

**T.C.**  
**KASTAMONU ÜNİVERSİTESİ**  
**FEN BİLİMLERİ ENSTİTÜSÜ**  
**MALZEME BİLİMİ VE MÜHENDİSLİĞİ ANA BİLİM DALI**



**YAPAY SİNİR AĞLARINI KULLANARAK SAYISAL  
GÖRÜNTÜLERDE KENAR TESPİTİ İÇİN OPTİMUM BİR  
YAKLAŞIM**

**YOSSF AHMED AHMED GHIT**

**YÜKSEK LİSANS TEZİ**

**DR. ÖĞR. ÜYESİ YASEMİN GÜLTEPE**

**OCAK - 2022**  
**KASTAMONU**

## TEZ ONAYI

**Yossf Ahmed Ahmed GHIT** tarafından hazırlanan “**YAPAY SİNİR AĞLARINI KULLANARAK SAYISAL GÖRÜNTÜLERDE KENAR TESPİTİ İÇİN OPTİMUM BİR YAKLAŞIM**” adlı tez çalışmasının savunma sınavı **21.01.2022** tarihinde yapılmış olup aşağıda verilen jüri tarafından oy birliği ile Kastamonu Üniversitesi Fen Bilimleri Enstitüsü **Malzeme Bilimi ve Mühendisliği Ana Bilim Dalı Yüksek Lisans Tezi** olarak kabul edilmiştir.

|                   |                                                         |       |
|-------------------|---------------------------------------------------------|-------|
| <b>Danışman</b>   | Dr. Öğr. Üyesi Yasemin GÜLTEPE<br>Kocaeli Üniversitesi  | ..... |
| <b>Jüri Üyesi</b> | Doç. Dr. Can Doğan VURDU<br>Kastamonu Üniversitesi      | ..... |
| <b>Jüri Üyesi</b> | Dr. Öğr. Üyesi Ayhan AKBAŞ<br>Abdullah Gül Üniversitesi | ..... |

Jüri üyeleri tarafından kabul edilmiş olan bu tez Kastamonu Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulunca onanmıştır.

|                  |                       |       |
|------------------|-----------------------|-------|
| Enstitüsü Müdürü | Prof. Dr. İzzet ŞENER | ..... |
|------------------|-----------------------|-------|

## TAAHHÜTNAME

*Bu tezin tasarımı, hazırlanması, yürütülmesi, araştırmalarının yapılması ve bulgularının analizlerinde bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu; ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını, bilimsel etiğe uygun olarak kaynak gösterildiğini bildirir ve taahhüt ederim.*

**Yossf Ahmed Ahmed GHIT**

## ÖZET

### YÜKSEK LİSANS TEZİ

#### YAPAY SİNİR AĞLARINI KULLANARAK SAYISAL GÖRÜNTÜLERDE KENAR TESPİTİ İÇİN OPTİMUM BİR YAKLAŞIM

YOSSF AHMED AHMED GHIT

KASTAMONU ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ  
MALZEME BİLİMİ VE MÜHENDİSLİĞİ ANA BİLİM DALI  
DANIŞMAN: DR. ÖĞR. ÜYESİ YASEMİN GÜLTEPE

Kenar algılama, dijital bir görüntüde, süreksizliklerin olduğu yerlerde görüntü parlaklığının keskin veya daha resmi olarak değiştiği noktaları belirlemeyi amaçlayan çeşitli yöntemlerdir. Kenar algılama, görüntü tanıma, özellik çıkarma ve bilgisayarla görme gibi farklı görünüş işleme işlemlerindeki en temel görevlerden biridir. Çeşitli matematiksel yöntemler içerir ve görüntü parlaklığının keskin bir şekilde değiştiği noktalar tipik olarak “kenarlar” adı verilen bir dizi eğri çizgi parçası halinde düzenlenir. Bu tezde, yapay sinir ağları kullanılarak sayısal görüntülerin kenarı araştırılmıştır. Dijital görüntülerin kenarlarını bulmak için çeşitli yöntemler ve algoritmalar vardır. Son yıllarda oldukça ilgi gören bu yöntemlerden biri de yapay sinir ağlarının kullanılmasıdır. Burada, klasik Sobel yönteminden elde edilen kenar bulma görüntüsü, yapay sinir ağı için bir öğretmen olarak kullanılır. Deneysel sonuçlar, sinir ağlarının kullanımının klasik yöntemlerle elde edilen görüntü kenarlarında önemli bir gelişme sağladığını göstermektedir.

**ANAHTAR KELİMELER:**Yapay sinir ağı, dijital görüntü kenar algılama, görüntü işleme

Ocak 2022, 60 Sayfa

## **ABSTRACT**

### **MSC THESIS**

#### **AN OPTIMUM APPROACH FOR DIGITAL IMAGE EDGE DETECTION WITH ARTIFICIAL NEURAL NETWORK**

**YOSSF AHMED AHMED GHIT**

**KASTAMONU UNIVERSITY INSTITUTE OF SCIENCE  
DEPARTMENT OF MATERIALS SCIENCE AND ENGINEERING  
SUPERVISOR: ASSIST. PROF. DR. YASEMİN GÜLTEPE**

Edge detection is a variety of methods aimed at identifying points in a digital image where image brightness changes sharply or more formally where discontinuities exist. Edge detection is one of the most fundamental tasks in different image processing operations such as pattern recognition, feature extraction and computer vision. It involves various mathematical methods, and the points where the image brightness changes sharply are typically arranged into a series of curved line segments called “edges”. In this thesis, the edge of digital images is investigated using artificial neural networks. There are various methods and algorithms for finding the edges of digital images. One of these methods, which has attracted a lot of attention in recent years, is the use of artificial neural networks. Here, the edge finding image obtained from the classical Sobel method is used as a teacher for the neural network. Experimental results show that the use of neural networks provides a significant improvement in image edges obtained by classical methods.

**KEYWORDS:**Artificial neural network, digital image edge detection, image processing

January 2022, 60 Page

## TEŞEKKÜR

Bana tezimi tamamlama gücü ve sabrı veren Yüce Allah'a en içten şükranlarımı sunuyorum. Her şeyden önce, Kastamonu Üniversitesindeki yüksek lisans eğitimim sırasında değerli danışmanım sayın Dr. Öğr. Üyesi Yasemin GÜLTEPE'ye, bu tez boyunca verdiği tavsiyeler, teşvikler, rehberlik ve sürekli geri bildirimler için teşekkür etmek isterim.

Tez izleme toplantılarında jüri üyesi olarak görev alan Doç.Dr. Can Doğan VURDU'ya ve Dr.Öğr.Üyesi Ayhan AKBAŞ'a değerli önerileri ve çalışmaya olan katkıları için teşekkürlerimi sunarım.

Hayatım boyunca desteklerini esirgemeyen aileme sonsuz teşekkürlerimi sunarım. Onların desteği olmadan eğitim kariyerimi sorunsuz bir şekilde tamamlayamazdım.

Kastamonu Üniversitesinde aldığım eğitim için yaptıkları destekten dolayı Libya Hükümetine ve Türkiye'deki Libya Büyükelçiliğine teşekkür ederim.

