 komşuluğu Şekil 3.2'de yer almaktadır.

0	1	0		1	1	1
1	1	1		1	1	1
0	1	0		1	1	1

Şekil 3.2: 4- komşuluk ve 8-komşuluk

Gri seviyeli görüntülere, ikilik düzende kullanılan bağıllık tanımının uyarlanması gereklidir. Çünkü gri seviyeli görüntülerde, piksellerin 0 veya 1 gibi sadece iki değeri bulunmamaktadır. Örneğin, 8-bitlik bir görüntüye ait piksellerin değerleri 0-255 arasında değişmektedir.

Bu görüntülerde bağıllığın incelenbilmesi için bulanık kümeler teorisinden yararlanılabileceği düşünülmüştür. Rosenfeld tarafından ilk olarak ortaya konulan bağıllık kavramı, 1984'te, bulanık kümeler teorisi kullanılarak genişletilmiştir [33]. Ortaya çıkan kavram çeşitli uygulamalarda, [31,34] kullanılmaktadır [35].

'Bulanık sistemler' veya 'bulanık mantık' olarak isimlendirilen 'bulanık kümeler teorisi' belirsizliğin temsili için 1965 yılında Lotfi A. Zadeh [36] tarafından geliştirilmiş ve daha sonra görüntü analizi ve model tanıma alanlarında kullanılmaya başlanmıştır.

3.2. Bulanık Küme

Σ , noktalar kümesi ve p , Σ 'nin elemanı olmak üzere, $p \in E$ ve $E \subset \Sigma$ olur. Bu bağıntı $\mu_E(p)$ karakteristik fonksiyonu ile,

$$\mu_E(p) = \begin{cases} 1, & p \in E \\ 0, & p \notin E \end{cases} \quad (3.1)$$

olarak gösterilebilir.

Karakteristik fonksiyon, belirli noktaların fonksiyonu olarak $[0,1]$ aralığında herhangi bir değer alabilir. $\mu_E(p)$, bulanık üyelik fonksiyonu olarak isimlendirilir. Bir noktanın üye veya üye olmama durumu $\mu_E(p)$, p 'nin E 'deki üyelik derecesini (DOM-degree of membership) gösterir. p , E 'nin içinde değil ise; $\mu_E(p) = 0$, p , E 'nin tamamen içinde ise; $\mu_E(p) = 1$, p , E 'nin tamamen içinde değil ise; $0 < \mu_E(p) < 1$ değerlerini alır.

Eğer p , E 'de kuvvetli bir DOM'a sahipse $\mu_E(p)$ 1'e yakın, eğer kümeye üye olma derecesi azsa, 0'a yakın bir değerdir. Eğer her $p \in \Sigma$ ve $\mu_E(p)$, $[0,1]$ kapalı aralığında değerlere sahipse bu durumda E bulanık bir kümedir.

Her $p \in \Sigma$ için $\mu_E(p) = 0$ ise, bulanık küme E , boş bir kümedir. Eğer E ve F gibi iki bulanık küme her $p \in \Sigma$ için, $\mu_E(p) = \mu_F(p)$ şartını sağlıyorsa eşit oldukları söylenebilir.

E 'nin tümleyeni \bar{E} ile gösterilir ve üyelik fonksiyonu, $\mu_{\bar{E}}(p) = 1 - \mu_E(p)$ olarak tanımlanır.

E ve F bulanık iki küme olmak üzere tüm $p \in \Sigma$ için bu iki kümenin birleşimi olan G 'nin üyelik fonksiyonu;

$$\mu_G(p) = \max\{\mu_E(p), \mu_F(p)\} \quad (3.2)$$

E ve F bulanık iki küme olmak üzere tüm $p \in \Sigma$ için bu iki kümenin kesişimi olan H 'nin üyelik fonksiyonu;

$$\mu_H(p) = \min\{\mu_E(p), \mu_F(p)\} \quad (3.3)$$

olarak tanımlanır.

3.3. Topoloji

Topoloji nesnelerin şekillerini, nesne üzerindeki noktalar arasındaki bağlantıları kullanarak inceleyen bir matematik dalıdır. Görüntüler üzerinde beliren nesnelerin topolojik özellikleri kenar bulma, görüntü iyileştirme, kontur takip etme gibi birçok görüntü uygulamasında kullanılır [37,38]. Çünkü, genel topoloji, sayılamaz sayıda uzay elemanı içeren kümelerden oluşur. Bu teori bir noktanın komşuluğunda sonsuz sayıda çok nokta olduğu kabullenmesinden yararlanır. Sayısal görüntülerin her zaman sınırlı sayıda eleman içermesinden dolayı sınırlı kümelerle ilgili bir topoloji tanımının yapılması gerekir. Topolojik özelliklerin, kesikli görüntülerde bulunabilmesi için sayısal topoloji kavramı oluşturulmuştur. Genelde basitliğinden dolayı ikilik görüntü göz önünde tutularak sayısal topolojiye ait tanımlamalar oluşturulmuştur. Gri seviyeli görüntülerin incelenebilmesi için, bulanık sayısal topoloji kavramı kullanılarak sayısal topoloji genelleştirilmiştir [32].

Sayısal topoloji (digital topology) tanımı, ilk olarak, 1970'lerin başında Rosenfeld [32] tarafından, görüntüde pikselleri bir düğüm ve komşu piksellerin bu düğüme kollarla bağlı olduğunu düşünülerek yapılmıştır. Bu düşünce ile, Rosenfeld, oluşturduğu grafiği 'komşuluk grafiği' olarak adlandırmış ve grafik üzerinde düğümdeki bir pikselin çevresinde dört veya sekiz komşu olabileceğini göstermiştir. Bu tanımlamayla bir pikselin çevresindeki pikseller ile 4-bağlı veya 8-bağlı olduğu görüntü işlemede çok yaygın kullanılan bir kavram olmuştur. Böylece piksellerin oluşturdukları sınırlı sayıda eleman içeren kümeler için topolojik kavramları tanımlayabilme olasılığı doğmuştur.

Sayısal görüntülerin topolojik analizleri görüntünün sadece iki değere sahip olmasından dolayı kolaydır. Konu gri seviyeli görüntüler olduğunda, görüntü içerisinde piksellerin bir çok değere sahip olması belirsizliklere yol açmaktadır. Gri seviyeli görüntülerin topolojik analizleri görüntüye bir eşik seviyeleme uygulandıktan sonra görüntünün sayısal hale getirilmesiyle yapılabilir. Yanlış eşikleme görüntüde, bağlı olmayan noktaların bağlı gözükmesine ve görüntü özelliklerinin kaybolmasına neden olacağından, gri seviyeli görüntülerin topolojik analizi için bulanık topoloji kavramı oluşturulmuştur [32]. Bulanık

topolojik ilişkiler oluşturulurken bulanık küme kuramından ve buradan elde edilen bulanık bağıllık tanımından yararlanılmıştır.

Bulanık bağıllık tanımı, görüntüdeki piksellerin ne ölçüde birbirleriyle bağlı olduğunu gösterir. Bu tanım, keskin (ikilik kümelerdeki) bağıllığın doğal genişlemesidir ve görüntülerin kesin olmayan topolojik yapılarını yansıtmaktadır. Literatürde algoritmalar, genellikle iki nokta arasındaki en iyi yolu hesaplayarak bulanık bağıllığı hesaplamaktadır [38].

Bulanık bağıllık kavramı, görüntü elemanlarının mümkün olan her piksel çifti arasında mümkün olabilecek her yol için atanan bağıllık kuvveti ile nesnedeki görüntü elemanlarının el sıkışma (hanging)-birlikteliği fikrini ortaya koymuştur. Bulanık bağıllık görüntüde yer alan öndeki nesnelere arkadaki nesnelere ayırmak için kullanılır.

3.4. Bulanık Topoloji

Σ , $I : \Sigma \rightarrow \{Z \cup 0\}$ görüntüsünden elde edilen piksellerin yerleşimleri olmak üzere, burada, Z tüm pozitif tamsayıların kümesidir ve I , belirli piksel yerleşiminden belirli bir gri değerine bir haritalamadır. Böylece $I(p)$, $p \in \Sigma$ pikselinin değeridir ve $I(\Sigma)$, yoğunluklardan oluşan görüntü matrisini gösterir. Genellikle $I(\Sigma)$ anlamında I kullanılmaktadır. Görüntülerle gri seviyelere bağlı olarak hesaplamalar yapılır. Bu hesapların yapılabilmesi için bazı tanımlamalara gerek duyulur.

Tanım 3.4.1: Görüntüdeki üyelik değeri: $I : \Sigma \rightarrow \{Z^+ \cup 0\}$ görüntü olsun. En az bir $p \in \Sigma$ için $I(p) \neq 0$ olduğunu kabul edelim. $\kappa = \max\{I(p) | p \in \Sigma\}$ olsun. I da p 'nin üyeliğin derecesi (DOM),

$$\mu_I(p) = \frac{I(p)}{\kappa} \quad (3.4)$$

olur. Bir piksel için DOM, $\mu_I(p)$, normalize edilmiş yoğunluk değeridir. $\mu_I(p)$, “aydınlık” piksellerin kümesinde p 'nin DOM'u olarak düşünülebilir. Benzer şekilde “karanlık” piksellerin kümesinde p 'nin üyelik derecesi,

$$\mu_{\bar{I}}(p) = 1 - \mu_I(p) \quad (3.5)$$

olarak tanımlanır.

Görüntüdeki özel komşuluklar için $E+p$ tanımı kullanılır. Yani $E \subset Z^2$, özel geometrik biçimlerde yakın piksel yerleşimlerinin kümesidir. $E + p \subset \Sigma$, $p \in \Sigma$ pikselinde yer alan E şeklinin komşuluğudur.

Görüntü içindeki, p pikselinin üyelik derecesi (DOM), $\mu_I(p) = \mu_{I(\Sigma)}(p)$ olarak veya komşuluk içinde yer alanlar $\mu_{I(E+p)}(p)$ olarak gösterilir. Bir pikselin DOM'u pikselin 0 ve 1 arasına ölçeklenmiş yoğunluğu olduğundan, tüm $E+p$ için, $\mu_{I(E+p)}(p) = \mu_I(p)$ olur. p pikselinin DOM'u her yerde $\mu_I(p)$ olarak yer alabilir. Genellikle p pikselinin gri seviyesi veya yoğunluk değeri $I(p)$ 'ye karşılık gelen DOM olarak kullanılır.

Tanım 3.4.2: E 'de p ve q pikselleri arasında $\rho = \{p = p_0, p_1, \dots, p_n = q\} \subset E \subset \Sigma$ herhangi bir yol olsun. Bu yolun şiddeti, $s_E(\rho)$, bu yolun üzerindeki pikseller arasındaki en küçük DOM değerine eşittir.

$$s_E(\rho) = \min_{0 \leq i \leq n} \{\mu_I(p_i)\} \quad (3.6)$$

Başka bir deyişle bir yol (zincir) en zayıf bağlantısı kadar güçlüdür. Bir görüntüde, iki piksel arasında sonlu sayıda yol vardır. İki piksel arasındaki bağlantı derecesini gösterebilmek için, yukarıdaki tanıma göre, pikselleri birbirine bağlayan mümkün olabilecek tüm yolların incelenmesi gereklidir.

Tanım 3.4.3: p ve q piksellerinin bağlantı derecesi, p ve q arasında $\rho \subset E$ olan tüm yollar için,

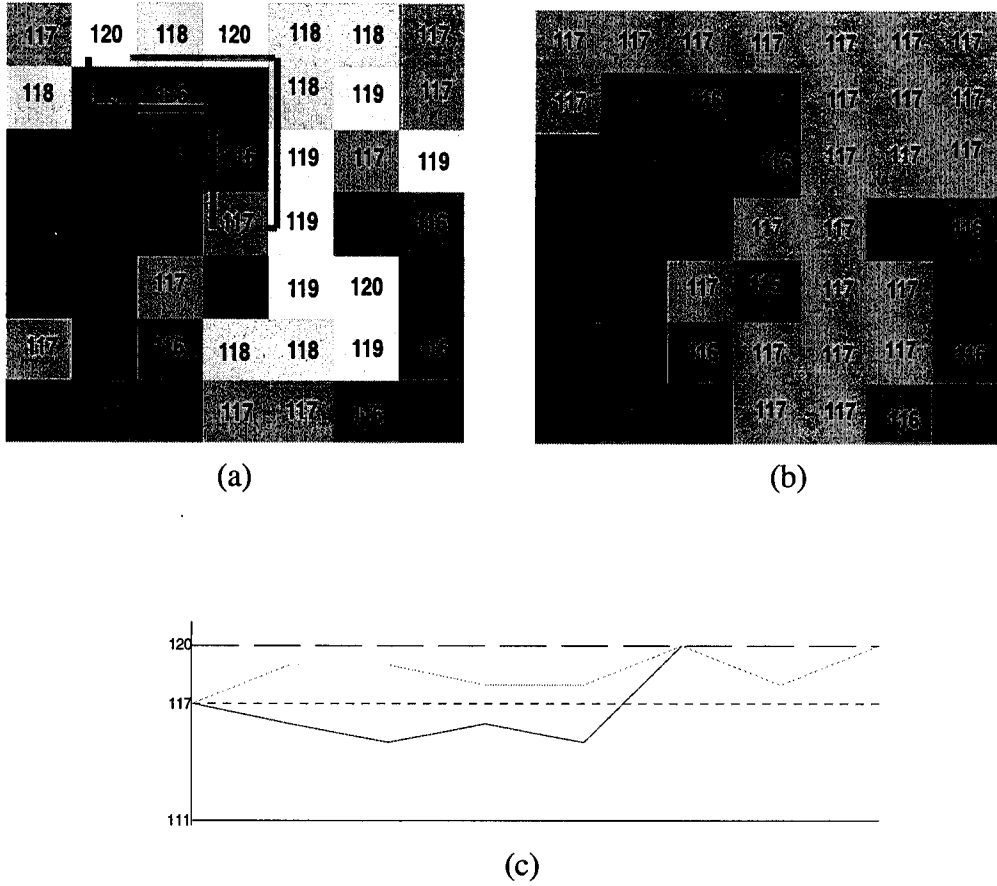
$$c_E(p, q) = \max_{\rho \subset E} \{s_E(\rho)\} \quad (3.7)$$

olarak tanımlanır. Bu bilgiye bağlantı derecesi (Degree of Connectedness) adı verilir ve DOC olarak kısaltılabilir. Aslında iki piksel arasındaki bağlantı derecesi, iki piksel arasındaki en kuvvetli yolun şiddetidir.

Tanım 3.4.4: $E \subset \Sigma$ şartını sağlayan herhangi bir küme için, E 'nin bağlantı derecesi;

$$c_E(E) \equiv \min_{p, q \in E} \{c_E(p, q)\} \quad (3.8)$$

olarak tanımlanır. Pikseller kümesinin bağlantı derecesi, kümenin iki üyesi arasındaki en küçük DOC olarak tanımlanır.



Şekil 3.3: (a) 7×7 'lik orijinal görüntü kesiti, (b) 7×7 'lik bağlantı haritası oluşturulmuş görüntü kesiti, (c) Bağlantı haritası oluşturulan kesitte bağlantı yolları

Tanım 3.4.5: E 'de bulunan p ve q pikselleri eğer ve sadece eğer, $0 \leq i \leq n$ için $p_i \in E$ olduğunda, $\rho^c = \{p = p_0, p_1, \dots, p_n = q\}$ ile tanımlı bir yol içeriyorsa ve $p_i \in \rho^c$ şartını sağlayan tüm pikseller için,

$$\mu_I(p_i) \geq \min\{\mu_I(p), \mu_I(q)\} \quad (3.9)$$

şartı sağlanıyorsa, bu iki piksel birbirine bağlıdır denebilir. $E \subset \Sigma$ 'de iki uç noktanın sahip olduğu DOM'dan daha küçük olmayan DOM'a sahip pikseller içeren bir yol var ise, E 'de ki bu iki piksel bağlıdır. Matematiksel olarak ifade etmek gerekirse,

$$c_E(p, q) = \min\{\mu_I(p), \mu_I(q)\} \quad (3.10)$$

eşitliği sağlanıyorsa p ve q bağlıdır. Bir başka deyişle, eğer iki pikseli bağlayan

en güçlü yol boyunca en küçük DOM eğer piksellerden birinin değerine ulaşıyorsa iki pikselin birbirine bağlı olduğu söylenebilir.

Yukarıda yer alan tanımlar kısaca aşağıdaki şekilde özetlenebilir:

- 1) $[0,1]$ kapalı aralığında p pikselinin DOM'u, $\mu_I(p)$, p 'den q 'ya $\rho(p,q)$ yolunun kuvveti, $s_E(\rho)$, p ve q piksellerinin DOC'u $c_E(p,q)$ olarak ifade edilir.
- 2) DOC yansımaya sahiptir, $c_E(p,p) = \mu_I(p)$.
- 3) DOC simetriktir, $c_E(p,q) = c_E(q,p)$.
- 4) İki pikselin DOC'u, ikisinin DOM'unun küçük olanından daha büyük değildir, $c_E(p,q) \leq \min\{\mu_I(p), \mu_I(q)\}$.

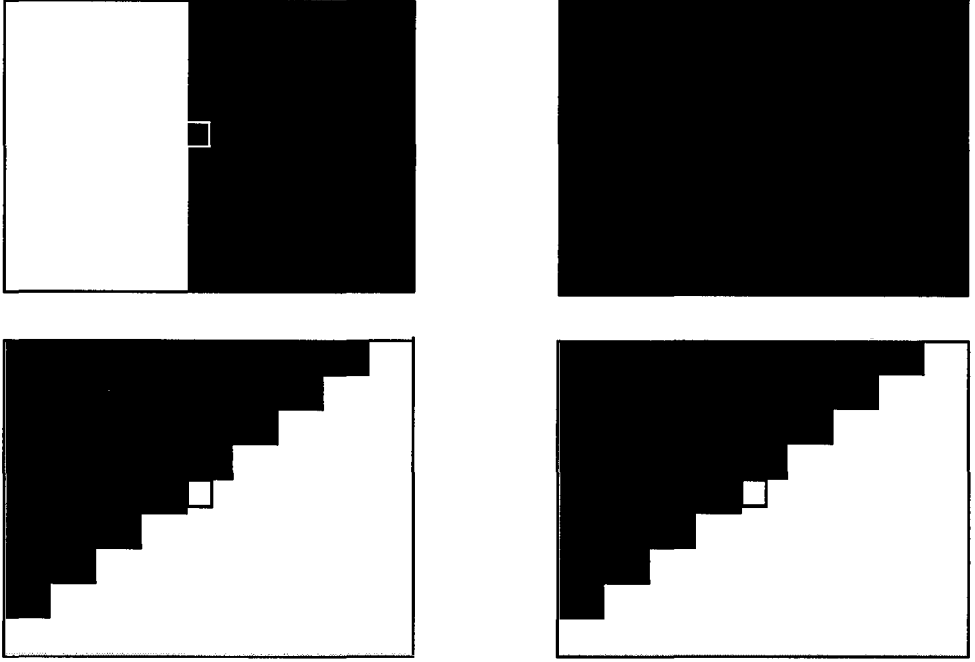
3.5. Bağlantı Haritası

P 'nin $2m+1$ satır ve $2m+1$ kolondan oluşan, bir maskenin ortasında yer alan piksel olduğunu kabul edersek, maskeye ait bağlantı haritası, P 'nin maske içinde yer alan diğer bütün piksellerle bağlantı ilişkilerini gösterir. Buna bağlantı haritası ismi verilir. Bağlantı haritalarının nasıl oluşturulabileceği Şenel ve diğerleri [11] tarafından açıklanmıştır. İlgili çalışmada, P 'den Q 'ya gidilebilecek tüm yollar üzerindeki piksel değerleri için, P noktası en küçük değere sahipse P noktasının Q noktasına olan bağlılık değeri α -bağlılığı olarak isimlendirilmiştir. σ , P noktasının komşuluğundaki pikseller, $I(P)$, P noktasının görüntü içerisindeki değeri ve $I(\sigma)$, P noktasının komşuluğundaki görüntü, $M(P)$ ve $M(\sigma)$ bu görüntülere ait bağlantı haritası değerleri olacak şekilde atanmıştır. σ_A , σ 'nun içinde P noktasına α -bağlı piksellerin oluşturduğu alt küme, σ_B , σ 'nun içinde P noktasına α -bağlı olmayan piksellerin oluşturduğu alt küme ve σ_C , σ 'nun içinde P noktasına bağlı olmayan piksellerin oluşturduğu alt küme olarak tanımlanır. Bu alt kümeler için bağlantı haritası;

$$M(P_i) = P_i \in \sigma_A \text{ için } I(P_i) \text{'ye eşit,}$$

$$M(Q_j) = Q_j \in \sigma_B \text{ için } I(P) \text{'ye eşit,}$$

$$M(R_k) = R_k \in \sigma_C \text{ için } c_\sigma(R_k, P) \text{'ye eşittir.}$$



Şekil 3.4: Birinci sütun orijinal görüntüler, ikinci sütun da bağlantı haritası oluşturulmuş görüntüler yer almaktadır.

4. BAĞLANTI HARİTASININ KENAR ALGILAMADA KULLANILMASI

Bu bölümde, basit [17-20] ve bileşik [3-8] yeni kenar algılama yöntemlerinden farklı olarak bu çalışmada oluşturulan kenar algılama yöntemleri sunulmaktadır. Önerilen yeni kenar bulma algoritmasının temeli, bağlantı haritasının oluşturulmasına dayanır.

4.1. Kenar Algılama Algoritması – 1

Bağlantı haritası oluşturulduktan sonra belirlenen eşik seviyeleri için kenarların algılanmış olduğu sonuç görüntüsü elde edilir. Elde edilen bu yeni kenar algılama yöntemiyle görüntüye kenar algılama sürecine başlamadan önce bir filtreleme yapmak gerekmemektedir. Oysa ki birçok yöntem kenar algılama sürecine başlamadan görüntüde düzgünleştirme ve iyileştirmeyi sağlamak için filtrelemelere gerek duyar [21-23]

Şekil 4.1’de verilen akış diyagramına paralel olarak yeni yonteme ait algoritma (8-bit’lik) gri seviyeli görüntüler için, aşağıdaki şekilde açıklanabilir.

- 1) orijinal görüntü (8-bit) alınır.
- 2) 0-255 arasındaki görüntü yoğunluk değerleri C matrisine atanır.
- 3) C matrisinin tersi T matrisine atanır.
- 4) T matrisine ait (7×7 , 9×9 veya 11×11 maskeler kullanılarak) bağlantı haritası oluşturulup bu değerler I matrisinde saklanır.
- 5) Görüntünün tüm elemanlarının yoğunluğu 255 olan Z matrisi oluşturulur.
- 6) I matrisindeki ve T matrisindeki piksellerin yoğunluk değerleri eşik seviyesinden küçükse, yeni oluşturulan matris Z’de, o piksele ait yoğunluk değeri olarak sıfır atanır. Aksi durumda Z matrisindeki o piksele ait yoğunluk değeri saklanır. Bu işlem matriste yer alan tüm pikseller için yapıldığında yeni Z matrisi elde edilmiş olur.
- 7) Z matrisinde yoğunluk değeri sıfır olmayan her pikselin dördü komşuluğundaki piksel yoğunluk değerleri kontrol edilip tek bir nokta mı yoksa kenara ait bir nokta mı olduğu kontrol edilir.

- 8) Kenara ait piksel değerlerine 255 değeri atanıp görüntüye ait kenarlar oluşturulur.

Türevden yararlanarak geliştirilen kenar algılama işlemleri gürültü içeren yüksek frekans bileşenlerini kuvvetlendirirler. Bu durumu ortadan kaldırmak için, bir ön aşama olarak alçak geçiren filtreler yaygın olarak düzgünleştirme için kullanılır. Düzgünleştirmenin miktarı, düzgünleştirme işleminin boyutuna veya ölçeğine göre değişir. Genelde, küçük ölçekli düzgünleştirme işlemleri kullanıldığında algılayıcı gürültüye karşı hassaslaşsa da yoğunluk değişimlerinin küçük ayrıntılarını ortaya çıkarır. Büyük ölçekli işlemler kullanıldığında yoğunluk değişimlerinin kaba detayları elde edilir. Bazı durumlarda algılanan kenarlar yanlış yerleştirilmiş olabilir. Bir görüntüdeki tüm kenarlar için optimal olan tek ölçekli bir düzgünleştirme işlemi seçmek oldukça zordur. Tek bir filtre boyutu iyi bir yerleştirme sağlarken aynı zamanda gürültü etkilerini yeteri kadar ortadan kaldıramayabilir [21].

Canny ve Log gibi gauss filtre kullanan kenar algılayıcılar, standart sapma σ gibi ekstra bir parametrenin kontrolünü gerektirdikleri için sorunludur. σ 'nun küçük değerleri ince detayların bulunmasını sağlarken gürültü hassasiyeti artmaktadır. σ 'nun büyük değerleri için ise kaba detaylar elde edilmesine karşın gürültü hassasiyeti azalır.

Alternatif olarak çok ölçekli kenar algılama yöntemleri önerilmektedir [10,39]. Çok ölçekli algılama yöntemleri farklı ölçeklerde düzgünleştirme işlemleri kullanarak her ölçekte farklı kenarları elde etmeyi ve daha sonra bu kenarları bir görüntüde birleştirerek görüntüye ait gerçek kenar gösterimini elde etmeye çalışırlar. Buradaki temel sorun, kaç tane filtrenin hangi ölçeklerle kullanılacağı ve filtre tepkilerinin tek bir sonuç görüntüsü oluşturacak şekilde nasıl birleştirilmesi gerektiğinin bilinmemesidir.

Bağlantı haritası kullanılarak oluşturulan yeni yöntemle ise değişik kenarlar için farklı ölçekte işlemler kullanmak gerekmemektedir. Temelde yeni yöntemle ait bulunmuş maskelerin olmamasının nedeni de budur. Bu yöntemde maske orijinal görüntünün kendisidir ve maskeye ait bulunan bağlantı haritası ile de görüntüye ait yoğunluk değerlerinden farklı olarak değişik ağırlık katsayıları kullanılmamaktadır. Bir bakıma, bağlantı haritasında orijinal görüntü ayrıca bir

filtreleme yapılmadan düzgünleştirilmektedir. Türevsel işlemler kullanan kenar algılama işlemlerinde ise boyutun ve katsayıların sabit olması ve görüntüye göre ayarlanamaması bir sorundur.

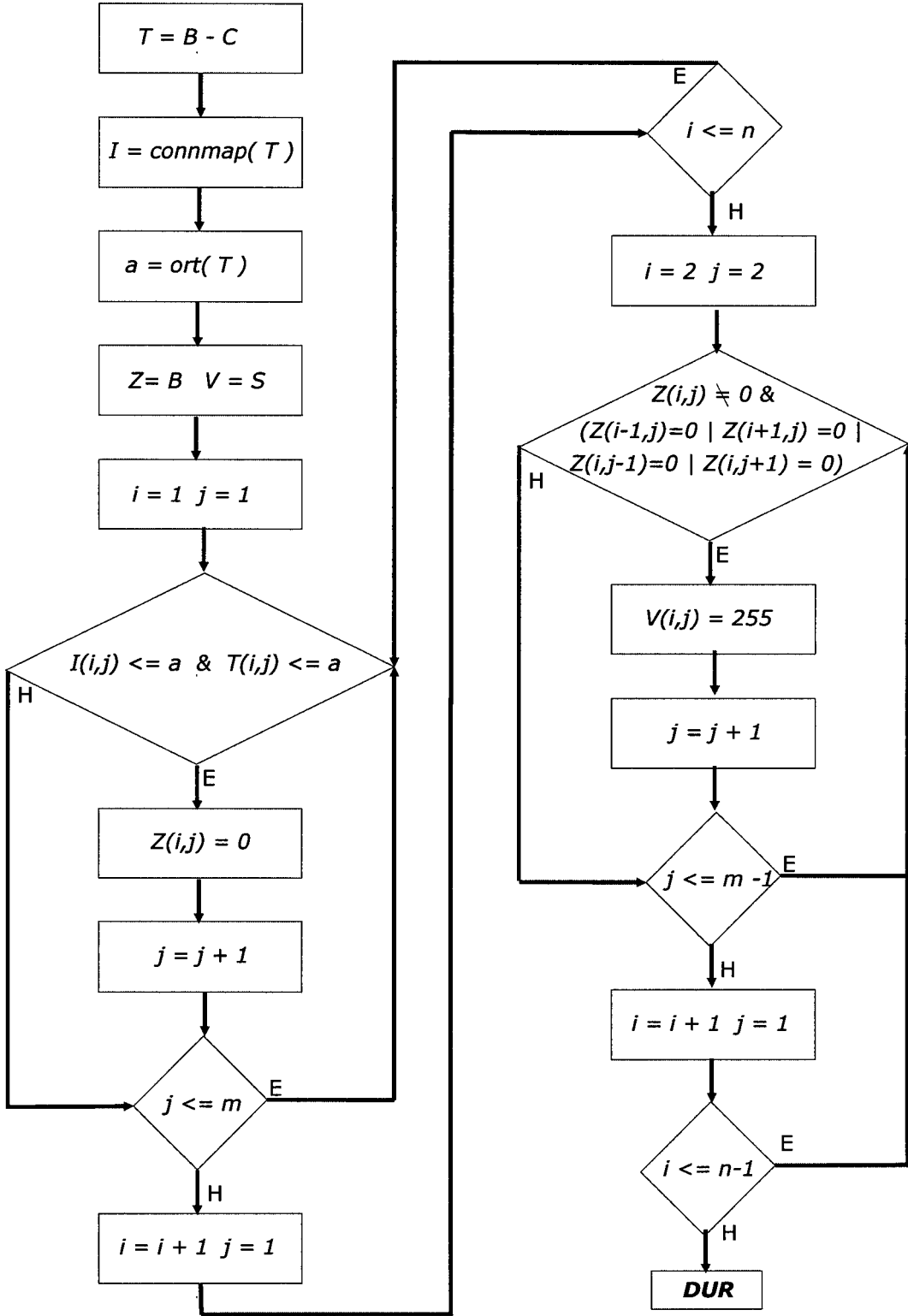
Bağlantı haritası kullanılarak geliştirilen bu yöntem farklı boyutta 60 görüntü üzerinde test edilmiştir. Şekil 4.3'de diğer yöntemlerin sonuçları ile beraber yeni yönteme ait sonuçlar verilmektedir. Yeni algoritma diğer algoritmalara göre daha uzun sürede sonuca ulaşmasına rağmen elde edilen sonuçlar tatmin edicidir. Matlab paket programı kullanılarak yazılan programlar çalıştırıldığında 256×256 boyutunda gri seviyeli bir görüntü için, 5×5 'lik Sobel maskeleri uygulanıp sonucun elde edilmesi 2,2 sn sürmekte iken bağlantı haritası yöntemi ile 7×7 'lik maske kullanılarak bağlantı haritası oluşturulmuş görüntülerde sonucu elde etme süresi 49,3 sn., 9×9 'luk maske kullanılarak bağlantı haritası oluşturulmuş görüntülerde bu süre 69,8 sn olarak elde edilir. Verilen bu süreler 1.6 Ghz., 128 Mb Ram olan bir kişisel bilgisayar kullanılarak elde edilmiştir. Kullanılan bilgisayar hızları ve hafızaları gelişen teknoloji ile artmaya devam ettiğinden bu da zamanla bir dezavantaj olmaktan çıkacaktır.

Yeni yöntemdeki sorun parlak yüzeylerle karşılaşıldığında yaşanmaktadır. Bunun nedeni bağlantı tanımları incelendiğinde anlaşılabilir. Bölüm 3'te verilen tanımlara göre bir yolun şiddeti $s_\sigma(\rho)$, o yol üzerindeki en küçük değerli üyelik değerine sahip noktanın üyelik değerine eşittir. Ayrıca iki noktanın birbirine bağlı olabilmesi için tanımlanan yol üzerindeki tüm piksellerin,

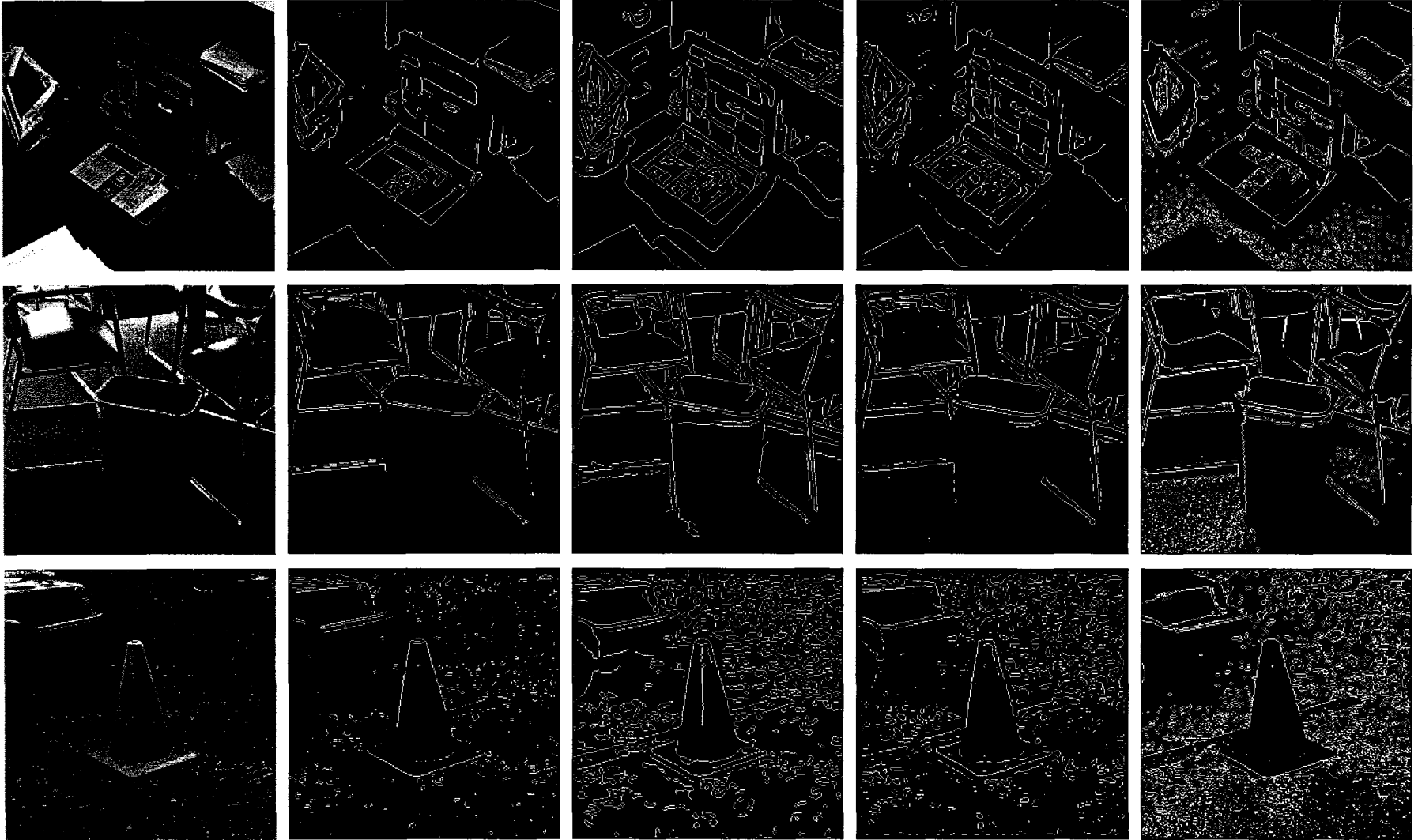
$$s_\sigma \geq \{\sigma(P), \sigma(Q)\} \quad (4.1)$$

şartını sağlaması gerekir. Bu durumda parlak yüzeylerin en yüksek değerli üyelik derecelerine sahip olduğu düşünülürse bağlantı haritası oluşturulurken daha düşük üyelik derecelerine çevrileceği bilinir. Böylece bu yüzeylerin belirginlikleri azalır ve kenar olup olmadığı belirlenirken fon olarak algılanırlar. Görüntünün tersi alınarak bağlantı haritası bulunmasından dolayı orijinal görüntüde yer alan parlak yüzeylerde problem ortaya çıkmaktadır.