Yossf Ahmed Ahmed GHIT

Kastamonu, 2022

## İÇİNDEKİLER

|                                                                   | <u>Sayfa</u> |
|-------------------------------------------------------------------|--------------|
| <b>TEZ ONAYI .....</b>                                            | <b>ii</b>    |
| <b>TAAHHÜTNAME .....</b>                                          | <b>iii</b>   |
| <b>ÖZET.....</b>                                                  | <b>iv</b>    |
| <b>ABSTRACT .....</b>                                             | <b>v</b>     |
| <b>TEŞEKKÜR .....</b>                                             | <b>vi</b>    |
| <b>İÇİNDEKİLER .....</b>                                          | <b>vii</b>   |
| <b>ŞEKİLLER DİZİNİ .....</b>                                      | <b>viii</b>  |
| <b>SİMGELER VE KISALTMALAR DİZİNİ .....</b>                       | <b>ix</b>    |
| <b>1. GİRİŞ.....</b>                                              | <b>1</b>     |
| <b>2. KENAR ALGILAMA YÖNTEMLERİ .....</b>                         | <b>5</b>     |
| 2.1 Tek Tip Fourier Katsayılarından Konsantrasyon Faktörleri..... | 5            |
| 2.2 Kenar Algılama Algoritma Bileşenleri .....                    | 9            |
| 2.3 Gürültü Filtreleme.....                                       | 12           |
| 2.3.1 Medyan Filtre.....                                          | 12           |
| 2.3.2 Ortalama Filtresi .....                                     | 13           |
| 2.3.3 Gauss Filtresi .....                                        | 13           |
| 2.4 Kenar Algılama Operatörü ile Maskeleme .....                  | 14           |
| 2.4.1 Canny Kenar Algılama Maskesi .....                          | 15           |
| 2.4.2 Robert Cross Kenar Algılama Maskesi.....                    | 16           |
| 2.4.3 Prewitt Operatör Maskesi .....                              | 17           |
| <b>3. YAPAY SİNİR AĞLARI .....</b>                                | <b>19</b>    |
| 3.1 Yapay Sinir Ağlarının Sınıflandırılması .....                 | 22           |
| 3.1.1 İleri Beslemeli Yapay Sinir Ağları .....                    | 22           |
| 3.1.2 Geri Beslemeli Yapay Sinir Ağları .....                     | 23           |
| 3.2 Derin Sinir Ağları .....                                      | 23           |
| 3.3 Yapay Sinir Ağları Eğitimi .....                              | 24           |
| <b>4. SİMÜLASYON SONUÇLARI.....</b>                               | <b>27</b>    |
| 4.1 Metot.....                                                    | 27           |
| 4.2 Simülasyon Sonuçları .....                                    | 30           |
| <b>5. SONUÇ VE ÖNERİLER .....</b>                                 | <b>38</b>    |
| <b>KAYNAKLAR .....</b>                                            | <b>40</b>    |
| <b>EKLER.....</b>                                                 | <b>46</b>    |
| EK A Diğer Görüntüler İçin Elde Edilen Sonuçlar.....              | 47           |
| <b>ÖZGEÇMİŞ.....</b>                                              | <b>60</b>    |

## ŞEKİLLER DİZİNİ

### Sayfa

|                                                                                                                                        |    |
|----------------------------------------------------------------------------------------------------------------------------------------|----|
| Şekil 2.1 Düzleştirilmiş fonksiyonlar (a: Test işlevi, b: Atlatma işlevi).....                                                         | 6  |
| Şekil 2.2 (a) 4-komşuluk ilişkili, (b) 8-komşuluk.....                                                                                 | 13 |
| Şekil 2.3 Canny kenar algılama maskeleme gösterimi .....                                                                               | 16 |
| Şekil 2.4 Roberts Cross kenar algılama maskesi .....                                                                                   | 17 |
| Şekil 2.5 Prewitt operatör maskesi.....                                                                                                | 17 |
| Şekil 3.1 Biyolojik sinir hücresi.....                                                                                                 | 20 |
| Şekil 3.2 Yapay sinir hücresi .....                                                                                                    | 20 |
| Şekil 3.3 Nöronlar için aktivasyon fonksiyonları .....                                                                                 | 22 |
| Şekil 3.4 Sinir ağı modelleri .....                                                                                                    | 24 |
| Şekil 4.1 Görüntü kenarı algılama için önerilen yöntemin akış şeması.....                                                              | 28 |
| Şekil 4.2 Örnek görüntüler .....                                                                                                       | 29 |
| Şekil 4.3 Sistem için tasarlanan YSA modeli .....                                                                                      | 31 |
| Şekil 4.4 Katmanlar ve nöronlar .....                                                                                                  | 31 |
| Şekil 4.5 Sinir ağı araçları.....                                                                                                      | 32 |
| Şekil 4.6 Sinir ağı eğitim performansı .....                                                                                           | 33 |
| Şekil 4.7 Epochlara karşı sinir ağı eğitim durumu .....                                                                                | 33 |
| Şekil 4.8 Eğitim regresyon grafiği .....                                                                                               | 34 |
| Şekil 4.9 a) Kameraman orijinal görüntüsü, b) Keyfi eşikleme ile Sobel<br>kenar algılama, c) Kenar algılama yöntemi birleşik YSA ..... | 35 |
| Şekil 4.10 a) Göl orijinal görüntüsü, b) Sobel kenar algılama, c) Kenar<br>algılama yöntemi birleşik YSA .....                         | 35 |
| Şekil 4.11 a) Julie orijinal görüntüsü, b) Sobel kenar algılama, c) Kenar<br>algılama yöntemi birleşik YSA .....                       | 36 |
| Şekil 4.12 a) Biber orijinal görüntüsü, b) Sobel kenar algılama, c) Kenar<br>algılama yöntemi birleşik YSA .....                       | 36 |
| Şekil 4.13 a) Lenna orijinal görüntüsü, b) Sobel kenar algılama, c) Kenar<br>algılama yöntemi birleşik YSA .....                       | 37 |



## SİMGELER VE KISALTMALAR DİZİNİ

### Simgeler

|                            |                                |
|----------------------------|--------------------------------|
| <b>b</b>                   | : Eşik Değeri                  |
| <b>F</b>                   | : Fonksiyon                    |
| <b>R</b>                   | : Gerçek Sayılar Kümesi        |
| <b><math>\sigma</math></b> | : Gauss Standart Sapması       |
| <b><math>\nabla</math></b> | : Gradyan Operatörü            |
| <b><math>\xi</math></b>    | : Kenar Eğrisinin Tanjant Yönü |
| <b>T(x)</b>                | : Toplam Fonksiyonu            |
| <b>w<sub>n</sub></b>       | : Ağırlık                      |

### Kısaltmalar

|             |                                          |
|-------------|------------------------------------------|
| <b>BP</b>   | : Geri Yayılım                           |
| <b>CMAC</b> | : Serebellar Model Artikülasyon Kontrolü |
| <b>GMDH</b> | : Grup Yöntemiyle Bilgi İşleme           |
| <b>LVQ</b>  | : Öğrenme Vektörü Niceleme               |
| <b>MLP</b>  | : Çok Katmanlı Algılayıcı                |
| <b>MRG</b>  | : Manyetik Rezonans Görüntüleme          |
| <b>MSE</b>  | : Ortalama Karesel Hata                  |
| <b>NaN</b>  | : Sayı Değil                             |
| <b>RBF</b>  | : Radyal Temel Fonksiyon                 |
| <b>ReLU</b> | : Rektifiye Doğrusal Birim               |
| <b>TanH</b> | : Hiperbolik Tanjant                     |
| <b>YSA</b>  | : Yapay Sinir Ağları                     |
| <b>YZ</b>   | : Yapay Zeka                             |

## 1. GİRİŞ

Görüntü işleme, gelişmiş bir görüntü elde etmek veya ondan bazı yararlı bilgiler çıkarmak için bir görüntü üzerinde bazı işlemleri gerçekleştirme yöntemidir. Girişin bir görüntü olduğu ve çıkışın görüntü veya o görüntüyle ilişkili özellikler olabileceği bir sinyal işleme türüdür. Günümüzde görüntü işleme hızla gelişen teknolojiler arasında yer almaktadır. Mühendislik ve bilgisayar bilimleri disiplinlerinde de temel araştırma alanını oluşturur (Balq, 2019).