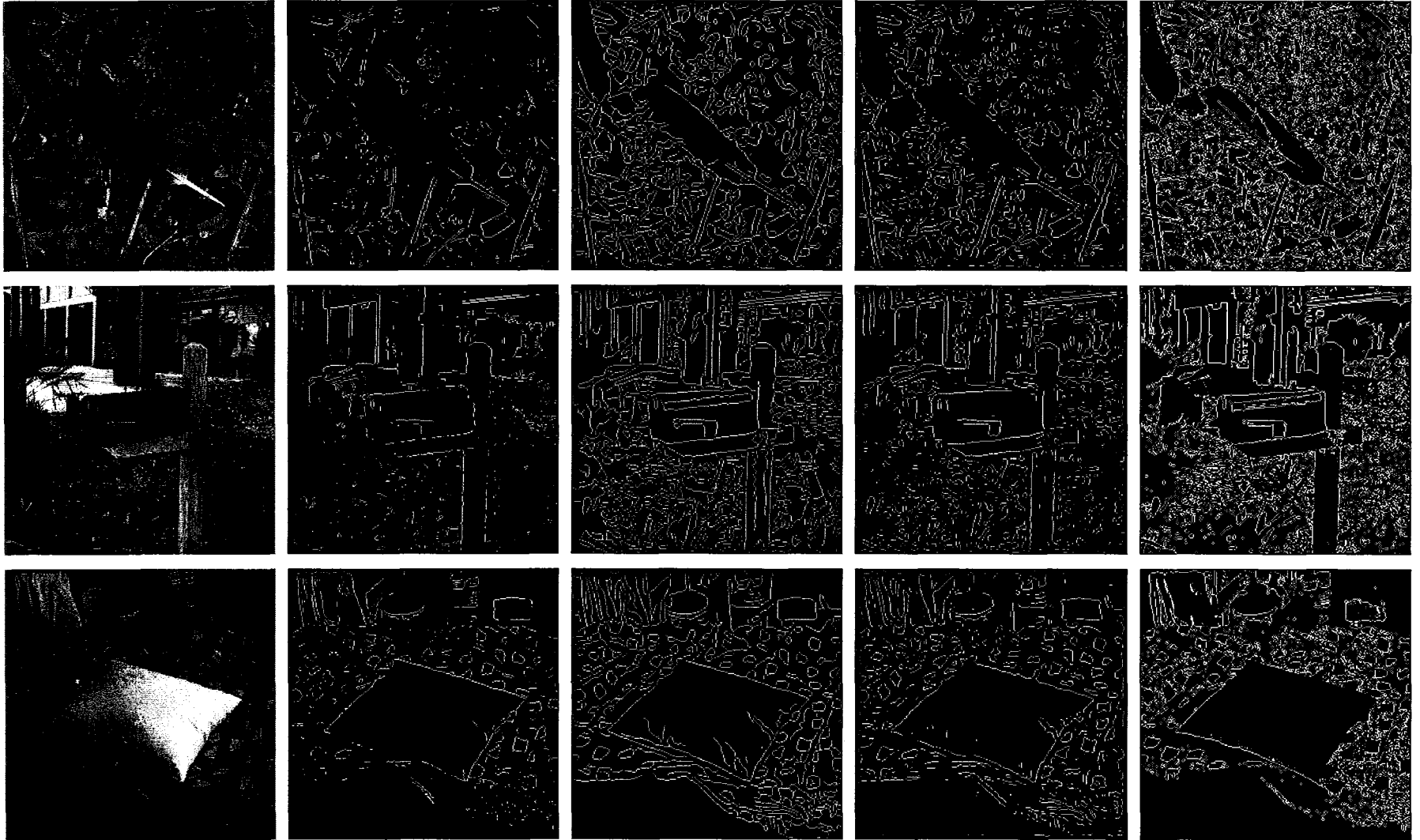
Bu algoritma, sadece gerçek görüntüler üzerinde uygulanmış ve nitelik olarak çıktıları incelenmiştir. Bu tez aşamasında, sonraki bölümde sunulan algoritmanın öncüsü olarak değerlendirilmelidir.



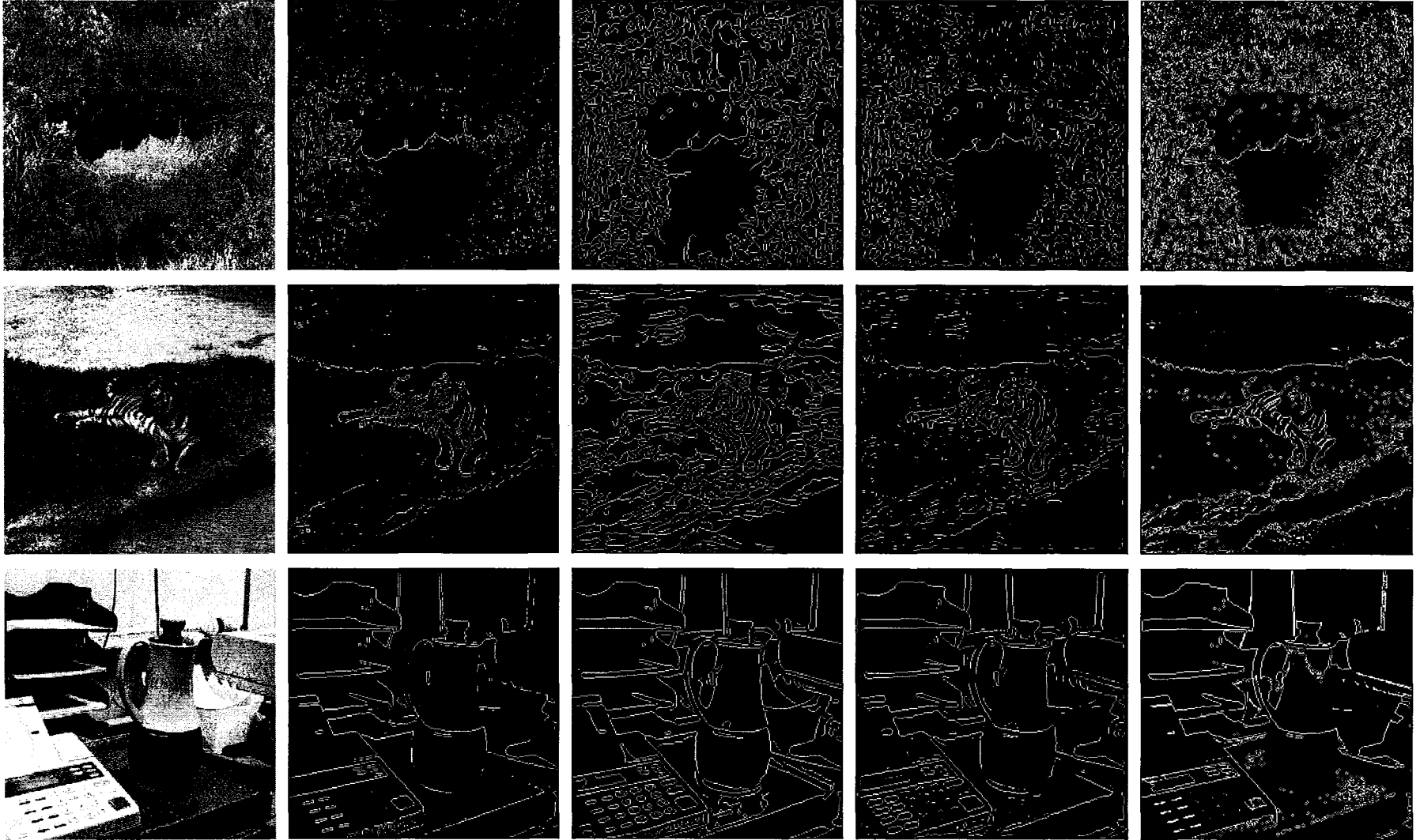
Şekil 4.1: Bağlantı haritası kullanılarak geliştirilen yönteme ait akış diyagramı.



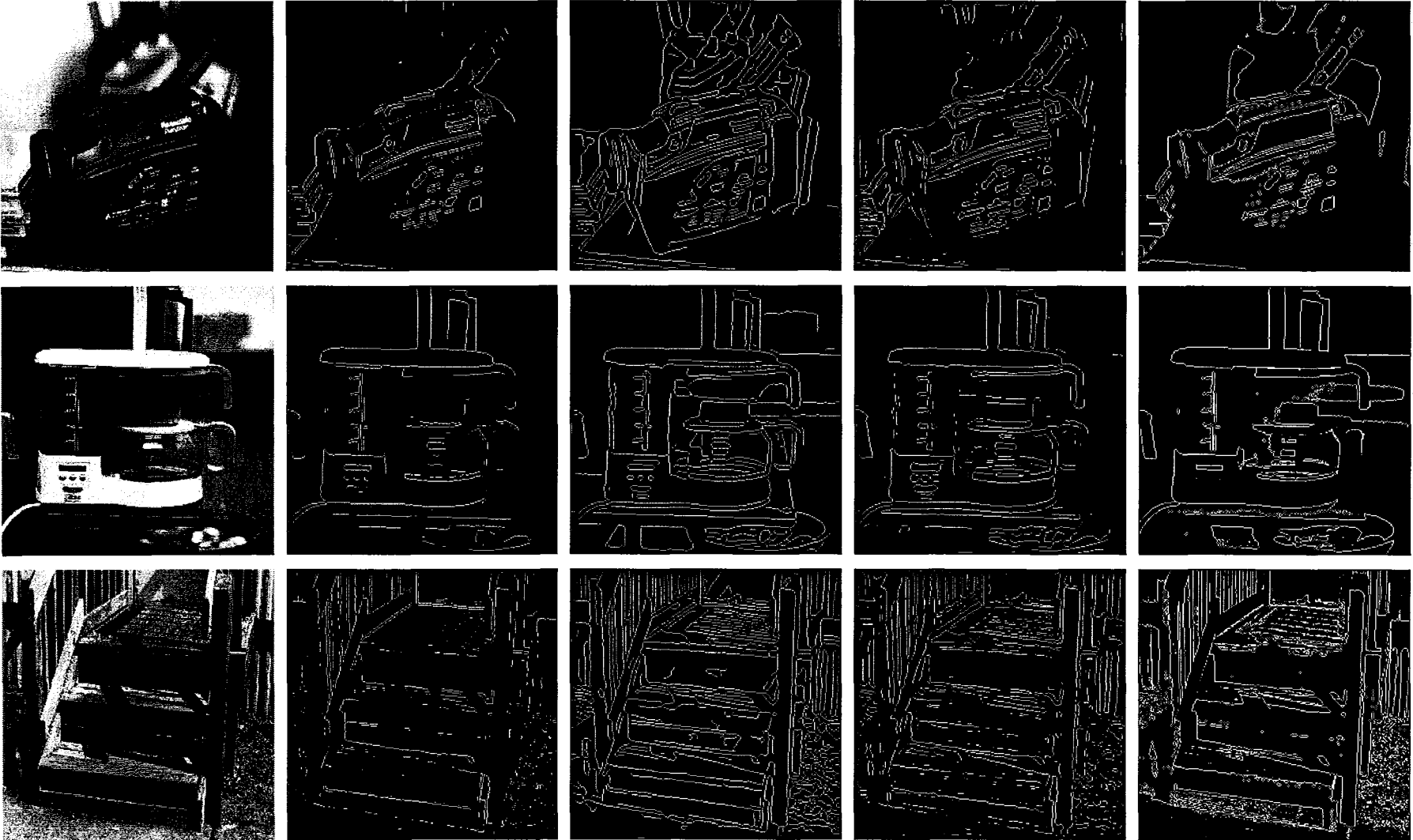
Şekil 4.2: Birinci kolon orijinal görüntüleri, ikinci kolon Sobel algılayıcı sonuçlarını, üçüncü kolon Canny algılayıcı sonuçlarını, dördüncü kolon Log algılayıcı sonuçlarını, beşinci kolon yeni algılayıcı sonuçlarını göstermektedir.



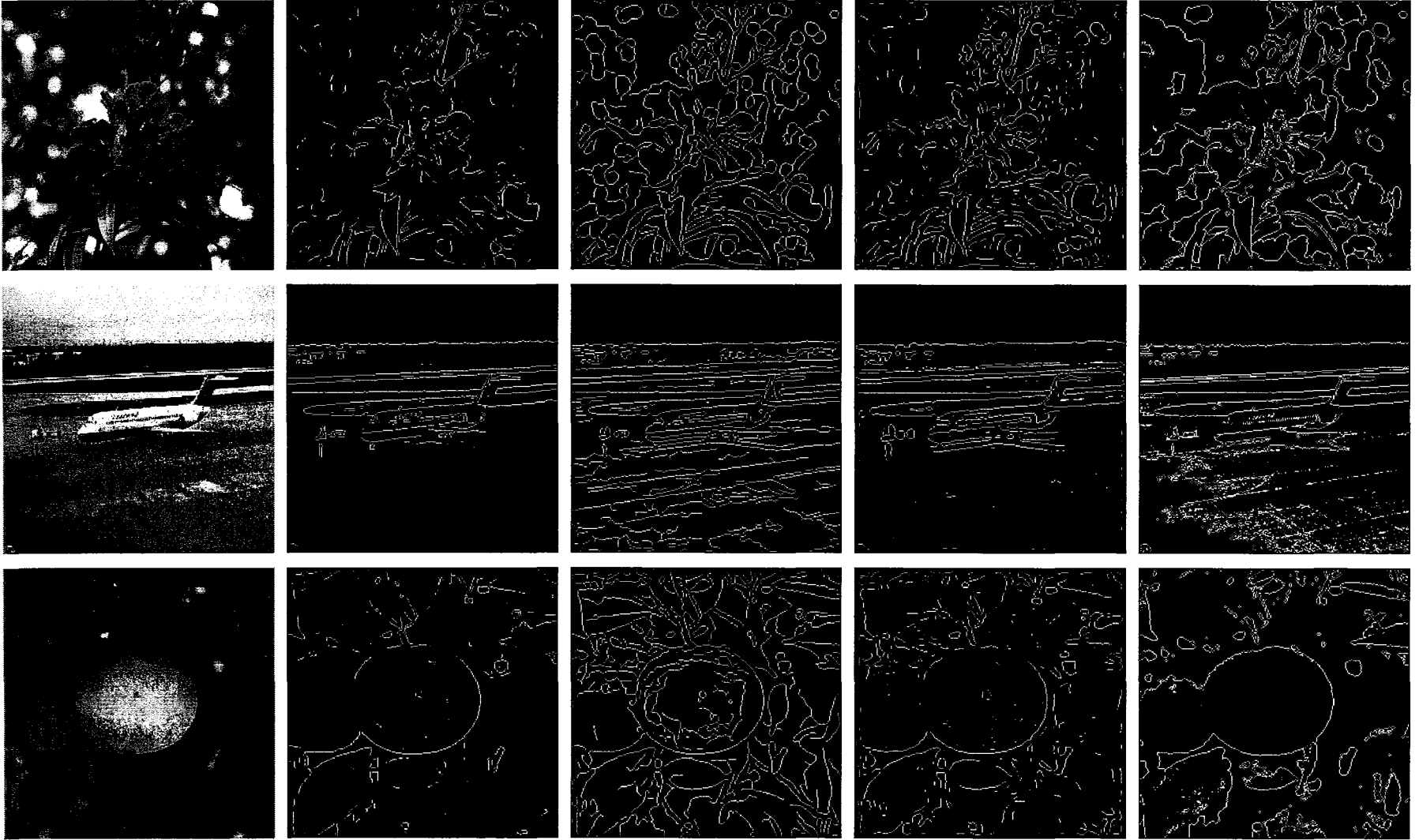
Şekil 4.2'ün devamı: Birinci kolon orijinal görüntüleri, ikinci kolon Sobel algılayıcı sonuçlarını, üçüncü kolon Canny algılayıcı sonuçlarını, dördüncü kolon Log algılayıcı sonuçlarını, beşinci kolon yeni algılayıcı sonuçlarını göstermektedir