Özellik çıkarma, örüntü tanıma, bölütleme, görüntü morfolojisi vb. gibi dijital görüntü işlemleri için yaygın olarak kullanılan bir teknolojidir (Xin vd., 2012). Kenar algılama, görüntü işleme içinde kendi başına iyi gelişmiş bir alandır. Kenar algılama, temel olarak görüntü bölütleme tekniği olup, görüntünün tanımlandığı uzamsal alanı anlamlı parçalara veya bölgelere ayırır. Kenarlar, bir görüntüdeki bölgeler arasındaki sınırları tanımlar, bu da segmentasyona ve nesne tanımaya yardımcı olur. Bir görüntüde gölgelerin nereye düştüğünü veya bir görüntünün yoğunluğundaki diğer herhangi bir belirgin değişikliği gösterebilirler. Kenar algılama, görüntü işlemede yaygın olarak kullanılan bir işlemdir ve görüntüdeki geometrik şekilleri tanımlamada ve ayırmada çok önemli bir rol oynar (Muthukrishnan, 2011). Kenar algılamanın kalitesi büyük ölçüde aydınlatma koşullarına, benzer yoğunluktaki nesnelerin varlığına, sahnedeki kenarların yoğunluğuna ve gürültüye bağlıdır (Nadernejad vd., 2008).

Kenar algılama, örüntü tanıma, özellik çıkarma ve bilgisayarla görme gibi farklı görüntü işleme işlemlerinde önemli bir aşamadır. Kenar algılama, bir görüntüdeki kenarları tanımlamayı amaçlayan algoritmalara atıfta bulunan görüntü işlemede bir terminolojidir. Kenarlar, görüntüdeki en önemli bilgileri içerir ve nesnenin konumu hakkında bilgi sağlayabilir (Canny, 1986). Kenar, bir nesne ile arka plan arasındaki sınırdır. Kenarlar, görüntüde önemli bilgiler içerir ve kenar algılama, görüntü işlemede düşük seviyeli bir işlem olarak kabul edilebilir. Bu amaçla geliştirilen farklı yöntemler arasında geleneksel yöntemler basit ve oldukça verimlidir. Görüntülerdeki kenarların şekli birçok parametreye bağlıdır. Örnek olarak, nesnenin geometrik ve optik özellikleri, aydınlatma koşulları ve görüntülerdeki gürültü seviyesi verilebilir.

Görüntü parlaklığındaki keskin değişiklikleri algılamanın amacı önemli olayları yakalamaktır. Kenar algılama, görüntüler içindeki nesnelerin sınırlarını bulmaya yönelik bir görüntü işleme tekniğidir. Parlaklıktaki süreksizlikleri algılayarak çalışır (Maini ve Aggarwal, 2009). Görüntü kalitesi, çıktı kenarında önemli bilgileri yansıtır ve görüntünün boyutu küçülür. Kenar algılamada bilgisayar belleğinde yüksek hacimli görüntü alanı kapsama sorununu çözmenin yollarından biri olarak kabul edilebilir. Kenarlar tespit edildiğinde depolama, internet üzerinden iletim ve bant genişliği sorunları çözülebilir (Barrow ve Tenenbaum, 1981; Lindeberg, 1998; Vincent, 2007). Kenarlar genellikle nesne sınırlarını temsil eden görüntü konumlarında meydana geldiğinden, görüntüler farklı nesnelere karşılık gelen alanlara bölündüğünde kenar algılama, görüntü bölütlemeye yaygın olarak kullanılır.

Kenar algılama, görüntü işleme, görüntü analizi, görüntü örüntü tanıma ve bilgisayarla görme tekniklerindeki temel adımlardan biridir. Literatürde birçok iyi bilinen kenar algılama algoritmaları önerilmiştir (Rosenfeld ve Kak, 1982). Bazı kenar algılama algoritmaları, tanınan nesnenin türüne bağlı olarak diğerlerinden daha iyi performans gösterir (Canny, 1986). Kullanılan farklılaşma operatörüne dayalı olarak birkaç sınıf kenar algılama algoritması mevcuttur (Senthilkumaran ve Rajesh, 2008). Kenar algılama algoritmaları genel olarak aşağıdaki kategorilere ayrılabilir (Chidiac ve Ziou, 1999):

- ✓ Gradyan tabanlı kenar algılayıcıları,
- ✓ Laplacian kenar algılayıcıları,
- ✓ Gauss kenar algılayıcıları,
- ✓ Renkli kenar algılayıcıları

Gradyan tabanlı kenar algılama algoritmaları, yönlü birinci türev işlemini kullanır. Gradyan tabanlı kenar algılama sistemlerinin avantajları basit olmaları ve çeşitli yönlerde kenarları algılayabilmeleridir. Gürültüye duyarlılıkları, dezavantaj olarak söylenebilir.

Sobel, Roberts, Prewitt ve Kirsch gibi iyi bilinen birkaç gradyan tabanlı kenar algılama algoritması mevcuttur (Argyle, 1971). Bu kenar algılama algoritmalarındaki temel fark, kullanılan evrişim çekirdeğinin türüdür.

Sobel kenar algılaması, görüntü yoğunluğu fonksiyonunun gradyanının bir yaklaşımını hesaplayan ayrı bir farklılaşma operatörü kullanır. Bir görüntü üzerinde 2 boyutlu uzaysal gradyan ölçümü gerçekleştirir ve kenarlara karşılık gelen yüksek uzaysal frekans bölgelerini vurgular. Gri tonlamalı görüntüde her noktadaki yaklaşık gradyan değerini bulmak için kullanılır. Gürültüyü bastırmak için ağırlıklar, merkezde daha yoğun olacak şekilde atanır (Sudarshan vd., 2011).

Birçok kenar algılama filtresi yalnızca belirli yönlerdeki kenarları algılar, bu nedenle farklı yönlerdeki kenarları algılayan filtre kombinasyonları, genellikle tüm kenarları algılayan kenar algılayıcıları elde etmek için kullanılır (Zwaag vd., 2002). Roberts Edge filtresi, sırayla yatay ve dikey bir filtre uygulayarak kenarları algılamak için kullanılır. Her iki filtre de görüntüye uygulanır ve nihai sonucu oluşturmak için toplanır.