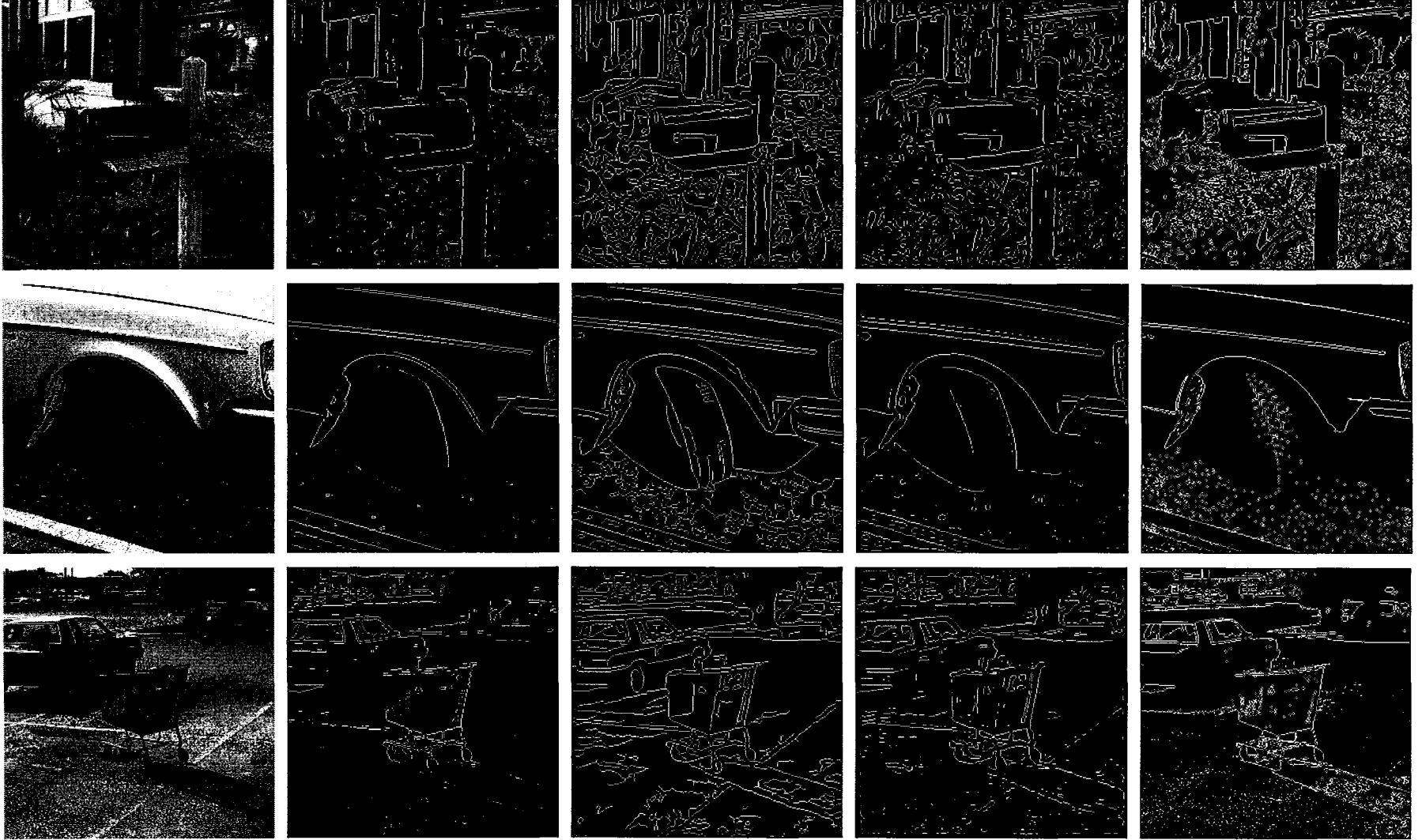
Şekil 4.2'ün devamı: Birinci kolon orijinal görüntüleri, ikinci kolon Sobel algılayıcı sonuçlarını, üçüncü kolon Canny algılayıcı sonuçlarını, dördüncü kolon Log algılayıcı sonuçlarını, beşinci kolon yeni algılayıcı sonuçlarını göstermektedir



Şekil 4.2'ün devamı: Birinci kolon orijinal görüntüleri, ikinci kolon Sobel algılayıcı sonuçlarını, üçüncü kolon Canny algılayıcı sonuçlarını, dördüncü kolon Log algılayıcı sonuçlarını, beşinci kolon yeni algılayıcı sonuçlarını göstermektedir.



Şekil 4.2'ün devamı: Birinci kolon orijinal görüntüleri, ikinci kolon Sobel algılayıcı sonuçlarını, üçüncü kolon Canny algılayıcı sonuçlarını, dördüncü kolon Log algılayıcı sonuçlarını, beşinci kolon yeni algılayıcı sonuçlarını göstermektedir.



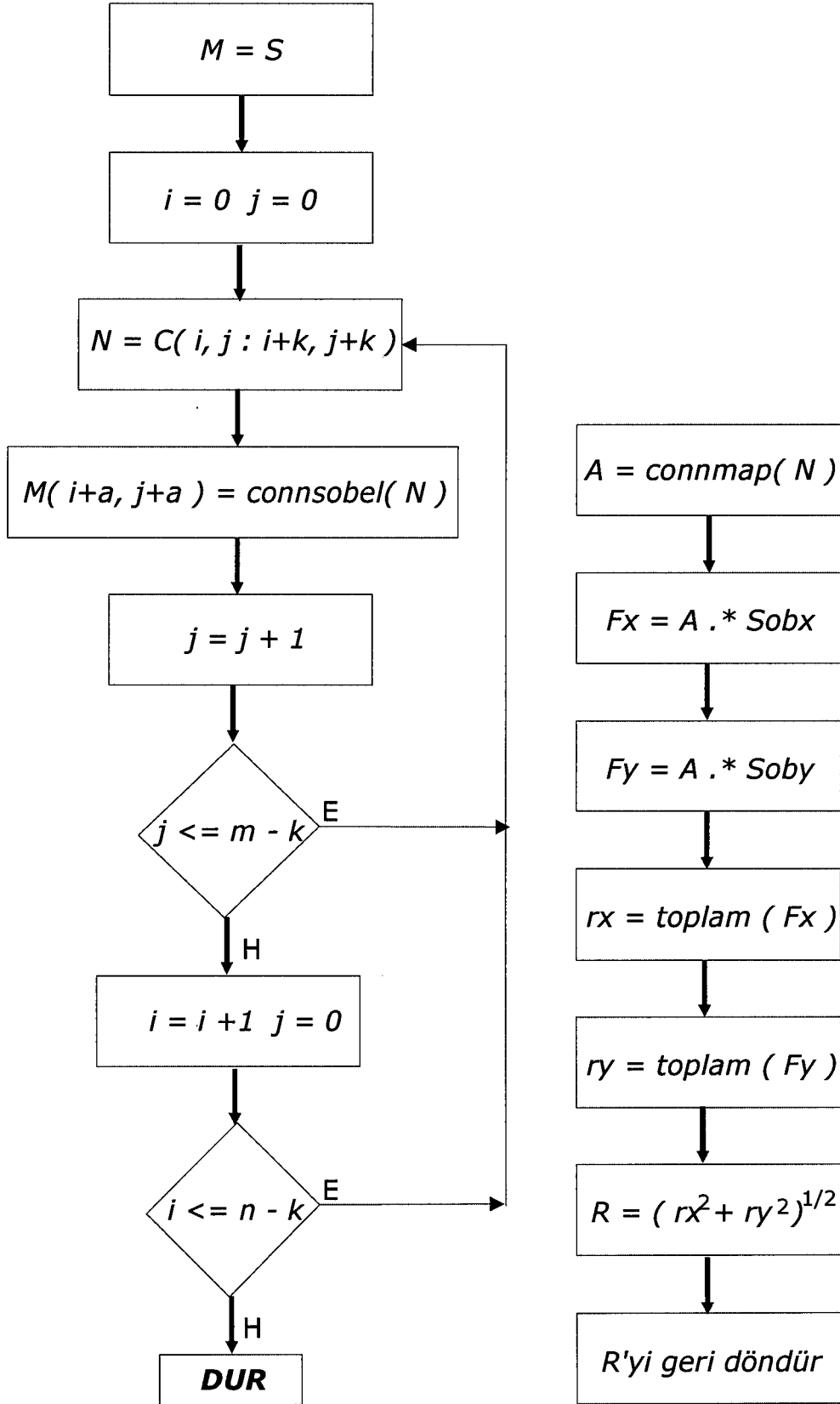
Şekil 4.2'ün devamı: Birinci kolon orijinal görüntüleri, ikinci kolon Sobel algılayıcı sonuçlarını, üçüncü kolon Canny algılayıcı sonuçlarını, dördüncü kolon Log algılayıcı sonuçlarını, beşinci kolon yeni algılayıcı sonuçlarını göstermektedir.

4.2. Kenar Algılama Algoritması - 2

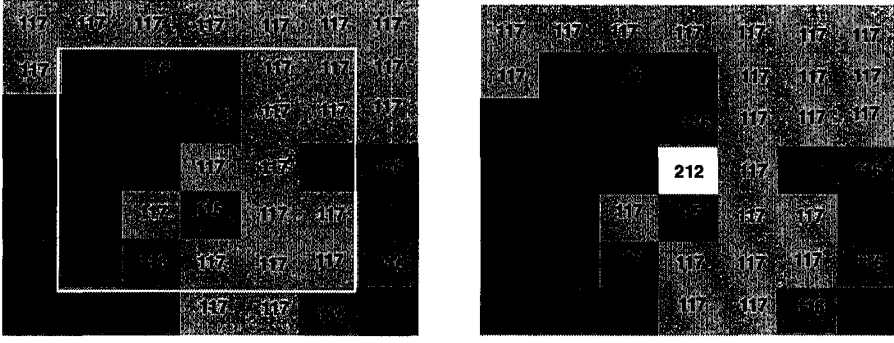
Bu bölümde önerilen yeni kenar algılama yöntemi, Sobel [21] tarafından ortaya konan yatay ve düşey yöndeki maskelerin evrişimine dayanan kenar algılama yönteminin bağlantı haritasıyla birleşiminden elde edilmiştir. Şekil 4.3'te yer alan her iki akış diyagramından yararlanarak yöntem aşağıdaki şekilde açıklanabilir:

- 1) S , tüm elemanları sıfır olan matris, C , $N \times N$ boyutunda orijinal görüntü matrisi olarak atanmıştır.
- 2) N , ($k \times k$, $k=5,7,9,11$) bağlantı haritası oluşturmak için kullanılan maske boyutunda görüntü parçasını gösteren matristir.
- 3) Maskeye ait bağlantı matrisi oluşturulduktan sonra, Sobel'e ait yatay yöndeki maskenin (3×3 ve 5×5 boyutlarında) elemanlarının bağlantı haritası oluşturulmuş matrisle merkezleri çakışacak şekilde yerleştirilerek eleman eleman çarpma yapılması ile elde edilen yoğunluk değerleri toplanıp r_x 'e atanır.
- 4) Maskeye ait bağlantı matrisi oluşturulduktan sonra, Sobel'e ait düşey yöndeki maskenin (3×3 ve 5×5 boyutlarında) elemanlarının bağlantı haritası oluşturulmuş matrisle merkezleri çakışacak şekilde yerleştirilerek eleman eleman çarpma yapılması ile elde edilen yoğunluk değerleri toplanıp r_y 'ye atanır.
- 5) Elde edilen r_x ve r_y büyüklüklerinin karelerinin toplamının karekökü alınarak, bulunan büyüklük maskenin merkez pikselinin yoğunluk değeri olarak orijinal görüntüdeki yerine atanır.

Geliştirilen bağlantı haritasına dayalı yeni kenar algılama yönteminin sonuçlarının iyi olması nedeniyle bağlantı haritası yönteminin farklı bir yaklaşımda uygulanabileceği fikri üzerine farklı bir yöntem üzerinde çalışılmıştır. Geliştirilen yöntemde, 7×7 ve 9×9 maskeler üzerinde elde edilen bağlantı haritaları üzerine 5×5 'lik Sobel işleci uygulanmıştır (Şekil 4.4). Bağlantı haritasının yumuşatma ve maske içine denk gelen, ilgisiz nesne parçalarını süzmesi ve sadece orta piksele bağlı olan nesneyi ortaya çıkarması özellikleri kullanılmaktadır. Sobel işlecinin



Şekil 4.3: Yönteme ait akış diyagramı.



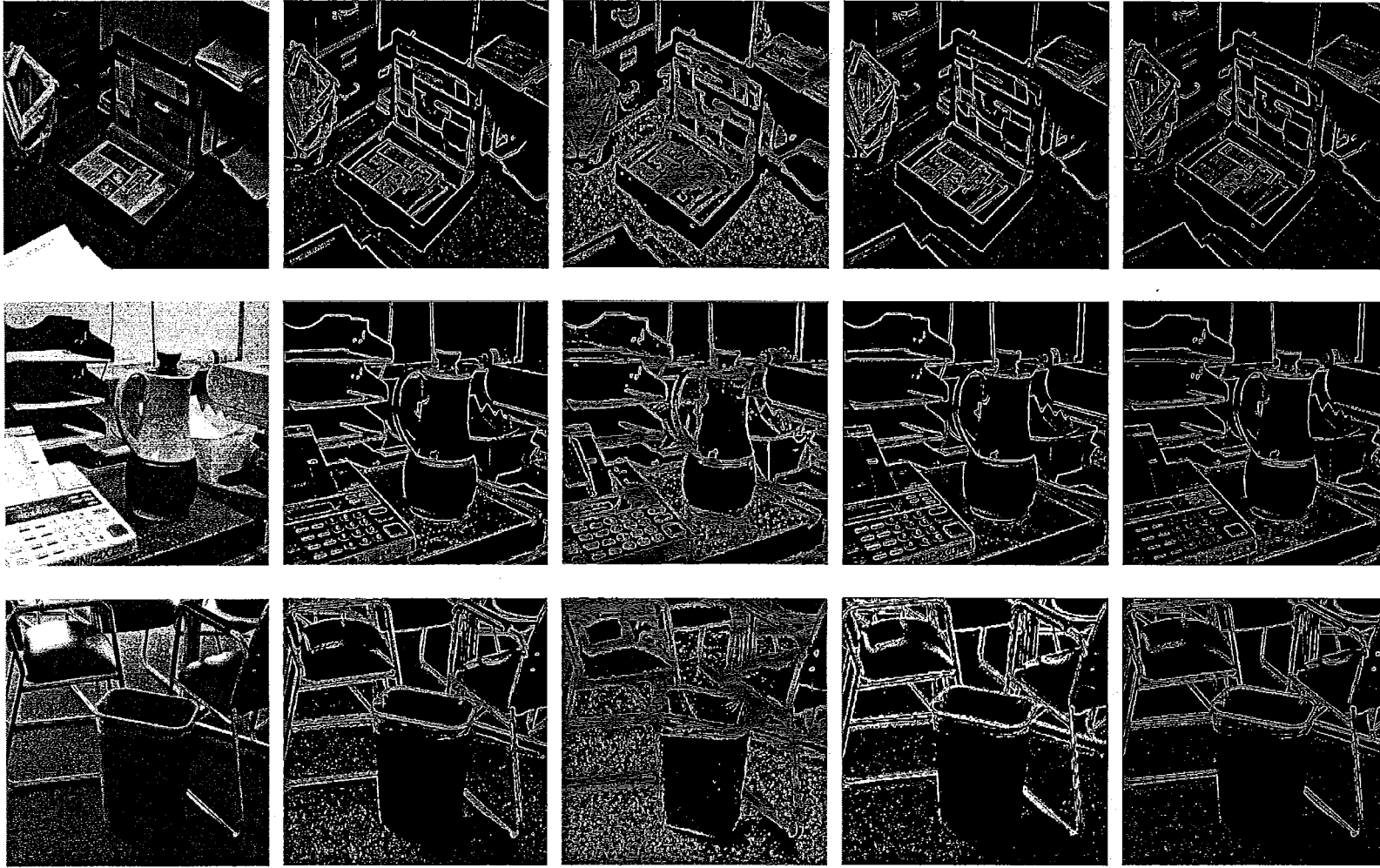
Şekil 4.4: 7×7'lik bağlantı haritası oluşturulan görüntü maskesi üzerine 5×5'lik Sobel işlecin yerleşimi

boyutu büyüdükçe daha fazla sayıda kenarı maske içine aldığından, kenar bulma sonucunda daha kalın kenarlar ortaya çıkmaktadır. Bağlantı haritası ön aşama olarak kullanıldığında, ilgisiz gradyan vektörleri nedeniyle ortaya çıkan kenarlar süzölmektedir. Bu şekilde ortaya çıkan yöntemin, yumuşatma işlemine ihtiyaç duymadığı ve büyük Sobel maskelerinin sağladığı avantajları bünyesinde bulundurduğu söylenebilir.

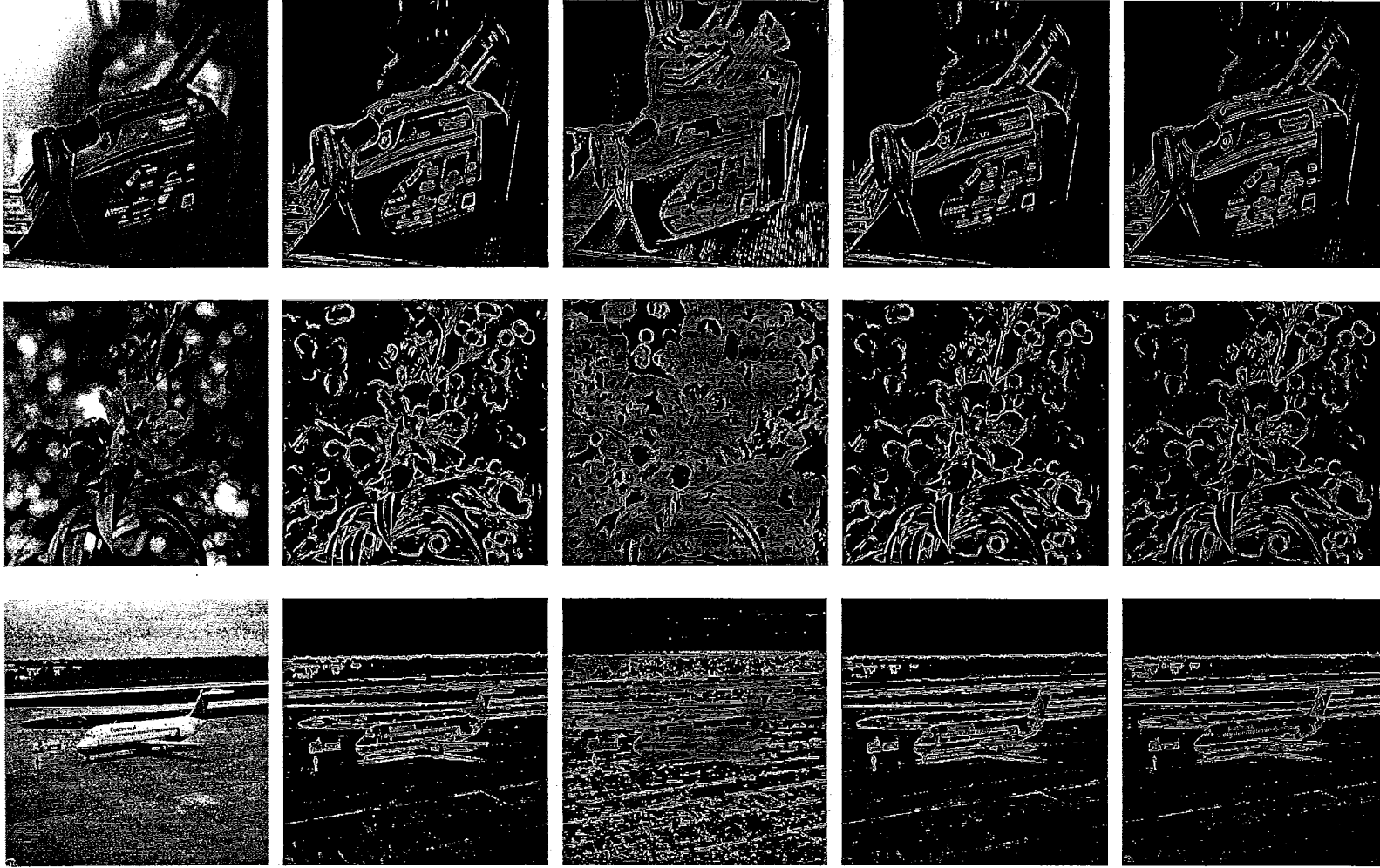
Bu filtreye Şekil 4.5'te verilen sonuçlar elde edilmiştir. Tek başına 5×5'lik Sobel işleci uygulandığında, büyük maskeyle daha fazla sayıda kenarın kapsanması nedeniyle, kenarların sık kullanılan Sobel işleci olan 3×3'lik maskeye göre daha kalın olarak bulunduğu söylenebilir. Yeni yöntemle, bağlılık haritasında “merkez noktayla bağlantısı olmayan” kenarların ortadan kaldırılmasıyla, tek başına 3×3'lük ve 5×5'lik Sobel işleçlerinden elde edilenlere göre daha ince kenarlar elde edilmiştir. Kolon dörtte orijinal görüntüye yeni yöntemin uygulanmasıyla, bağlantı haritası yöntemine ait aydınlık noktaları bağlamaya dayalı özellikten dolayı, aydınlık kenarların ortaya çıkarılması sağlandığından bazı karanlık bölgelerde kenarlar doğru olarak elde edilememiştir. Bu nedenle beşinci kolonda yer alan görüntüler orijinal görüntünün tersine yeni yöntemin uygulanması ile elde edilmiştir. Burada ise dördüncü kolonda tam olarak elde edilemeyen kenarların elde edildiği görülmüştür.