Prewitt kenar filtresi, görüntüdeki keskin yoğunluk geçişlerine ve görüntüdeki düşük Poisson tipi gürültü varsayımına dayanmaktadır. Bu yöntemde görüntü, farklı yönlerde kenarlara duyarlı farklı evrişim çekirdekleri seti ile kıvrılır. Her pikselde, seçilen tüm evrişim çekirdeklerinin maksimum yanıtı ile gradyan büyüklüğü. Bir pikselin kenar büyüklüğü ve yönü, pikselin yerel alanıyla en iyi eşleşen şablon tarafından belirlenir. Prewitt'in kenar algılama yöntemi, gürültülü koşullar altında daha iyi performans sergiler. Roberts, Sobel ve Prewitt arasındaki temel fark, ayrıklaştırma yaklaşımlarının farklı oluşundadır (Alkım, 2005).

Klasik kenar algılama yöntemleri yerine yapay sinir ağları oldukça yaygın bir teknoloji olarak kullanılabilir. Yapay sinir ağı (YSA) daha az işlem yükü sağlaması ve gürültünün etkisini azaltmada daha avantajlı olması nedeniyle klasik kenar algılama yöntemine göre daha çok tercih edilmektedir (Wang vd., 2000; Xueli, 1996). YSA, eğitim aşamasında birden çok girdi ve çıktı kullanılabilirliği açısından daha kullanışlıdır.

YSA, tıbbi görüntülemenin birçok bilgisayar destekli teşhisinde başarıyla kullanılmıştır (Jiang vd., 2010). Ayrıca, tıbbi görüntü bölütleme ve kenar tespiti, herhangi bir bilgisayar destekli teşhisin kenar tespiti gibi diğer teknikleri gerektireceği tüm tıbbi görüntüleme uygulamaları için önemli bir problem olmaya devam etmektedir. Ek olarak, ANN, segmentasyonun yanı sıra kenar algılamayı tamamlamak için tıbbi görüntüleri içerik tutarlı bölgelere sınıflandırmak için öğrenme yeteneğinden ve eğitim mekanizmalarından yararlanmak için kullanılmıştır.

Bu tez çalışmasında, kenar algılama için yapay sinir ağları sunulmuştur. Kenar algılama, görüntü işleme ve bilgisayarlı görme alanlarında en çok çalışılan ancak çözülmemiş problemlerden birisidir. Yüz tanıma (Zhao vd., 2000), araç plaka tanıma (Ha ve Shaberi, 2016), beyin tümörü tespiti (Sharma vd., 2012), şerit tespiti (Hines ve Watson, 1983), arazi tespiti (Yim ve Oh, 2003) ve manyetik rezonans görüntüleme (MRG) sistemleri gibi çeşitli görüntü tanıma uygulamalarında son yıllarda giderek daha popüler hale gelmiştir.

## 2. KENAR ALGILAMA YÖNTEMLERİ

Görüntünün kenarları, görüntünün sınırlarını ve bölgelerini karakterize eder. Görüntülerdeki kenarlar, güçlü yoğunluk kontrastlarına sahip alanlardır. Kenar algılama, görüntü işleme ve görüntü analizindeki en önemli görevlerden biridir (Savant, 2014). Kenar algılama işlemi, bir görüntüdeki önemli yapısal özellikleri korurken veri miktarını önemli ölçüde azaltır ve gereksiz bilgileri filtreler.

Kenar algılamayı gerçekleştirmenin birçok yolu vardır. Ancak gradyan ve Laplacian olmak üzere iki kategoriye ayrılabilir. Gradyan yöntemi, görüntünün birinci türevinde maksimum ve minimuma bakarak kenarları tespit eder. Laplacian yöntemi, kenarları bulmak için görüntünün ikinci türevinde sıfır geçişleri arar (Shrivakshan ve Chandrasekar, 2012).

Fourier dönüşümü, dijital sinyal işlemede önemli bir yoldur. Görüntü analizinde de kullanılır. İki boyutlu görüntüler, uzaydaki iki boyutlu sinyal olarak kabul edilebilir, bu nedenle frekans spektrumunu elde etmek için fourier dönüşümü kullanılır. Fourier dönüşümü ile sayısal görüntü işlemede kullanılan yöntem genel olarak uzaysal alan ve frekans alanı yöntemleri olmak üzere ikiye ayrılabilir. Uzamsal alan yönteminde işleme, bir görüntünün piksel değeri doğrudan manipüle edilerek gerçekleştirilir, frekans alanı yönteminde ise dijital görüntü bilgisi önce Fourier dönüşümü yoluyla dönüştürülür ve ardından Fourier dönüşüm sonucu değiştirilir. Manipülasyon tamamlandıktan sonra, görüntü bilgisini tekrar elde etmek için ters Fourier dönüşümü gerçekleştirilir. Frekans alanı yöntemi, uzamsal alan yöntemi kullanılarak çözülmesi zor olan belirli sorunları çözmek için kullanılabilir (Gonzalez ve Richard, 1993; Robert, 1989).

### 2.1 Tek Tip Fourier Katsayılarından Konsantrasyon Faktörleri

Bu bölümde, verilen sonlu sayıda düzgün Fourier katsayılarından parçalı düzgün bir fonksiyonun kenarlarını (atlama süreksizliklerini) saptamak için konsantrasyon faktörü yöntemi hakkında bilgi verilmiştir. Aşağıdaki koşulları karşılayan temel bir  $f: \mathbb{R} \rightarrow \mathbb{R}$  fonksiyonunu ele alınır (Gelb ve Song, 2014):

(i)  $[0, 1]$  üzerinde desteklenir.

(ii)  $(0, 1)$ 'de sonlu sayıda sıçrama süreksizliği var.

Aşağıdaki biçimde  $f$ 'nin sonlu sayıda Fourier katsayısı veriliyor.

$$\hat{f}(j) = \langle f, \phi_j \rangle, \quad -m \leq j \leq m \quad (2.1)$$

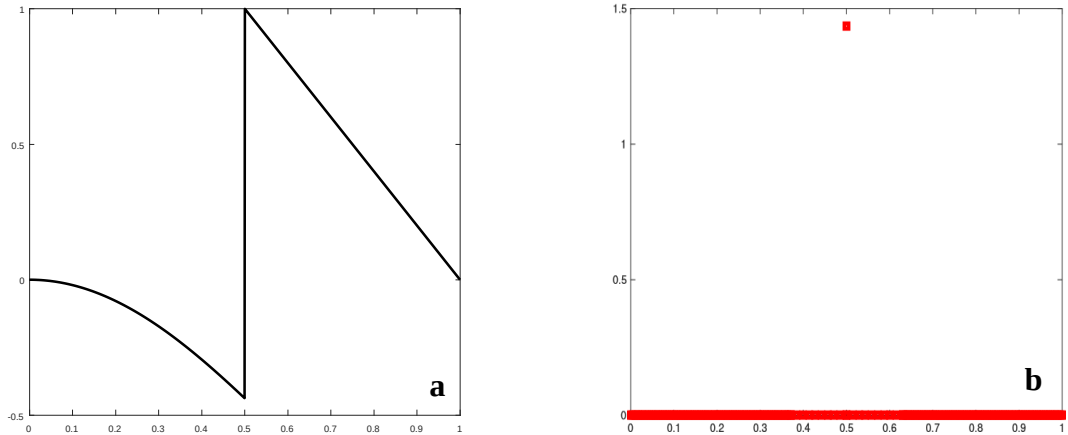
Burada;  $\phi_j(x) = e^{2\pi i j x}$ ,  $x \in [0,1]$  ve  $\langle \cdot, \cdot \rangle$ 'i iç çarpımı  $L^2[0,1]$ 'deki normal iç çarpımdır.