4-komşuluğa bakılarak bağlantı haritası oluşturulması nedeni ile yeni yöntemin uygulanması sırasında yönelmiş kenarların bulunması bazen sorunlar yaratmaktadır. Bu sorun 8-komşuluk kullanılarak ortadan kaldırıldığında bağlantı

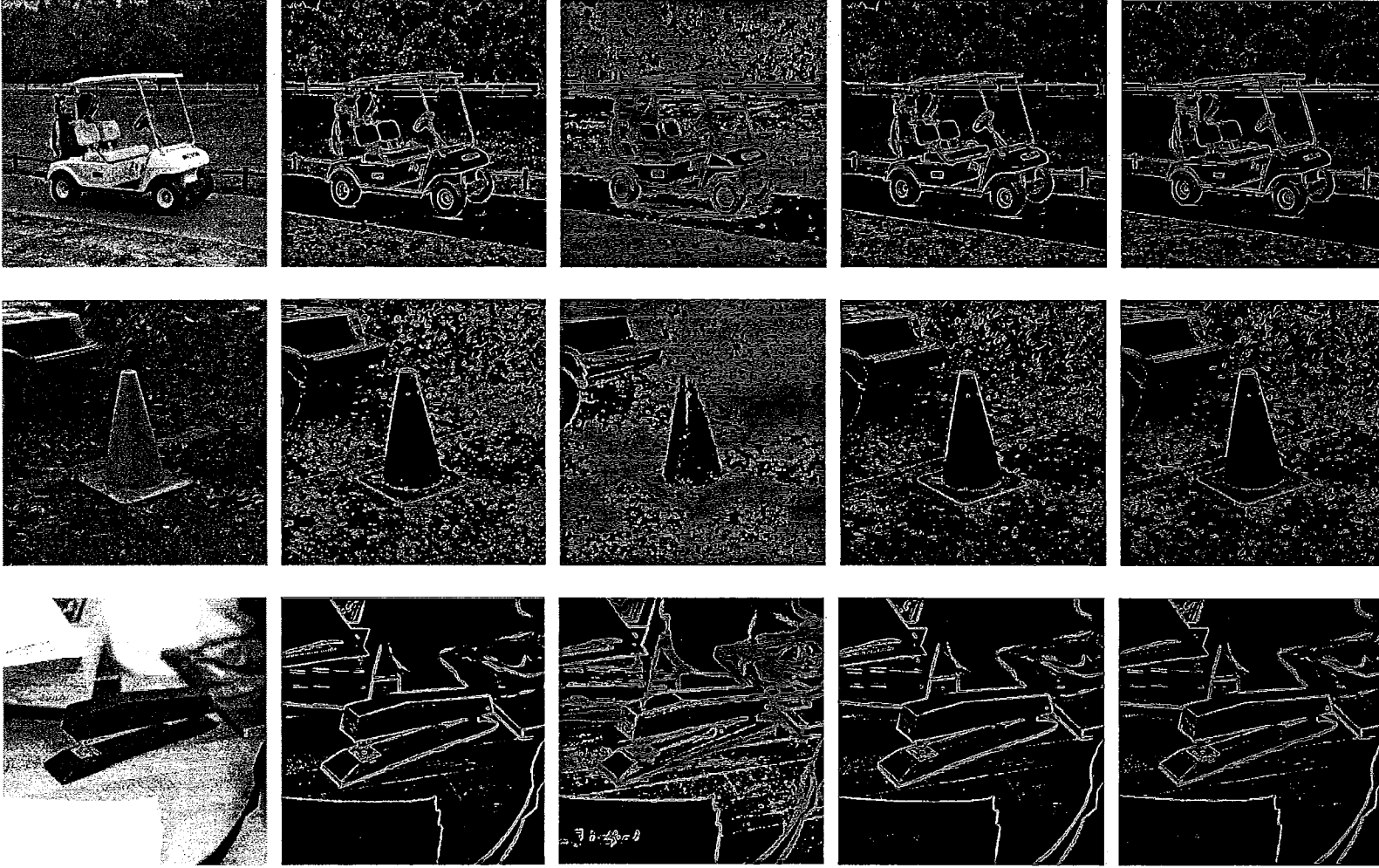
haritası yöntemine dayalı Sobel işleçlerin kullanımı ile oluşturulan yöntem daha kullanışlı hale getirilebilir.



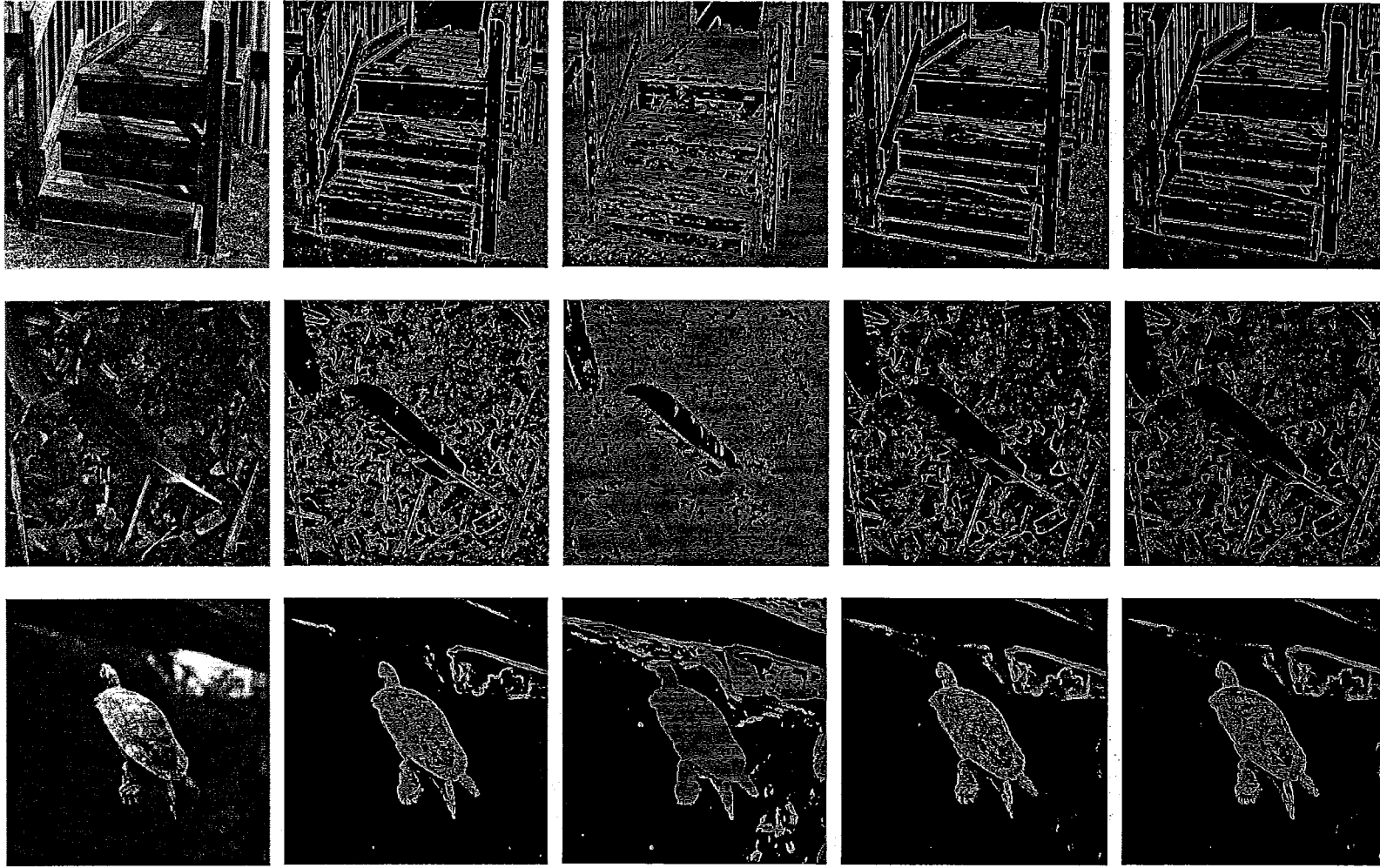
Şekil 4.5: Birinci kolon orijinal görüntü, ikinci kolon Sobel 3x3 sonucu, üçüncü kolon Sobel 5x5 sonucu, dördüncü kolon con7sob5 sonucu, beşinci kolon orijinal görüntünün tersine con7sob5 uygulanması ile elde edilen sonuç



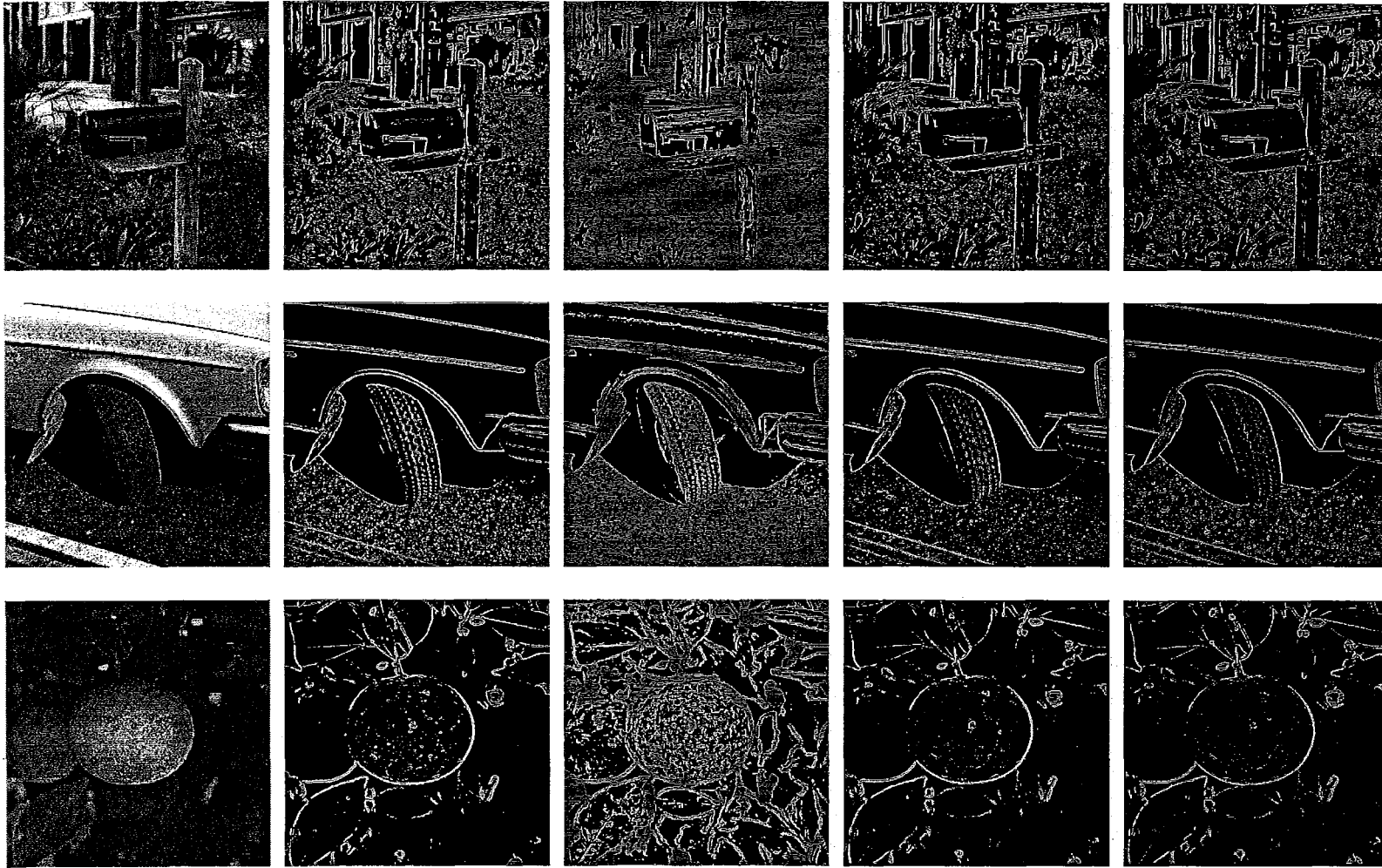
Şekil 4.5 (devamı): Birinci kolon orijinal görüntü, ikinci kolon Sobel 3x3 sonucu, üçüncü kolon Sobel 5x5 sonucu, dördüncü kolon con7sob5 sonucu, beşinci kolon orijinal görüntünün tersine con7sob5 uygulanması ile elde edilen sonuç



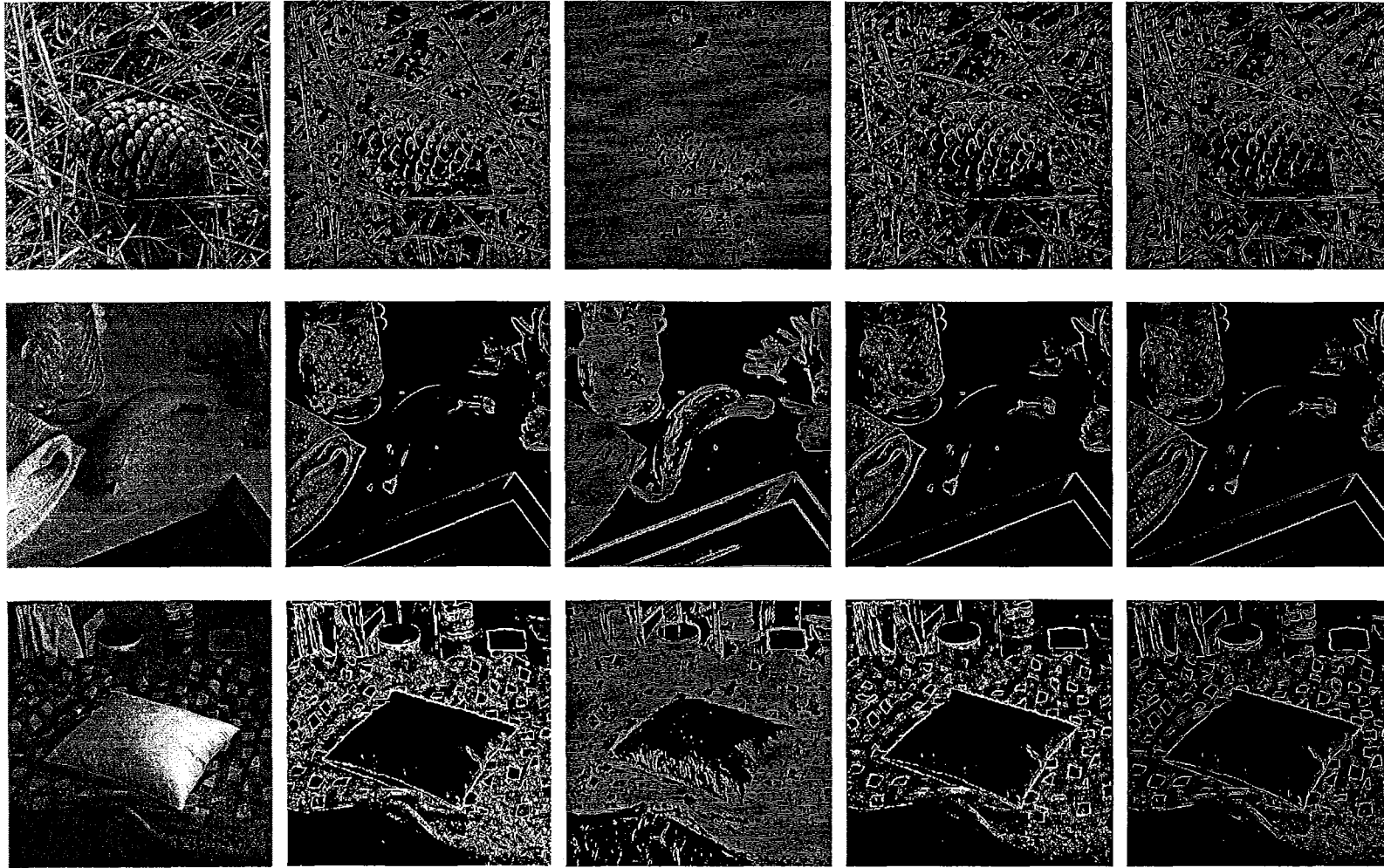
Şekil 4.5 (devamı): Birinci kolon orijinal görüntü, ikinci kolon Sobel 3x3 sonucu, üçüncü kolon Sobel 5x5 sonucu, dördüncü kolon con7sob5 sonucu, beşinci kolon orijinal görüntünün tersine con7sob5 uygulanması ile elde edilen sonuç