Altta yatan parçalı düzgün  $f$  fonksiyonunun atlama süreksizliklerinin (kenarlarının) konumunu ve  $f$ 'nin Fourier verilerinden buna karşılık gelen atlama değeri belirlenir.

Daha sonra kenar algılama problemini bir fonksiyon yaklaşımı problemine çevirebilir. Bu amaçla, karşılık gelen  $[f]: \mathbb{R} \rightarrow \mathbb{R}$  ile tanımlanan Denklem 2.2 ile bir atlama fonksiyonunu ele alınır (Gelb ve Hines, 2011).

$$[f](x) = \lim_{t \rightarrow x^+} f(t) - \lim_{t \rightarrow x^-} f(t) = f(x^+) - f(x^-), \quad x \in [0, 1] \quad (2.2)$$

$[f]$  atlama fonksiyonunun iyi tanımlı olması önemlidir, çünkü  $f$  fonksiyonu parçalı süreklidir ve  $[f]$ ,  $[f]$ 'nin atlama değerini aldığı atlama süreksizlikleri dışında her yerde sıfırdır.



Şekil 2.1 Düzleştirilmiş fonksiyonlar (a: Test işlevi, b: Atlama işlevi)

Şekil 2.1 bize tek atlama süreksizliği ile parçalı düzgün fonksiyona karşılık gelen düzleştirilmiş fonksiyonunun bir örneğini vermektedir.

Altta yatan parçalı fonksiyonun kenarlarını (sıçrama süreksizliklerini) tespit etmenin, Denklem 2.2'deki karşılık gelen atlama fonksiyonunu kurtarmaya eşdeğer olduğuna işaret ediyoruz. Altta yatan parçalı düzgün  $f$  fonksiyonunun atlama süreksizliklerini belirlemek için sayısal bir algoritma oluşturmak için, Denklem 2.1'deki atlama fonksiyonunu  $[f](\cdot)$  aşağıdaki biçimde yeniden yazmak çok daha uygun olacaktır.

$f$  burada  $\chi_{\xi_j}$ , tarafından tanımlanan bir gösterge işlevidir.

$$\chi_{\xi_j}(x) = \begin{cases} 1, & \text{for } x = \xi_j \\ 0, & \text{otherwise} \end{cases} \quad (2.4)$$

Genelliği kaybetmeden, sunumun basitliği ve algoritmik gelişimin amacı için, altta yatan parçalı düzgün  $f$  fonksiyonunun  $\xi \in (0,1)$ 'de sadece tek bir kenarı (atlama süreksizliği) olduğunu varsayıyoruz.

$$[f](x) = [f](\xi)\chi_{\xi}(x), \quad x \in [0, 1] \quad (2.5)$$

Daha sonra sayısal deneylerimizde, yöntemimizin birden çok atlama süreksizliğine sahip fonksiyonlara kolayca genişletilebileceğini göstereceğiz.

Denklem 2.5'deki gösterge fonksiyonu  $\chi_{\xi}$  sadece kümesi sıfır ölçüye sahip tek bir noktada önemsiz olduğundan,  $L^2[0,1]$ 'de doğrudan bir yaklaşıklık elde etmek pratik değildir. Bunun yerine, aşağıdaki biçimde  $\chi$  gösterge fonksiyonunun  $h$  yumuşak bir yaklaşımı sunulmaktadır.

$$h_{\epsilon}(x) = h\left(\frac{x - \xi}{\epsilon}\right), \quad \epsilon > 0 \quad (2.6)$$



Burada; negatif olmayan  $h$  fonksiyonu  $(-\infty, 0)$  üzerinde artıyor ve  $(0, \infty)$  üzerinde azalıyor ve 0'da 1 tepe değeri alarak 0 etrafında ortalıyor.

$$h(0) = \max(h(x)) = 1 \quad x \in \mathbb{R} \quad (2.7)$$

Düzleştirilmiş  $h$ , parametre 0'a yakın olduğunda  $\chi_\xi$  gösterge işlevine yakın olacaktır, yani,

$$h_\epsilon(x) \rightarrow \chi_\xi(x), \text{ as } \epsilon \rightarrow 0^+ \quad (2.8)$$

Pratik amaç için, sayısal değerlendirme, düzleştirilmiş  $h$ 'nin 0 civarında çan şeklinde olmasını ve küçük olmasını gerektirecektir.

Çevremizdeki en yaygın görüntü işleme, insanın görsel sisteminde gerçekleşir. Bu sistemde ışık kaynaklarından yansıyan veya yayılabilen ışık ışınları göze ulaştığında bu sinyaller fotoreseptörler tarafından algılanır ve beyin tarafından yorumlanmaya uygun elektrik sinyallerine dönüştürülür.

Ancak sadece görünür ışık spektrumunda tanımlanan ve dalga boyu yaklaşık 400 nm ile 700 nm arasında değişen elektromanyetik dalgalar olan ışık ışınları, dönüştürülerek merkezi sinir sistemine iletilir ve bu sinyallerin görülmesi beyin tarafından yorumlanır.

Dışarıdaki siyah kağıdın görüntüsü, aslında iç mekandaki beyaz kağıdın görüntüsünden daha yoğundur. Siyah kağıt, fiziksel olarak daha yoğun olmasına rağmen neden hala siyah görünüyor? Bu fenomene parlaklık sabitliği diyoruz. Algılanan parlaklık, uyarının gerçek fiziksel yoğunluğuna eşit değildir. Bunun yerine, algılanan parlaklık, aydınlatma koşullarından bağımsız olarak yüzey yansımasına bağlıdır.

Gözlere ulaşan görsel uyarın, hem aydınlatma seviyesine (iç mekan ışığına karşı dış mekan ışığı) hem de yüzeylerin yansımasına bağlıdır. Ama aslında görsel uyarının kendisiyle ilgilenilmeyebilir. Daha ziyade, görsel uyarana yol açan dünyadaki yüzeylerin (siyah beyaz kağıdın yansımaları) fiziksel özellikleriyle ilgilenilir. Böylece görsel sistem aydınlatmayı dışlar ve tipik olarak algıladığımız tek şey kağıdın görelî

yansımasıdır (siyah beyaza karşı). Yukarıda insan görme sistemlerinin temelini oluşturan bilgilerden kısaca bahsedilmiştir. Bilgisayarla görme, insan görüşüyle pek çok benzerliğe sahiptir, ancak ikisi arasında önemli farklılıklar vardır. İnsan görüşü, hala tam olarak anlaşılamayan karmaşık bir süreçtir. Bilgisayarla görme, bilgisayarların insan görme yeteneklerine ulaşmasını sağlayan insan görüşünün teknolojik bir uygulamasıdır (Jähne, 2013).

İnsan gözü, bir kameraya çok benzeyen karmaşık bir optik sistemdir; ışık, baktığınız bir nesneden yansır ve korneadan gözlemlere girer. Daha sonra ışık, birlikte göze giren ışık miktarını kontrol eden gözbebeği ve irisden geçer. Hepsi birlikte çalıştığında, ışığı retina adı verilen gözün arkasına odaklarlar. Işık retinaya çarptığında, retinadaki küçük hücreler onu elektrik sinyallerine dönüştürür (Bovik, 2010).