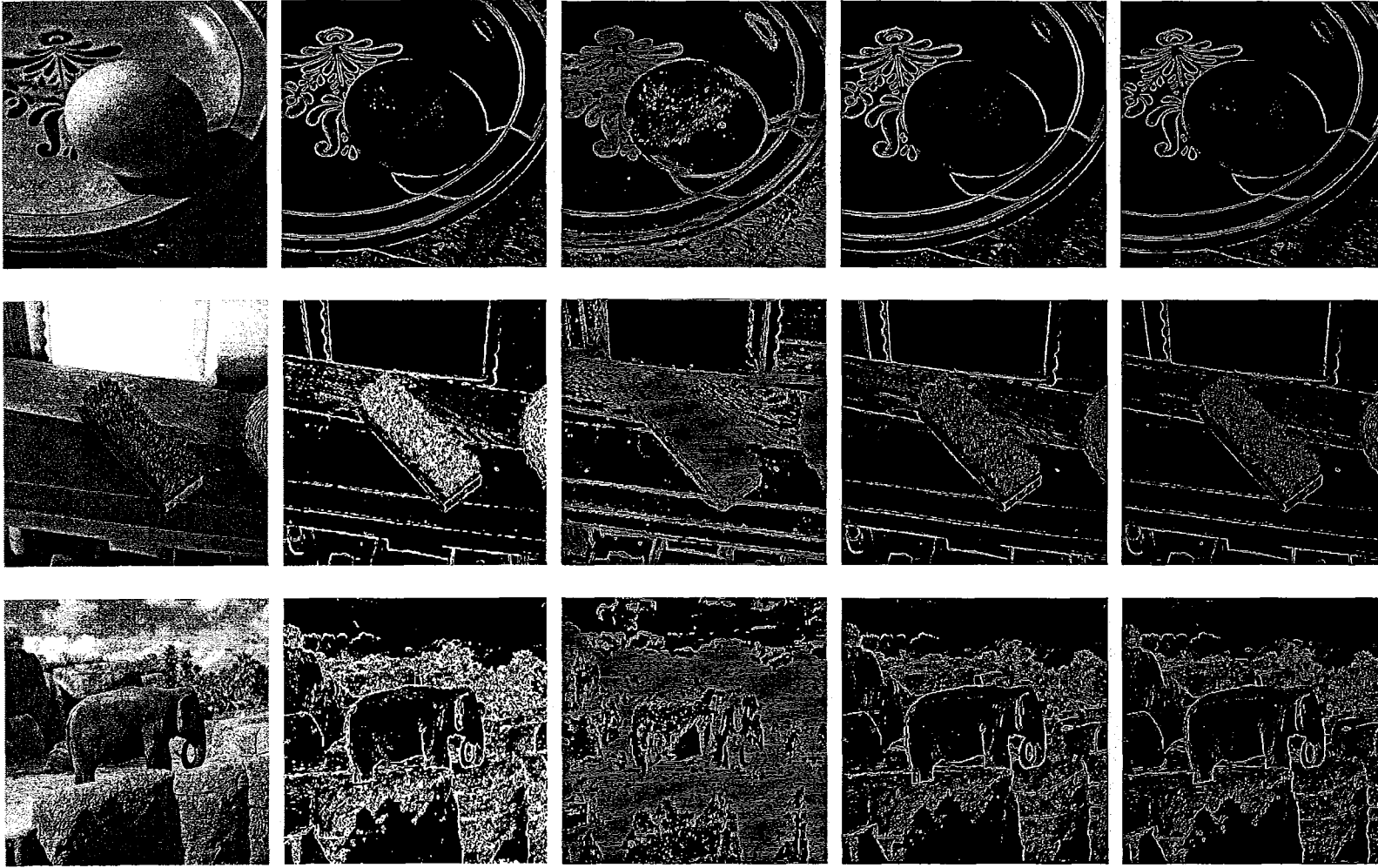
Şekil 4.5 (devamı): Birinci kolon orijinal görüntü, ikinci kolon Sobel 3x3 sonucu, üçüncü kolon Sobel 5x5 sonucu, dördüncü kolon con7sob5 sonucu, beşinci kolon orijinal görüntünün tersine con7sob5 uygulanması ile elde edilen sonuç



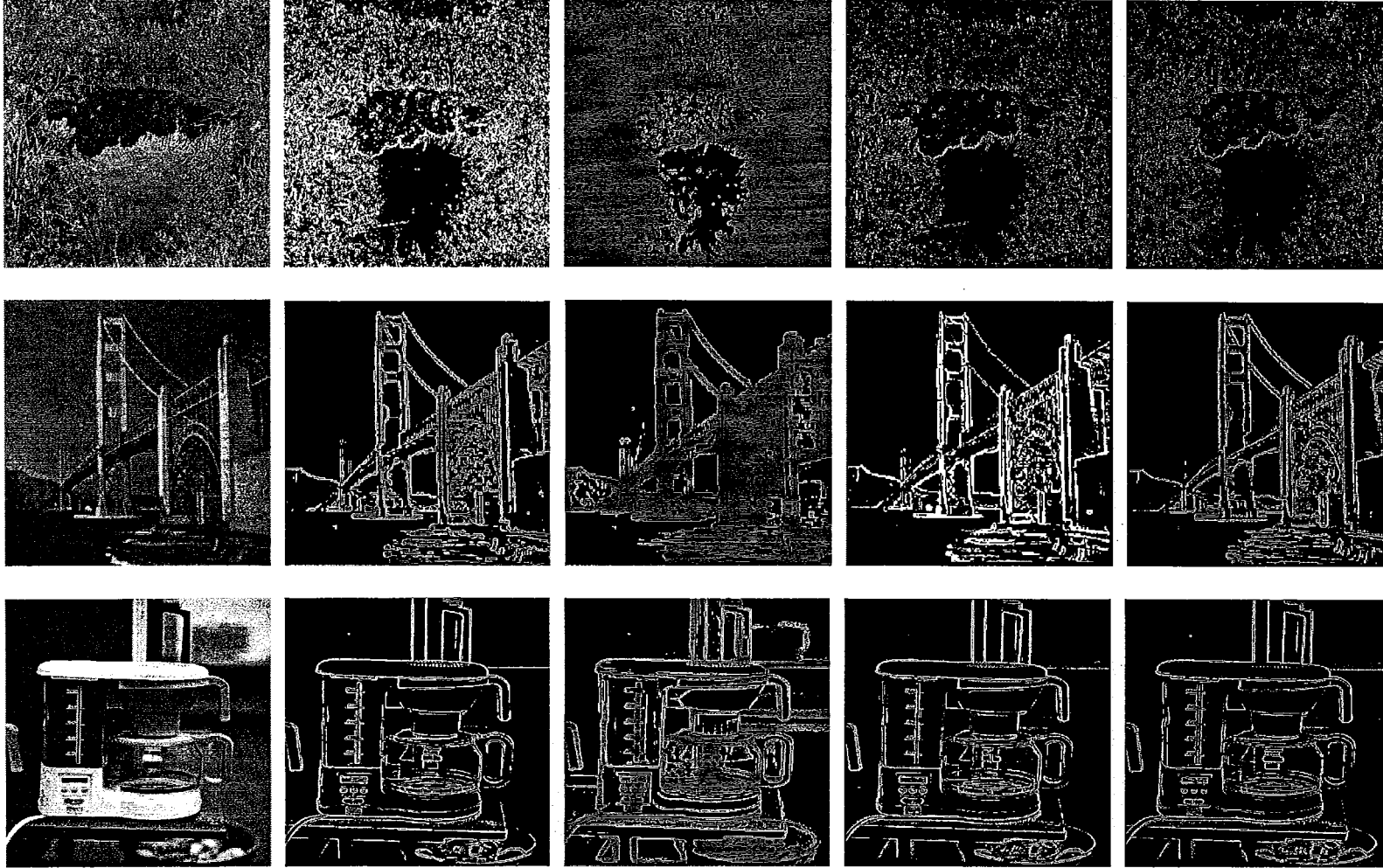
Şekil 4.5 (devamı): Birinci kolon orijinal görüntü, ikinci kolon Sobel 3x3 sonucu, üçüncü kolon Sobel 5x5 sonucu, dördüncü kolon con7sob5 sonucu, beşinci kolon orijinal görüntünün tersine con7sob5 uygulanması ile elde edilen sonuç



Şekil 4.5 (devamı): Birinci kolon orijinal görüntü, ikinci kolon Sobel 3x3 sonucu, üçüncü kolon Sobel 5x5 sonucu, dördüncü kolon con7sob5 sonucu, beşinci kolon orijinal görüntünün tersine con7sob5 uygulanması ile elde edilen sonuç



Şekil 4.5 (devamı): Birinci kolon orijinal görüntü, ikinci kolon Sobel 3x3 sonucu, üçüncü kolon Sobel 5x5 sonucu, dördüncü kolon con7sob5 sonucu, beşinci kolon orijinal görüntünün tersine con7sob5 uygulanması ile elde edilen sonuç



Şekil 4.5 (devamı): Birinci kolon orijinal görüntü, ikinci kolon Sobel 3x3 sonucu, üçüncü kolon Sobel 5x5 sonucu, dördüncü kolon con7sob5 sonucu, beşinci kolon orijinal görüntünün tersine con7sob5 uygulanması ile elde edilen sonuç

Şekil 4.6'da, geleneksel Sobel 3x3'lük işlecin ve 7x7'lik bağlantı haritası üzerine 5x5'lik Sobel işlecinin uygulanması ile elde edilen yeni yöntemin Gauss gürültü ilave edilmiş görüntülere uygulanması ile elde edilen sonuçlar yer almaktadır. Gauss gürültünün standart sapması σ , 0'dan başlayarak 10'arlık adımlarla 70'e kadar arttırılarak görüntülere ilave edilmiş ve bu görüntülere kenar bulma algoritmalarının uygulanmasıyla Şekil 4.6'daki sonuçlar elde edilmiştir. Sonuçlar incelendiğinde Sobel 3x3'lük işlecin standart sapmanın 30 ve daha fazla olduğu durumlarda kenar bilgilerini ortaya çıkaramadığı ve sonuç görüntüsünde çok fazla gürültünün yer aldığı görülmüştür. Yeni yöntemle ait sonuçlarda kenar bilgisinin $\sigma=60$ için elde edilen sonuç da dahil olmak üzere tüm görüntülerde yer aldığı ve sonuç görüntülerinde gürültünün azlığı dikkat çekicidir.



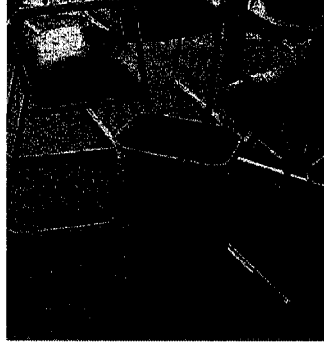
Şekil 4.6: $\sigma=0$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla Sobel 3x3 ile con7sob5 sonuçları



Şekil 4.6 (devamı): $\sigma=10$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla Sobel 3x3 ile con7sob5 sonuçları



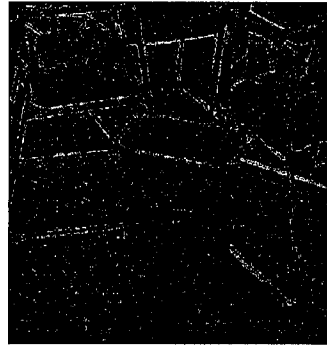
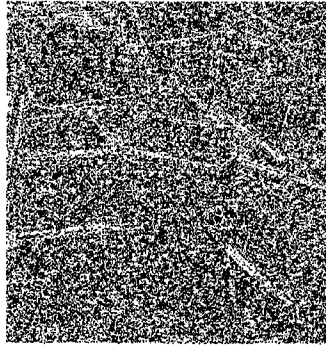
Şekil 4.6 (devamı): $\sigma=20$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla Sobel 3x3 ile con7sob5 sonuçları



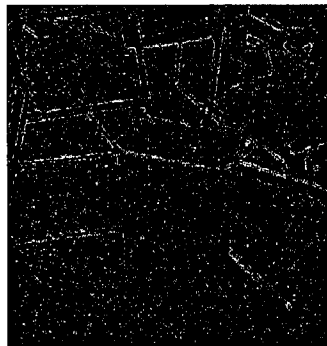
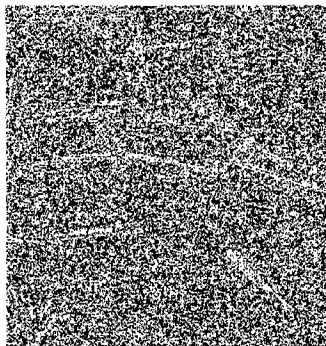
Şekil 4.6 (devamı): $\sigma=30$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla Sobel 3x3 ile con7sob5 sonuçları



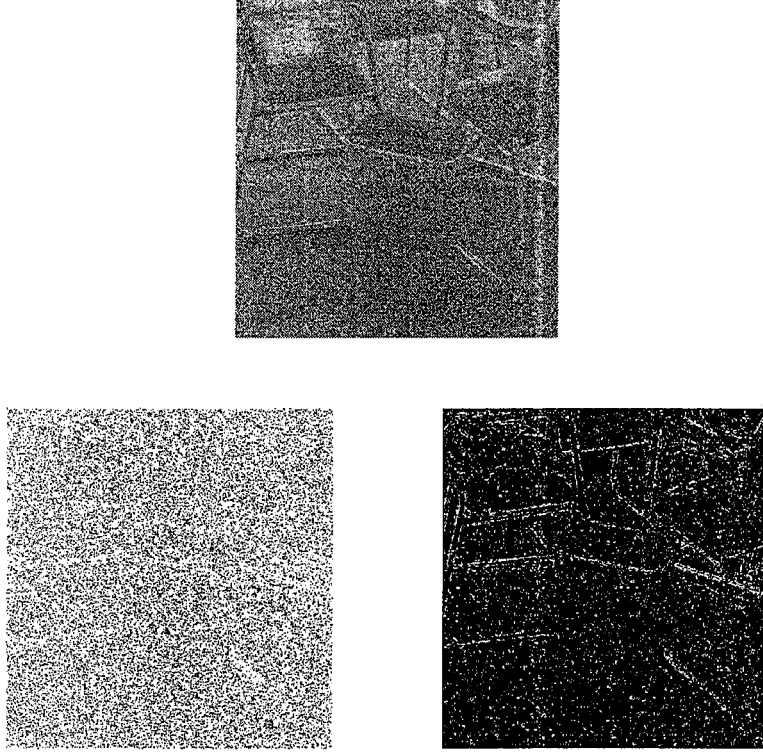
Şekil 4.6 (devamı): $\sigma=40$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla Sobel 3x3 ile con7sob5 sonuçları



Şekil 4.6 (devamı): $\sigma=50$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla Sobel 3x3 ile con7sob5 sonuçları



Şekil 4.6 (devamı): $\sigma=60$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla Sobel 3x3 ile con7sob5 sonuçları



Şekil 4.6 (devamı): $\sigma=70$ için Gauss gürültü uygulanmış orijinal görüntü ve sırasıyla Sobel 3x3 ile con7sob5 sonuçları

5. SONUÇLAR

Bu çalışmada, daha önce oluşturulan kenar algılama yöntemlerinden farklı olarak başlangıçta bir filtreleme kullanmadan görüntüye doğrudan uygulandığında algılanan kenarlarla beraber sonuç görüntüsünü oluşturabilen bir yöntem geliştirilmiştir. Çeşitli ebatlardaki maskeler üzerinde bağlantı haritası oluşturularak uygulanan bu yöntemle, çeşitli (aydınlatma, görüntüyü alan kameranın yeri gibi) nedenlerle gölgelerin yer aldığı bölgelerin daha iyi ayırt edildiği görülmüştür. Bunun yanında, yine yansımalar nedeniyle oluşan parlak yüzeylerin algılanmasında zorluklar olduğu Bölüm 4'te yer alan sonuçlarda gözlenmiştir.

İkinci bir yöntem olarak, yaygın olarak kullanılan Sobel işlecinin yer aldığı ve bağlantı haritasının kullanıldığı bir algoritma geliştirilmiştir. Yatay ve düşey yönlerde oluşturulmuş maskelerin kullanıldığı bu uygulamada, bağlantı haritası yönteminin yönden bağımsız olması ve Sobel maskesinin boyutunun 3x3'ten daha büyük seçilmesiyle algılanan kenar sayısı artmıştır. Daha büyük seçilen Sobel işleçlerin etkisi olarak algılanan kenarların genişlemesi beklenirken Bölüm 4.1'de ortaya konan bağlantı haritası yöntemiyle elde edilen kenarlar tek başına Sobel uygulanan görüntülerde algılanan kenarlardan daha ince olarak elde edilmiştir. Buna neden olarak, ön aşama olarak kullanılan bağlantı haritasının, pikseller arası bağlantı özelliklerini kullanan bir çeşit yumuşatma işlemi olduğu özelliği verilebilir. Merkez noktaya ilgili gradyan bileşenleri kullanılmakta ve merkez piksele bağlı olmayan piksel gruplarının oluşturduğu gradyanlar dikkate alınmamaktadır. Bu nedenle, kenarlar daha ince elde edilmektedir.

5x5'lik Sobel işleçlerin uygulanması sonucu elde edilen kenarların yer aldığı görüntüler ve 7x7'lik maskelerle oluşturulan bağlantı haritası bulunmuş görüntüye 5x5'lik Sobel işleçlerin uygulanmasıyla elde edilen kenarların yer aldığı görüntüler Şekil 4.4'de verilmektedir.

Bölüm 4.2'de bağlantı haritası ve Sobel işlecinin birleştirilmesi ile oluşturulan yeni uygulamanın Gauss, düzgün (uniform), students-t gürültüler altında incelenen bazı özellikleri istatistikler haline getirilmiştir. Gauss ve düzgün gürültüler, görüntülerde görülebilen türlerdir. Students-t gürültü türü, parametrik

olarak değiştirilebilen kuyukları (tailed) olan bir dağılım türüdür. Bu şekilde students-t dağılımı kullanılarak, geliştirilen yöntemin kuyuklu gürültülere karşı tepkisi incelenmiştir.

Bu yöntemde, bağlantı haritasının uygulandığı maske boyutu, uygulanan Sobel İşlecine göre daha büyük seçilmiştir. Bunun nedeni, bağlantı haritasına mümkün olduğunca daha fazla kenar girmesi ve merkez noktaya bağlı olanların korunmasının istenmesidir. Daha sonra, daha küçük boyutlarda olan Sobel işleci elde edilen haritalara uygulanmış ve sonuç diğer bir görüntü matrisine yerleştirilmiştir. Bağlantı haritası 9×9 maske üzerinde hesaplanarak, Sobel 5×5 işleci uygulanmış ve sonuç Con9Sob5 olarak isimlendirilmiştir. Benzer şekilde, Con7Sob5 olarak gösterilen sonuçlarda, 7×7 bağlantı haritası üzerine 5×5'lik Sobel uygulanmıştır. 3×3 ve 5×5'lik geleneksel Sobel uygulanarak, sırasıyla, Sobel 3×3 ve Sobel 5×5 olarak gösterilmektedir.

Geliştirilen yöntemin, gürültü ortamında adım kenara (step edge) olan tepkisini incelemek için, 10000×30 boyutunda suni olarak üretilmiş bir görüntü kullanılmıştır. Görüntünün, 16. kolonuna bir süreksizlik yerleştirilmiştir. Sol tarafından bulunan sabit yüzeydeki piksel değeri 80 olarak yerleştirilirken, sağ tarafta bulunan sabit alanın değeri 180 olarak verilmiştir. Bu görüntü üzerine, değişen miktarlarda Gauss, düzgün ve students-t dağılımındaki gürültü eklenmiştir.

Çizelgelerdeki (5.13 – 5.24) değerler incelendiğinde sadece Sobel işleçlere ait elde edilen değerlerin bağlantı haritası ve Sobel işleçlerin ortak kullanılarak oluşturulduğu yonteme ait değerlerden büyük olduğu görülmektedir. Bu beklenen bir sonuçtur, çünkü, bağlantı haritası kullanan yöntemde maske içine gelen kenarlar daha az sayıda olmaktadır. Bu durum işlecin tepkisinin bir ölçüde az olmasının nedenidir.

Oluşturulan 10000×30 boyutundaki suni görüntüye, sıfırdan başlayarak yetmiş kadar 10 adımla değişen varyanslarda Gauss, düzgün (uniform) ve students-t dağılımlarındaki gürültüler eklenmiştir. Geleneksel Sobel ve bağlantı haritası ön aşamasını kullanan Sobel işlecinin sonuçları Çizelgeler 5.13 – 5.24'te yer almaktadır. Students-t dağılımın sonsuzda sıfır olan bir yapıya neden olmasından dolayı sıfır varyansa ait sonuçlar tanımsızdır. Bu nedenle student-t

dağılıma ait sonuçlar 10 ile 70 arasında değişen varyanslar için tablolarda yer almaktadır.

5.1. Kenar Algılama Algoritması – 1'e Ait Sonuçlar

Bölüm 4.1'de açıklandığı gibi 7×7 ve 9×9 boyutlarında maskeler kullanılarak bağlantı haritası oluşturuldu. Orijinal görüntüye Gauss, düzgün, students-t dağılımlı gürültüler ilave edildi. Gürültü ilave edilmiş görüntülerden bağlantı haritası kullanılarak kenarların algılanması için oluşturulan algoritma kullanılarak kenarların yer aldığı görüntü elde edildi.

10000×30 boyutunda suni görüntü üzerine, yöntemin uygulanmasıyla elde edilen sonuç görüntülerdeki bütün satırlar toplanarak 1×30 'luk bir vektör elde edilmiş ve normalizasyon amacıyla 10,000'e bölünmüştür. Bu şekilde elde edilen vektör, sonuçta elde edilen kenarın ortalama profilini vermektedir.

5.1.1. Maksimum Değerler

Çizelge 5.1-5.3'te yer alan 1×30 'luk vektörün en büyük değerleridir. Maksimum değer, filtrenin süreksizlik üzerinde verebildiği en büyük değerlerdir. Bu değer ne kadar büyük olursa, filtrenin daha iyi çalıştığı anlamına gelmektedir. Bununla birlikte, görüntülerdeki gürültü, süzülmediği zaman bu değerlerin daha da artmasına neden olabilmektedir. Ön aşamada, bağlantı haritasının süzme işlemini gerçekleştirmesi nedeniyle, geliştirilen 1. algoritmanın tepkisinin diğerlerine göre daha az olması beklenebilir.

7×7 ve 9×9 'luk maskelerle bağlantı haritasının oluşturulmasıyla kenarların yer aldığı görüntüler sırasıyla $ea7 \times 7$ ve $ea9 \times 9$ algoritmalarının çalıştırılmasıyla elde edilmektedir. Kenarı belirleyen yoğunluk değerleri her iki algoritma sonucunda birbirine yakın olarak elde edilmiştir. Bağlantı haritası bulunurken, 7×7 ve 9×9 komşuluklarda, bağlantı açısından önemli bir değişiklik olmamaktadır, çünkü, daha büyük pencereye daha fazla birbiriyle ilgisiz nesne parçası girmekte ve bunlar da bağlantı haritasında süzülmemektedir.

Çizelge 5.1: Gauss gürültü uygulanmış görüntülerde kenar üzerindeki maksimum kenar algılama değerleri

	$\sigma = 0$	$\sigma = 10$	$\sigma = 20$	$\sigma = 30$	$\sigma = 40$	$\sigma = 50$	$\sigma = 60$	$\sigma = 70$
ea 7x7	0	0.74	0.74	0.64	0.64	0.54	0.51	0.51
ea 9x9	0.74	0.74	0.74	0.64	0.56	0.56	0.61	0.54

Çizelge 5.2: Düzgün gürültü uygulanmış görüntülerde maksimum yoğunluk değerleri

	$\sigma = 0$	$\sigma = \sqrt{10}$	$\sigma = \sqrt{20}$	$\sigma = \sqrt{30}$	$\sigma = \sqrt{40}$	$\sigma = \sqrt{50}$	$\sigma = \sqrt{60}$	$\sigma = \sqrt{70}$
ea 7x7	0.74	0.74	0.71	0.66	0.59	0.48	0.56	0.49
ea 9x9	0.74	0.74	0.74	0.69	0.66	0.51	0.48	0.51

Çizelge 5.3: Student-t gürültü uygulanmış görüntülerde maksimum yoğunluk değeri

	$\sigma = 0$	$\sigma = 10$	$\sigma = 20$	$\sigma = 30$	$\sigma = 40$	$\sigma = 50$	$\sigma = 60$	$\sigma = 70$
ea 7x7	0	0.74	0.74	0.74	0.74	0.74	0.74	0.74
ea 9x9	0	0.74	0.74	0.74	0.74	0.74	0.74	0.74

5.1.2. Yumuşatma Oranları

Yumuşatma oranı, filtrenin düzgün alanlardaki tepkisini göstermektedir. Tepkinin bu bölgede küçük olması, gürültü bileşenlerinin sanal kenar olarak algılanmaması anlamına gelmektedir.

ea7x7 ve ea9x9 algoritmaları sonucunda Çizelge 5.4-5.6'da yer alan yoğunluk değerlerinin ortalamaları Gauss ve düzgün dağılımlar için sıfıra yakın değerlerdir. Students-t dağılımının uygulandığı görüntülerden elde edilen sonuçlar ise sıfırdır. Bunun nedeni ea7x7 ve ea9x9 algoritmaları sonucunda kenarın yer aldığı süreksizliğin (suni görüntüde 16. piksel) daha belirgin olarak tespit edilmesi ve dolayısıyla, ortalamanın hesaplandığı 1-15 ve 17-30 pikselleri arasında yoğunluklarda bir değişim olmamasıdır. Student's t dağılımı, daha uç noktalardaki değerler ürettiğinden, bağlantı haritasıyla daha kolay şekilde filtrelenebilmektedir.

Çizelge 5.4: Gauss gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin ortalaması

	$\sigma = 0$	$\sigma = 10$	$\sigma = 20$	$\sigma = 30$	$\sigma = 40$	$\sigma = 50$	$\sigma = 60$	$\sigma = 70$
ea 7x7	0	0	0.011	0.09	0.14	0.19	0.22	0.27
ea 9x9	0	0	0.011	0.09	0.14	0.18	0.23	0.25

5.2. Kenar Algılama Algoritması -2'ye Ait Sonuçlar

5.2.1. Maksimum Değerler

Suni görüntü üzerine, çeşitli boyutlarda yatay ve düşey çalışan Sobel işleçleri uygulanmış ve yöntemlerden elde edilen sonuç görüntülerindeki bütün satırlar toplanarak 1×30 'luk bir vektör elde edilmiştir ve sonuç 10,000'e bölünmüştür. Bu şekilde elde edilen vektör, kenarın bulunduğu yerin özellikleriyle ilgili bilgi vermektedir. Kenar üzerinde elde edilen maksimum değer bilgileri, Gauss gürültü için Çizelge 5.13'de, düzgün gürültü için Çizelge 5.14'te ve Students t dağılımı için Çizelge 5.15'te verilmiştir.

Çizelge 5.13: Gauss gürültü uygulanmış görüntülerde kenar üzerindeki maksimum kenar algılama değerleri

	$\sigma = 0$	$\sigma = 10$	$\sigma = 20$	$\sigma = 30$	$\sigma = 40$	$\sigma = 50$	$\sigma = 60$	$\sigma = 70$
Sobel 3x3	24.401	24.498	25.438	26.1	27.437	29.291	27.518	25.242
Sobel 5x5	1182.488	1169.143	1133.687	1157.086	1058.075	1317.605	1017.532	1316.212
Con7Sob5	0.98	0.918	0.878	0.809	0.722	0.853	0.846	0.69
Con9Sob5	0.945	0.919	0.874	0.907	0.725	0.645	0.727	0.813

Çizelge 5.14: Düzgün gürültü uygulanmış görüntülerde maksimum yoğunluk değerleri

	$\sigma = 0$	$\sigma = \sqrt{10}$	$\sigma = \sqrt{20}$	$\sigma = \sqrt{30}$	$\sigma = \sqrt{40}$	$\sigma = \sqrt{50}$	$\sigma = \sqrt{60}$	$\sigma = \sqrt{70}$
Sobel 3x3	24.401	24.498	25.438	26.1	27.437	29.291	27.518	25.242
Sobel 5x5	1182.488	1169.143	1133.687	1157.086	1058.075	1317.605	1017.532	1316.21
Con7Sob5	0.98	0.918	0.878	0.809	0.722	0.853	0.846	0.69
Con9Sob5	0.945	0.883	0.866	0.883	0.751	0.801	0.901	0.614

Çizelge 5.15: Student-t gürültü uygulanmış görüntülerde maksimum yoğunluk değeri

	$\sigma = 10$	$\sigma = 20$	$\sigma = 30$	$\sigma = 40$	$\sigma = 50$	$\sigma = 60$	$\sigma = 70$
Sobel 3x3	24.438	24.512	24.49	24.444	24.409	24.388	24.333
Sobel 5x5	1179.246	1181.028	1179.488	1180.92	1181.306	1182.916	1184.576
Con7Sob5	0.977	0.971	0.972	0.974	0.977	0.972	0.972
Con9Sob5	0.936	0.94	0.936	0.94	0.94	0.939	0.94

Çizelge 5.13 - 5.15'de yer alan değerler incelendiğinde Sobel 5×5 'lik işlecin sonuçları yatay ve düşey yönde kullanılan maskeleri oluşturan ağırlık katsayılarının büyük olması nedeni ile diğer uygulama sonuçlarından oldukça

büyüktür. Bağlantı haritası kullanıldığında bir yumuşatma işlemi yapılmış olması nedeniyle kenar bilgisinin yer aldığı bölgeye ait bilgiler daha küçük değere sahiptir. Görüntüde kenarın yer aldığı süreksizliklerin belirlendiği yerleri gösterdiği düşünüldüğünde tabloda yer alan büyük değerlerin daha iyi sonuçlar olduğu düşünülebilir. Aslında maksimum değerın büyüklüğünden çok daha önemlisi bu maksimumların yer aldığı bölgelerin 1x30'luk vektörde sahip olduğu piksel sayılarıdır. Geleneksel 3x3'lük Sobel işleç için 14-17 arasında 3 piksellik, Sobel 5x5'lik işleç için 13-18 arasında 5 piksellik, Con7Sob5 ve Con9Sob5 için 12-15 arasında 3 piksellik bir genişlik oluşmuştur. Bu piksel sayıları bulunan kenarın kalınlığını temsil etmektedir. Bağlantı haritası kullanılarak oluşturulan Con7Sob5 ve Con9Sob5 algoritmaları sonucunda 5x5'lik Sobel işleçlerden daha ince ve Sobel 3x3'lük işleçle aynı kalınlıkta kenarlar elde edildiği buradan da anlaşılmaktadır.

5.2.2. Yumuşatma Oranları

Uygulanan yöntemlerin yumuşatma özellikleri, adım kenarın dışındaki pikseller kullanılarak gerçekleştirilmiştir. Özellikle gürültülü ortamlarda, gürültünün özelliğine göre verdikleri tepkilerin incelenmesi, işlemlerin tanımda ve uygulama alanlarının bulunmasında önemli bir aşamadır. Bu amaçla, her satır boyunca, Sobel 3x3 için 2-14 ve 17-29 arasındaki piksel değerlerinin, Sobel 5x5 için 3-13 ve 18-28 arasındaki piksel değerlerinin, Con7Sob5 için 4-12 ve 15-27 arasındaki piksel değerlerinin, Con9Sob5 için 3-12 ve 15-26 arasındaki piksel değerlerinin ortalaması alınmıştır.

Sonuçlar, Gauss gürültü için Çizelge 5.16'da, düzgün gürültü için Çizelge 5.17'de ve Students-t dağılımı için Çizelge 5.18'de verilmiştir. Sobel 5x5 için elde edilen varyans değeri, çok sayıda gürültünün, kenar olduğu düşünülerek işleme girmesiyle büyük olmaktadır. Ama Con7Sob5 ve Con9Sob5'te gürültülerin bir kısmının, merkeze bağlı olmadığı düşünülerek bağlantı haritasında görünmemesi nedeniyle süzüldüğü anlaşılmaktadır. Bu nedenle, geleneksel Sobel 5x5'ten elde edilen sonuçlara göre daha küçük değerlere sahip olan bağlantı haritası yöntemleri daha iyi sonuç vermektedir.

Çizelge 5.16: Gauss gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin ortalaması

	$\sigma = 0$	$\sigma = 10$	$\sigma = 20$	$\sigma = 30$	$\sigma = 40$	$\sigma = 50$	$\sigma = 60$	$\sigma = 70$
Sobel 3x3	0.485	0.588	0.966	1.497	2.314	3.088	3.571	4.971
Sobel 5x5	28.13	30.373	35.342	47.08	56.938	66.745	74.098	102.839
Con7Sob5	0	0.044	0.079	0.117	0.138	0.167	0.177	0.23
Con9Sob5	0	0.039	0.079	0.113	0.141	0.159	0.173	0.221

Çizelge 5.17: Düzgün gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin ortalaması

	$\sigma = 0$	$\sigma = \sqrt{10}$	$\sigma = \sqrt{20}$	$\sigma = \sqrt{30}$	$\sigma = \sqrt{40}$	$\sigma = \sqrt{50}$	$\sigma = \sqrt{60}$	$\sigma = \sqrt{70}$
Sobel 3x3	0.485	0.588	0.966	1.497	2.314	3.088	3.571	4.971
Sobel 5x5	28.13	30.373	35.342	47.08	56.938	66.745	74.098	102.839
Con7Sob5	0	0.044	0.079	0.117	0.138	0.167	0.177	0.23
Con9Sob5	0	0.04	0.07	0.117	0.145	0.157	0.195	0.198

Çizelge 5.18: Students-t gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin ortalaması

	$\sigma = 10$	$\sigma = 20$	$\sigma = 30$	$\sigma = 40$	$\sigma = 50$	$\sigma = 60$	$\sigma = 70$
Sobel 3x3	0.488	0.486	0.488	0.487	0.487	0.485	0.487
Sobel 5x5	28.191	28.167	28.13	28.121	28.172	28.115	28.151
Con7Sob5	0.004	0.004	0.003	0.004	0.004	0.003	0.004
Con9Sob5	0.004	0.003	0.003	0.004	0.003	0.003	0.003

Çizelge 5.15–5.18’de yer alan değerler incelendiğinde Sobel 5×5’lik işlecin sonuçları yatay ve düşey yönde kullanılan maskeleri oluşturan ağırlık katsayılarının büyük olması nedeni ile diğer uygulama sonuçlarından oldukça büyüktür. Bağlantı haritası oluşturulan görüntülere 5×5’lik Sobel işlecin uygulanması ile elde edilen değerler (tepe noktasına ait olan yoğunluk değeri dahil edilmeden hesaplanan yoğunluk değerlerinin ortalaması) sadece 5×5’lik Sobel işlecin uygulanması ile elde edilen değerlerden oldukça küçüktür. Çünkü Sobel işleç uygulanmadan önce görüntüye hiçbir filtreleme uygulanmaması ve dolayısı ile düzgülleştirme yapılmadan kenar algılama algoritmasının uygulanması ile yoğunluk değerlerinin ortalaması con7sob5 ve con9sob5 uygulanması ile elde edilen değerlerden büyük olarak elde edilmektedir. Bunun nedeni görüntünün bağlantı haritası oluşturulurken aslında bir anlamda

düzgünleştirme yapılıyor olmasıdır. Bağlantı haritası hesaplanırken kullanılan maskenin boyutu büyüdükçe görüntüdeki düzgünleştirme miktarı arttığından dolayı Con7Sob5 yönteminin uygulanması ile elde edilen değerler Con9Sob5 yönteminin uygulanması ile elde edilen değerlerden büyüktür.