Bilgisayarla görme, bilgisayarların fotoğraf ve video gibi dijital görüntülerin içeriğini görmesini ve anlamasını sağlayan bir yapay zeka biçimidir. İnsan görüşünün nesneleri algılama biçimine benzer şekilde, bilgisayarın çevrelerini algılamasına ve nesneleri tanımlamasına olanak tanır. İnsan vizyonunun önceden tanımlanmış özelliklerini toplamak için algoritmalar kullanılır ve insan vizyonunun yeteneklerini bilgisayarlara benzetmek için modeller ve programlar üretirler. Bilgisayarlara, tıpkı insan görüşünün yaptığı gibi görsel bilgileri elde etme, analiz etme ve işleme yeteneği verir.

## **2.2 Kenar Algılama Algoritma Bileşenleri**

Bahsedilen görüntü işleme uygulamalarında oldukça önemli bir yere sahip olan kenar algılama algoritmaları en çok araştırılan çalışmalar arasında yer almaktadır. 1986 yılında J. Canny tarafından tanıtılan Canny kenar algılama yönteminin tanıtılmasıyla birlikte, hız kazanan kenar bulma çalışmaları kapsamında birçok yöntem geliştirilmiştir.

Sayısal görüntülerde kenar algılama, piksel değerlerindeki keskin değişimleri, diğer bir deyişle görüntüde sürekliliğin kaybolduğu durumları belirlemek için geliştirilmiş çeşitli matematiksel yöntemleri içeren bir görüntü işleme türüdür. Görüntüdeki bu keskin değişiklikler, kenar adı verilen tipik eğri ve çizgi kümelerini oluşturur. İki boyutlu görüntülerde kenar algılama, iki boyutlu bir sinyal işleme türüdür. Aslında

görüntüdeki kenar algılama, tek boyutlu sinyallerde adım algılama olarak bilinen sürekliliği bozan noktaların bulunması veya sinyalin zaman içinde süreksizlik gösterdiği noktaların bulunması olarak tanımlanan değişiklik algılamaya benzer bir konudur. Kenar algılama; görüntü işleme, makineyle görme, bilgisayarla görme ve özellikle öznitelik algılama ve öznitelik çıkarma alanlarında temel bir araç olarak karşımıza çıkmaktadır (Umbaugh, 2010).

Görüntüde süreksizlik gösteren yerlerin bulunması, görüntü verisinin önemli yönlerini ve görüntü uzayındaki özelliklerde meydana gelen değişiklikleri tespit etmeyi amaçlar. Genel kabul bağlamında görselde süreksizliğe neden olan sebepler şu şekilde sıralanabilir (Barrow ve Tenenbaum, 1981; Xiang vd., 2017):

- ✓ Derinlemesine süreksizlikler,
- ✓ Yüzey oryantasyonunda süreksizlikler,
- ✓ Malzeme özelliklerindeki değişiklikler,
- ✓ Görüntüdeki ışık farklılıkları.

Bir görüntüye bir kenar algılama algoritması uygulamak, nesnelerin kenarlarını veya yüzey yönelimindeki süreksizlikleri gösteren birleşik eğri kümeleriyle sonuçlanacaktır. Bu şekilde kenar algılama işlemi, görüntüde bulunan önemli yapısal özellikleri korurken, amaca hizmet etmekten uzak olan önemli miktarda gereksiz veriyi filtreleyebilir. Kenar algılama aşamasında istenilen çıktının alınması, aşamalı veri işleme süreçlerinin daha kolay devam etmesine katkı sağlayacak bir adımdır. Bununla birlikte, gerçek hayatta, ortalama karmaşıklığa sahip görüntülerde bile bu tür ideal kenarları tespit etmek her zaman mümkün değildir. Literatürde yaygın olarak kullanılan birçok kenar algılama yöntemleri vardır.

Sobel operatörü, en sık kullanılan algılama yöntemlerinden biridir ve görüntü yoğunluğunun gradyanının maksimum olduğu noktalarda kenarları döndürür. Gradyan değeri bazı eşikleri aşan konumlar, bildirilen uç konumlardır. Prewitt yöntemi, türevin Prewitt yaklaşımını kullanarak kenarları bulur ve görüntü yoğunluğunun gradyanının

maksimum olduđu noktalarda kenarları döndürür. Canny kenar detektörü, standart kenar algılama metodolojisi olarak kabul edilir ve görüntü gradyanının yerel maksimumlarını arayarak kenarları bulur. Görüntü gradyanı, sınır tespitinde önemlidir, çünkü görüntüler genellikle nesneler arasındaki sınırdaki en hızlı şekilde değişir. Sınıfta, kenarları bulmak için gradyan, şu şekilde kullanılabilir: 1) görüntü gradyanının büyüklüğü yüksektir 2) Gradyan yönünde olan diğer konumlarda gradyan büyüklüğünden daha yüksektir (URL-1, 2005).

Kenar algılamada temel adımlar; filtreleme, geliştirme/netleştirme, tespit ve yerelleştirmedir. Bu dört adım, bir kenar algılama sürecinin çalışmasını oluşturur; burada, filtrelemede gürültünün azaltıldığı, keskinleştirmenin piksellerin yerel yoğunluğundaki değişime odaklandığı, gürültü olarak kabul edilecek pikseller üzerinde algılama çalışmaları ve son adım yerelleştirme, yerelleştirmenin gerekliliğinin kenar inceltme ve bağlantı olduğu durumlarda, kenarların tam konumu ile ilgilenir (Lakshmi ve Sankaranarayanan, 2010). Yoğunluktaki değişimler, bir görüntüde kenarın varlığının nedenleridir; yoğunluktaki değişime geometrik olaylar, yani yüzey, derinlik, renk ve doku süreksizlikleri ve geometrik olmayan olaylar, yani gölgeler, iç yansımalar ve aydınlatma değişiklikleri neden olur (Nayak, 2014).

Gürültüyü gidermek için sıklıkla lineer olmayan bir filtre olan medyan filtresi kullanılmaktadır (Rajasekaran ve Senthilkumar, 2014). Gürültü azaltma, örneğin bir görüntüde kenar algılama, sinyal işleme ve görüntü işleme için yumuşatıcılar gibi sonraki işlemenin sonuçlarını geliştirmek için yaygın bir ön işleme aşamasıdır. Belirli koşullar altında, bu filtre gürültüyü ortadan kaldırırken kenarları koruduğu için dijital görüntü işlemede yaygın olarak kullanılır. Bu filtrenin temel fikri, herhangi bir girişi komşu girişlerin medyanı ile değiştirerek giriş yoluyla sinyal girişi yoluyla çalışmaktır.

Medyan filtresini uyguladıktan sonra, önerilen yöntem olağan adımları takip eder: Görüntü yumuşatma ve filtreleme, gradyan büyüklüğünü ve gradyan yönünü bulma, maksimum olmayan bastırma ve histerezis ile kenar izleme (Abo-Zahhad vd., 2014). Bir görüntüdeki belirli bir noktanın piksel değeri, komşuluğundaki ortalama bir değerle değiştirilir; bu, yalnızca gürültü penetrasyonunu azaltmakla kalmaz, aynı şekilde yalıtılmış noktayı da ortadan kaldırabilir.

Uygulama alanına göre inceltme basamağının kullanılıp kullanılmaması tercih edilebilir. Ancak öznelitek tespiti veya çıkarımı gibi şablon eşleştirme işlemlerinin yapıldığı uygulama alanlarında, işlem kolaylığı ve daha iyi doğruluk sağladığı için genellikle iyileştirme aşaması tercih edilir. Bu aşamada, kenar tek piksel kalınlığa inceltilirken şablonun kalınlığı belirlenir.

## **2.3 Gürültü Filtreleme**

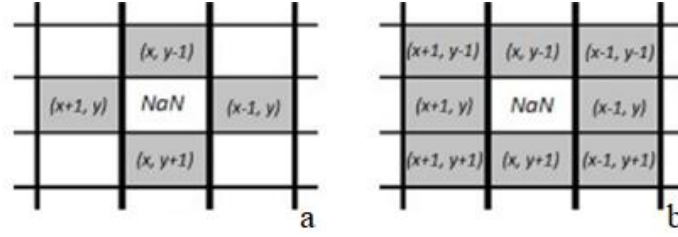
Görüntüyü etkileyebilecek farklı gürültü türleri şunlardır: Benek gürültüsü, eklemeli gürültü, Gauss gürültüsü vb. görüntülerden gürültüyü kaldırırken göz önünde bulundurulması gereken en önemli husus, görüntülerin faydalı bilgilerinin veya kenarlarının tahrif edilmemesi gerektiğidir. Gürültü giderme tekniği öyle olmalıdır ki, sadece gürültülü bilgiyi kaldırarak ve böylece görüntünün gerekli bilgisini tutacaktır.

### **2.3.1 Medyan Filtre**

Medyan filtre, darbe gürültüsünü, yani tuz-ve-biber gürültüsünü azaltma yeteneği nedeniyle en popüler olandır (Shah vd., 2020). Bir görüntünün bir noktasında medyan filtrelemesi yapabilmek için önce komşuluktaki piksellerin değerlerini sıralar, medyanı belirler ve ardından bu değeri filtrelenmiş görüntüdeki karşılık gelen piksele atarız.

Sayısal görüntülerde medyan filtresini kullanmanın temel tekniği, her piksel ve ona eşit uzaklıktaki komşu piksellerin değerleri için bu pikselin bir alt kümesini oluşturmak ve bu kümedeki medyan değeri büyüklük sırasına göre belirlemektir (Arce, 2005; Arias-Castro ve Donoho, 2009; Huang vd., 1979). Medyan filtreleme ile yeni değer belirleme sürecini açıklamak için, bir  $I(x, y)$  pikseli ve bu pikselin sekiz komşu pikseli aşağıda tartışılmıştır (Şekil 2.2).

Şekil 2.2,  $I$  görüntüsündeki  $I(x, y)$  pikselinin ve birincil yakınlıktaki komşu piksellerin gri seviye değerlerini göstermektedir.



Şekil 2.2 (a) 4-komşuluk ilişkili, (b) 8-komşuluk ilişkili

İnterpolasyon işlemi, 0 ve 3 olarak etiketlenen piksellerin komşuluğuna bağlıdır. P=NaN (Sayı değil) olarak yeniden tanımlanan bu piksellerin, yatay yönde iki komşu ve dikey yönde iki komşu olmak üzere dört komşusu vardır.  $(x+1, y)$ ,  $(x-1, y)$ ,  $(x, y+1)$  ve  $(x, y-1)$  şeklindedir. Bunların dışında  $(x-1, y-1)$ ,  $(x-1, y+1)$ ,  $(x+1, y-1)$  ve  $(x+1, y+1)$  olmak üzere dört köşegen komşuluk daha vardır. Bu komşuluklar Şekil 2.2 (a) ve (b) ile gösterilmiştir.

### 2.3.2 Ortalama Filtresi

Ortalama filtre, görüntüyü yumuşatmak ve görüntüdeki gürültüyü azaltmak veya ortadan kaldırmak için kullanılan kayan pencere tabanlı bir uzamsal filtreleme yöntemidir. Ortalama filtre yönteminde, komşu piksel değerlerinin ortalaması alınarak penceredeki merkez pikselin değeri bulunur. Pencere boyutu, ortalaması alınacak komşu piksellerin sayısını belirler. Denklem 2.9’de verilen matematiksel denklem, pencereyi komşu pikseller üzerinde kaydırarak görüntünün herhangi bir  $(i, j)$  noktasındaki uzaysal filtreleme yanıtı için kullanılır.

$$M[i, j] = \frac{1}{M} \sum_{k=i-1}^{i+1} \sum_{l=j-1}^{j+1} f[k, l] \quad (2.9)$$

Denklemdeki  $M$  değeri hesaplamada kullanılan piksel sayısını,  $k$  ve  $l$  değerleri ise bu piksellerin konumunu temsil etmektedir (Gupta ve Shailendra, 2013).

### 2.3.3 Gauss Filtresi

Gauss yumuşatma operatörü, görüntüleri “bulanıklaştırmak” ve ayrıntı ile paraziti ortadan kaldırmak için kullanılan 2 boyutlu bir evrişim operatörüdür. Bu anlamda,

ortalama filtreye benzer, ancak bir Gauss (“çan şekilli”) tümseğinin şeklini temsil eden farklı bir çekirdek kullanır (Hall, 2007). Gauss fonksiyonu şu şekilde ifade edilir:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (2.10)$$

Denklem 2.10’da  $x$  ve  $y$  kartezyen düzlem koordinatlarıdır,  $\sigma$  ise standart sapmayı gösterir. Gauss filtresindeki  $\sigma$  terimine yumuşatma ölçeği denir. Gauss filtresinin yanıtı üzerinde ölçeğin önemli bir etkisi vardır.  $\sigma$  değeri ne kadar büyük olursa, görüntü bulanıklaşır ve gürültüye duyarlılık azalır.

## 2.4 Kenar Algılama Operatörü ile Maskeleme

Bu bölüm, kenar algılamanın en temel ve işlemine adını veren kenar algılamanın doğrudan gerçekleştirildiği aşama ile ilgilidir. Çalışma kapsamında Canny, Prewitt, Sobel ve Roberts Cross olarak bilinen kenar tespit yöntemlerine odaklanılacağından, bunlarla ilgili piksel komşulukları üzerinden hesaplanan diferansiyel değer tespit yöntemlerinden bahsedilecektir.

Bu yöntemlerdeki temel yaklaşım, görüntü verilerinin önemli yönlerini ve görüntü alanı özelliklerindeki değişiklikleri tespit etmek için görüntüdeki süreksizliği veya piksel değerlerindeki keskin değişiklikleri analiz etmektir. Söz konusu değişiklikleri belirlemek için görüntü, komşu piksellere göre değer farkını her piksel için ayrı ayrı tahmin eden diferansiyel değer algılama yaklaşımı ile işlenir. Bunun için bu yöntemlerde, komşu pikseller arasındaki değişimi tespit etmek için onlardan benzersiz maskeler kullanılır. Bu maskeler, pikselin komşuluğundaki diğer piksel değerleri kullanılarak görüntüdeki her piksele uygulanır. Böylece bu aşamada girdi olarak kullanılan görüntü ile aynı gri renk seviyesi değerlerine sahip bir kenar algılama görüntüsü elde edilir.