5.2.3. Tepe Yüksekliği

Düzgün alanlarda, kenar işlecinin tepkisi incelendiğinde, belirli bir ortalama değeri sağladığı görülmektedir. Kenarın bulunduğu tepe noktasıyla, düzgün alanlardaki ortalama arasındaki fark aslında kenar işlecinin verdiği tepkidir. Bu değeri ölçebilmek için, Çizelgeler 5.13-5.15 ve 5.16-5.18 gruplarında verilen değerler birbirinden çıkarılarak Çizelge 5.19, 5.20 ve 5.21 elde edilmiştir. Çizelgelerdeki değerler incelendiğinde büyük olan değerlerin istenilen sonuç yani daha iyi olduğu düşünülebilir. Değerlerin büyük olması sahte (phantom) kenarların yer aldığı bir algılama sonucunu göstermektedir. Elde edilen sonuçlardan, bağlantı haritası yönteminin düzgün alanlarda belirli bir yumuşatma sağlarken, sahte (phantom) kenarların belirmesini engellediği ve tepe noktasının bulunmasında geleneksel Sobel işlecine göre daha düşük tepki verdiği ve çizelgelerde yer alan küçük değerlerin daha iyi sonucu işaret ettiği anlaşılmaktadır.

Çizelge 5.19: Gauss gürültü uygulanmış görüntülerde tepe noktası ve diğer yoğunluk değerlerinin ortalaması arasındaki fark

	$\sigma = 0$	$\sigma = 10$	$\sigma = 20$	$\sigma = 30$	$\sigma = 40$	$\sigma = 50$	$\sigma = 60$	$\sigma = 70$
Sobel 3x3	23.916	23.91	24.472	24.604	25.123	26.203	23.947	20.27
Sobel 5x5	1154.358	1138.77	1098.344	1110.006	1001.137	1250.86	943.433	1213.372
Con7Sob5	0.98	0.874	0.799	0.692	0.584	0.686	0.669	0.46
Con9Sob5	0.945	0.881	0.795	0.795	0.585	0.486	0.554	0.592

Çizelge 5.20: Uniform gürültü uygulanmış görüntülerde tepe noktası ve diğer yoğunluk değerlerinin ortalaması arasındaki fark

	$\sigma = 0$	$\sigma = \sqrt{10}$	$\sigma = \sqrt{20}$	$\sigma = \sqrt{30}$	$\sigma = \sqrt{40}$	$\sigma = \sqrt{50}$	$\sigma = \sqrt{60}$	$\sigma = \sqrt{70}$
Sobel 3x3	23.916	23.91	24.472	24.604	25.123	26.203	23.947	20.27
Sobel 5x5	1154.358	1138.77	1098.344	1110.006	1001.137	1250.86	943.433	1213.372
Con7Sob5	0.98	0.874	0.799	0.692	0.584	0.686	0.669	0.46
Con9Sob5	0.945	0.843	0.797	0.766	0.607	0.644	0.706	0.416

Çizelge 5.21: Student-t gürültü uygulanmış görüntülerde tepe noktası ve diğer yoğunluk değerlerinin ortalaması arasındaki fark

	$\sigma = 10$	$\sigma = 20$	$\sigma = 30$	$\sigma = 40$	$\sigma = 50$	$\sigma = 60$	$\sigma = 70$
Sobel 3x3	23.95	24.027	24.002	23.957	23.922	23.903	23.846
Sobel 5x5	1151.055	1152.861	1151.358	1152.799	1153.134	1154.8	1156.425
Con7Sob5	0.973	0.967	0.969	0.971	0.973	0.969	0.968
Con9Sob5	0.932	0.936	0.933	0.936	0.937	0.935	0.937

5.2.4. Düzgün Alanlardaki Dalgalanma Varyansları

Çizelge 5.22: Gauss gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin varyansı.

	$\sigma = 0$	$\sigma = 10$	$\sigma = 20$	$\sigma = 30$	$\sigma = 40$	$\sigma = 50$	$\sigma = 60$	$\sigma = 70$
Sobel 3x3	0.110	0.132	0.159	0.129	0.241	0.454	0.645	1.721
Sobel 5x5	372.243	359.280	354.509	487.885	593.315	640.143	151.984	583.641
Con7Sob5	0	0.001	0.001	0.001	0.003	0.002	0.003	0.005
Con9Sob5	0	0	0.002	0.002	0.002	0.003	0.004	0.006

Çizelge 5.23: Düzgün gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin varyansı

	$\sigma = 0$	$\sigma = \sqrt{10}$	$\sigma = \sqrt{20}$	$\sigma = \sqrt{30}$	$\sigma = \sqrt{40}$	$\sigma = \sqrt{50}$	$\sigma = \sqrt{60}$	$\sigma = \sqrt{70}$
Sobel 3x3	0.11	0.132	0.159	0.129	0.241	0.454	0.645	1.721
Sobel 5x5	372.243	359.28	354.509	487.885	593.315	640.143	151.984	583.641
Con7Sob5	0	0.001	0.001	0.001	0.003	0.002	0.003	0.005
Con9Sob5	0	0.001	0.002	0.002	0.003	0.003	0.003	0.003

Çizelge 5.24: Student-t gürültü uygulanmış görüntülerde tepe noktası hariç diğer yoğunluk değerlerinin varyansı

	$\sigma = 10$	$\sigma = 20$	$\sigma = 30$	$\sigma = 40$	$\sigma = 50$	$\sigma = 60$	$\sigma = 70$
Sobel 3x3	0.11	0.11	0.11	0.11	0.11	0.11	0.109
Sobel 5x5	369.967	370.273	371.147	368.371	370.637	370.056	369.803
Con7Sob5	0	0	0	0	0	0	0
Con9Sob5	0	0	0	0	0	0	0

Çizelge 5.21-5.24'te yer alan varyans değerleri incelendiğinde bağlantı haritası ve 5×5'lik Sobel işlecin beraber kullanılması ile oluşturulan yöntem sonuçlarının (Con7Sob5 ve Con9Sob5) sadece 5×5'lik Sobel işlecin uygulanması ile elde edilen sonuçlardan küçük hatta 0 olduğu görülebilir. Bunun nedeni students-t dağılımının kuyruk (tail) bölgesinde yer alan gürültülerin bağlantı

haritasının merkeze bağılı olmayan pikselleri ortadan kaldırması özelliğinden dolayı bağlantı haritası oluşturulan görüntüde yer almamasıdır. Kenar algılama sürecinden sonra oluşturulan 1×30 'luk vektörde tepe noktası hariç Sobel 3×3 'lük işleç için 2-14 ve 17-29 arası, Sobel 5×5 'lik işleç için 3-13 ve 18-28 arası, Con7Sob5 için 4-12 ve 15-27 arası, Con9Sob5 için 5-12 ve 15-26 arası pikseller kullanılarak dalgalanma varyansları hesaplanmıştır. Çizelgelerden görüleceği gibi bağlantı haritası kullanılarak oluşturulan yöntem sonuçları Sobel işleçlere göre Gauss, düzgün ve students-t dağılımların uygulanmış olduğu görüntü için daha küçük değerlere sahip olmasına rağmen en iyi sonuç students-t dağılımının uygulandığı görüntülerden elde edilmiştir. Yine de tüm gürültüler için Con7Sob5 ve Con9Sob5 uygulanması durumunda Sobel işleçlere göre daha iyi sonuçlar elde edilmektedir.

5.2.5. Algoritma Hızları

Çizelge 5.25: Değişik boyutlu görüntüler üzerinde çalıştırılan algoritmaların hızı (saniye olarak)

	64×64	128×128	256×256	512×512	1024×1024
Sobel 3×3	0.1	0.4	1.7	7	27.9
Sobel 5×5	0.2	0.5	2.2	8.7	35.8
Con7Sob5	2.8	10.1	47.1	194.6	672.7
Con9Sob5	3.8	14.5	70.1	289.7	1160.4

Çizelge 5.25'te, 64×64 , 128×128 , 256×256 , 512×512 ve 1024×1024 boyutlarında beş farklı görüntü için Matlab paket programı kullanılarak oluşturulan, Sobel işleçlere ve bağlantı haritası ile Sobel işlecin birleştirilmesi ile oluşturulan yeni yonteme ait programların çalıştırılması ile elde edilen uygulama sürelerine (saniye olarak) ait değerler yer almaktadır. Bağlantı haritası hesaplanarak oluşturulan con7sob5 ve con9sob5 programları bağlantı haritası hesabının görüntünün boyutuna bağılı olarak ek bir süre gerektirmesi nedeni ile tek başına Sobel işlecin uygulanması ile çalıştırılan programdan daha uzun sürede sonuca ulaşmaktadır.

5.3. Gelecekte Yapılması Planlanan Çalışmalar

Bu tez çalışmasında, bulanık topoloji kavramları kullanan, iki algoritma geliştirilmiştir. Özellikle, Sobel işleci ve bağlantı haritası kullanan yöntemin (Con7Sob5), geleneksel 5x5 Sobel işlecinden, bazı durumlarda daha iyi çalıştığı gösterilmiştir. Özellikle gürültülü resimlerde, Con7Sob5 işleminin daha başarılı sonuç verdiği görülmüştür.

Sentetik görüntüler üzerinde yapılan çalışmalar yanında, bir ön aşama olarak kenar algılamanın kullanıldığı sistemlerde, bu tez çalışmasında geliştirilen algoritmaların uygulanması gerekmektedir. Bu şekilde, kenar algılama algoritmalarının, sistemin genel sonucuna etkisi incelenebilir ve daha iyi nesne karşılaştırma olanağı sağlanmış olabilir. Böyle bir çalışmanın gerektirdikleri, bu tez çalışmasında yapılamayacak kadar geniş kapsamlıdır ve gelecekte planlanmaktadır.

Genel olarak, bağlantı haritası, diğer görüntü işleme işlemlerinde de kullanılabilecek türde genel amaçlı bir yöntemdir. Görüntü iyileştirme, köşe algılama gibi işlemlerde de kullanılabileceği görülmektedir.

KAYNAKLAR

- [1] LOH, A. W. K., ROBEY, M. C. ve WEST, G. A. W., *Analysis of the Interaction Between Edge and Line Finding Techniques*, The Journal of the Pattern Recognition Society , **34**, 1127-1146, (2001).
- [2] MOON, H., CHELLAPPA, R. ve ROSENFELD, A., *Optimal Edge- Based Shape Detection*, IEEE Transactions on Image Processing, (11)11, (2002).
- [3] SUTHAHARAN, S., *Image and Edge Detail Detection Algorithm for Object-Based Coding*, Pattern Recognition Letters, **21**, 549-557, (2000).
- [4] ALZHRANI, F. M. ve CHEN, T., *A Real- Time Edge Detector: Algorithm and VLSI Architecture*, Real-Time Imaging, **3**, 363-378, (1997).
- [5] VLIET, L. J., YOUNG, I. T. ve BECKERS, G. L., *An Edge Detection Model Based on Nonlinear Laplace Filtering*, Pattern Recognition and Artificial Intelligence, E.S. Gelsema and L. N. Kanal (eds.), Elsevier Science Publishers B. V., 63-73, (1988).
- [6] AARNINK, R. G., ROSETTE, J. M. C. H., FEITZ, Wouter F. J., DEBRUYNE, F. M. ve WIJKSTRA, H., *A Preprocessing Algorithm for Edge Detection with Multiple Scales of Resolution*, European Journal of Ultrasound, **5**, 113-126, (1997).
- [7] LEE, JSJ., HARALICK, RM. ve SHAPIRO, LG., *Morphologic Edge Detection*, IEEE T Robot Autom, 142-156, (1987).
- [8] HOU, Z. J. ve WEI, G. W., *A New Approach to Edge Detection*, Pattern Recognition, **35**, 1559-1570, (2002).
- [9] CAMPILHO, A., *Edge detection*, System Design Engineering, SD776 Image Processing Applications, (1999).
- [10] ZIOU, D. ve TABBONE, S., *Edge Detection Techniques - An Overview*, Technical Report, No. 195, Dept. Math & Informatique, Universit de Sherbrooke, (1997).
- [11] ŞENEL, H. G., Peters, R. A. ve Dawant, Benoit, *Topological Median Filters*, IEEE Transactions on Image Processing, **11**, 2, (2002).
- [12] PAVLIDIS T., *Why Progress in Machine Vision is so Low*, Pattern Recognition Letters **13**, 221-225, (1992) .

- [13] KITCHEN, L. J. ve ROSENFELD, A., *Edge Evaluation Using Local Edge Coherence*, IEEE Trans. System Man and Cybernetic, (1981).
- [14] DAVIES, E. R., *Machine Vision*, Academic Press, (1997).
- [15] ZIOU, D. ve KOUKAM, A., *Knowledge-Based Assistant for the Selection of Edge Detectors*, Pattern recognition, Published by Elsevier Press, 15(31), 587-596, (1998).
- [16] SALOTTI, M., BELLET, F. ve GARBAY, C., *Evaluation Of Edge Detectors: Critics And Proposal*, Workshop on Performance Characteristics of Vision Algorithms, Cambridge, (1996).
- [17] PREWITT, J., *Object Enhancemet And Extraction*, Picture Processing and Psychopictorics (B. Lipkin ve A. Rosenfeld, editör), NY, Academic Pres, (1970).
- [18] SOBEL, I., *Camera Models and Perception*, Ph.D. tezi, Stanford University, Stanford, CA, (1970).
- [19] CANNY, J., *A Computational Approach to Edge Detection*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 8, 679-700, (1986).
- [20] HILDRETH, E. C., *Edge Detection*, A. I. Memo No. 858, Artificial Intelligence Laboratory, Massachusetts Institute of Tecnology, September, (1985).
- [21] BASU, M., *Gaussian - Based Edge - Detection Methods – A Survey*, IEEE Transactions on Systems, Man, Cybernetics – Part C: Applications and Reviews, 32 (3), (2002).
- [22] SHIGERU, A., *Consistent Gradient Operators*, IEEE Transacactions on Pattern Analysis and Machine Intelligence, 22 (3), (2000).
- [23] WEI, G. W. ve JIA, Y. Q., *Synchronization – Based Image Edge Detection*, Europhysics Letters, 59 (6), 814-819, (2002).
- [24] GONZALEZ, R. ve WINTZ, P., *Digital Image Processing*, Addison-Wesley, (1987).
- [25] ROBERTS, L. G., *Machine Perception of Three-Dimensional Solids*, in optical and Electro-Optical Information Processing (J. Tippett, ed.), 159-197, MIT Pres, (1965).
- [26] MCCANE, B., *Edge Detection*, Cosc 453: Computer Vision Notes, Department of Computer Science, University of Otago, (2001).

- [27] SOBEL, I., *An Isotropic 3×3 Gradient Operator*, Machine Vision for Three – Dimensional Scenes, Freeman, H., Academic Pres, NY, 376-379, (1990).
- [28] DUDA, R. O. ve HART, P. E., *Pattern Classification and Scene Analysis*, John Wiley and Sons, NY, 271-273, (1973).
- [29] WILSON, J. N., RITTER, G. X. ve RITTER, G. X., *Handbook Of Computer Vision Algorithms in Image Algebra*, 85-136, (2000).
- [30] TORRE, V. ve POGGIO, T., On Edge Detection, Massachusetts Institute of Tecnology, (1984).
- [31] UDUPA, J. K. ve SAMARASEKERA, S., *Fuzzy Connectedness and Object Definition: Theory, Algorithms and Applications in Image Segmentation*, Graphical Models and Image Processing, 58 (3), 246-261, (1996).
- [32] ROSENFELD, A., *Fuzzy Digital Topology*, Inform. Control, **40**, 76-87, (1979).
- [33] ROSENFELD, A., *Fuzzy Geometry of Image Subsets*, Pattern Recognition Lett. **2**, 311-317, (1984).
- [34] UDUPA, J. K. ve SAMARASEKERA, *Fuzzy Connectedness and Object Definition*, In SPIE Proceedings Medical Imaging, **2431**, 2-10, (1995).
- [35] YAO, L. ve JIE, T., *Medical Image Segmentation via Fuzzy Connectedness-Basd Fuzzy C-Means Method*, Proceedings of 6th International Conference on Pattern Recognition and Information Processing (PRIP2001), **1**, 257-263, (2001).
- [36] ZADEH, L. A., *Fuzzy Sets*, Inform. Control, **8**, 338-353, (1965).
- [37] UDUPA, J. K., *Multidimensional Digital Boundaries*, CVGIP: Graphical Models and Image Processing, (4)56, 311-323, (1994).
- [38] ECKHARDT, U. ve LATECKI, L., *Digital Topology*, In Current Topics in Pattern Recognition Research Trends, Council of Scientific Information, (1994).
- [39] GONZALEZ, J. M., GUIL, N. ve ZAPATA, E. L., *A Multiresolution Generalized Hough Transform*, VIII, Simposium Nacional de Reconocimiento de Formas Analysis de Imagenes (SNRFAI), (1999).