Ancak bu kenar bulma yöntemlerinde koordinat düzleminde  $x$  ve  $y$  eksenlerinin her biri için ayrı ayrı uygulanacak bir çift maske bulunmaktadır. Görüntüde her iki eksenindeki diferansiyel değeri belirlemede kullanılan maske çiftlerinin her ikisi de kenar algılama yöntemi için maske formunu oluşturmaktadır. Daha sonra iki eksen

için ayrı ayrı hesaplanan fark değerlerinden ilgili piksel için tek bir fark değeri belirlenir. Başka bir deyişle, eksenlerle aynı doğrultudaki iki vektörün birleşiminin büyüklüğü hesaplanır.

Görüntü üzerinde bir  $P_0$  pikseli için x eksen ve y eksen yönlerinde maskeleme sonrası elde edilen değerler sırasıyla  $D_x$  ve  $D_y$  olup,  $I(x, y)$  pikseli için hesaplanan gradyan değeri  $D$  (kenar algılama değeri) aşağıda verilmiştir:

$$\nabla I(x, y) = \begin{bmatrix} D_x \\ D_y \end{bmatrix} \quad (2.11)$$

$$D = \|\nabla I(x, y)\| \quad (2.12)$$

$$D = \sqrt{D_x^2 + D_y^2} \quad (2.13)$$

Yukarıda,  $\nabla$  gradyan operatörüdür,  $\nabla I(x, y)$  x ve y konumundaki  $I$  imajının gradyanı ve  $\|\cdot\|$  Öklid normudur. Bu tekniklerin maske biçimleri aşağıda açıklanmıştır.

#### 2.4.1 Canny Kenar Algılama Maskesi

Canny kenar algılama, 1986 yılında John F. Canny tarafından geliştirilen bir kenar algılama operatörüdür (Canny, 1986). Canny kenar detektörü, kenar algılamanın üç genel kriterini karşılamayı amaçlar: 1) Düşük hata oranı- bu, yalnızca mevcut kenarları algıladığı anlamına gelir. 2) İyi yerelleştirme - Gerçek kenar pikselleri ile algılanan kenar pikselleri arasındaki farkın en aza indirilmesi gerekir. 3) Minimum yanıt - algılanan kenar, birçok kez değil, yalnızca bir kez işaretlenmelidir.

Canny kenar algılamanın adımları aşağıdaki gibidir: 1) Bir filtre kullanarak tüm gürültüleri giderin. 2) Görüntünün gradyanını bulun. 3) Maksimum olmayan bastırma gerçekleştirilir. 4) Histerezis- Son adımda, üst ve alt olarak adlandırılan iki eşik kullanılır. Bu yöntemde kullanılan  $2 \times 2$  maskeleme, Şekil 2.3'teki gibi x ve y eksen bileşenleri şeklinde belirtilebilir.



|   |        |
|---|--------|
| 1 | -<br>1 |
| 0 | -<br>a |

a) x eksen maskesi

|        |        |
|--------|--------|
| 1      | 0      |
| -<br>1 | -<br>a |

b) y eksen maskesi

Şekil 2.3 Canny kenar algılama maskeleme gösterimi

Bir I görüntüsündeki bir  $I(x, y)$  pikseli için Canny kenar algılama maskeleme işlemi için gradyan değeri çiftinin hesaplanması aşağıdaki gibidir:

$$D_x = I(x, y) - I(x + 1, y) - a I(x + 1, y + 1) \quad (2.14)$$

$$D_y = I(x, y) - I(x, y + 1) - a I(x + 1, y + 1) \quad (2.15)$$

Burada a değeri sıfır olabilirken, pratik alanına uygun olarak bu değışkende değışiklikler yapılabilir. Canny kenar bulma tekniğı için birçok modifikasyon çalışması olmasına rağmen, a için genellikle en yakın “0.707 ( $1 / \sqrt{2}$ )” değeri tercih edilir.

#### 2.4.2 Roberts Cross Kenar Algılama Maskesi

Roberts çapraz kenar detektörü, ilk olarak 1963 yılında Lawrence Roberts tarafından önerilen gradyan tabanlı bir kenar detektörüdür. Robert'ın çapraz operatörü, basit ve hesaplanması hızlı bir görüntü üzerinde 2 boyutlu uzaysal gradyan ölçümü gerçekleştirmek için kullanılır. Robert'ın çapraz operatöründe, her noktada piksel değeri, o noktadaki giriş görüntüsünün mutlak büyüklüğünü temsil eder. Roberts çapraz operatörü, çapraz olarak bitişik pikseller arasındaki farkların karelerinin toplamını hesaplayarak elde edilen ayrık farklılaşma yoluyla bir görüntünün gradyanını yaklaşık olarak hesaplar.

Roberts Çapraz kenar algılama yöntemine ait maske çifti, Şekil 2.4'te verilmiştir. Şekil 2.4'de gösterildiğı gibi bir çift 2\*2 evrişim çekirdeğinden oluşur. Bir çekirdek basitçe değeri 90° döndürölür. Bu Sobel operatörüne çok benzemektedir.

|   |        |
|---|--------|
| 1 | 0      |
| 0 | -<br>1 |

a) x eksen maskesi

|        |   |
|--------|---|
| 0      | 1 |
| -<br>1 | 0 |

b) y eksen maskesi

Şekil 2.4 Roberts Cross kenar algılama maskesi

Bir I görüntüsündeki  $I(x, y)$  pikseli için Roberts Cross kenar algılama yöntemi için gradyan değeri çiftinin hesaplanması aşağıdaki gibidir:

$$D_y = I(x + 1, y) - I(x, y + 1) \quad (2.16)$$

$$D_y = I(x + 1, y) - I(x, y + 1) \quad (2.17)$$

### 2.4.3 Prewitt Operatör Maskesi

Prewitt operatörü, çok çeşitli uygulamalarda dijital görüntülerdeki kenar desenlerinin tespitinde kullanılır. Bu yöntemde maskenin ortasındaki eleman ile kenar değeri hesaplanacak piksel değeri geometrik olarak çarpılarak 3x3 boyutunda maskeler kullanılır. Sobel operatörünün aksine, bu operatörün maskelerin merkezine daha yakın olan piksellere herhangi bir vurgu yapmadığına dikkat edilmelidir. Bu tekniğin temel avantajı, görüntülerde yatay ve dikey kenarlarda daha iyi performans ve gürültülü görüntü için daha yüksek tepkiler sağlamasıdır. Şekil 2.5’de, Prewitt operatör maskesine ait formlar verilmiştir.

|    |   |   |
|----|---|---|
| -1 | 0 | 1 |
| -1 | 0 | 1 |
| -1 | 0 | 1 |

a) x eksen maskesi

|    |    |    |
|----|----|----|
| 1  | 1  | 1  |
| 0  | 0  | 0  |
| -1 | -1 | -1 |

b) y eksen maskesi

Şekil 2.5 Prewitt operatör maskesi

Bir  $I$  görüntüsü üzerindeki  $I(x, y)$  pikseli için Prewitt kenar algılama yönteminin gradyan değeri çiftinin hesaplanması aşağıdaki gibidir:

$$D_x = I(x+1, y-1) + I(x+1, y) + I(x+1, y+1) - I(x-1, y-1) - I(x-1, y) - I(x-1, y+1) \quad (2.18)$$

$$D_y = I(x-1, y-1) + I(x, y-1) + I(x+1, y-1) - I(x-1, y+1) - I(x, y+1) - I(x+1, y+1) \quad (2.19)$$



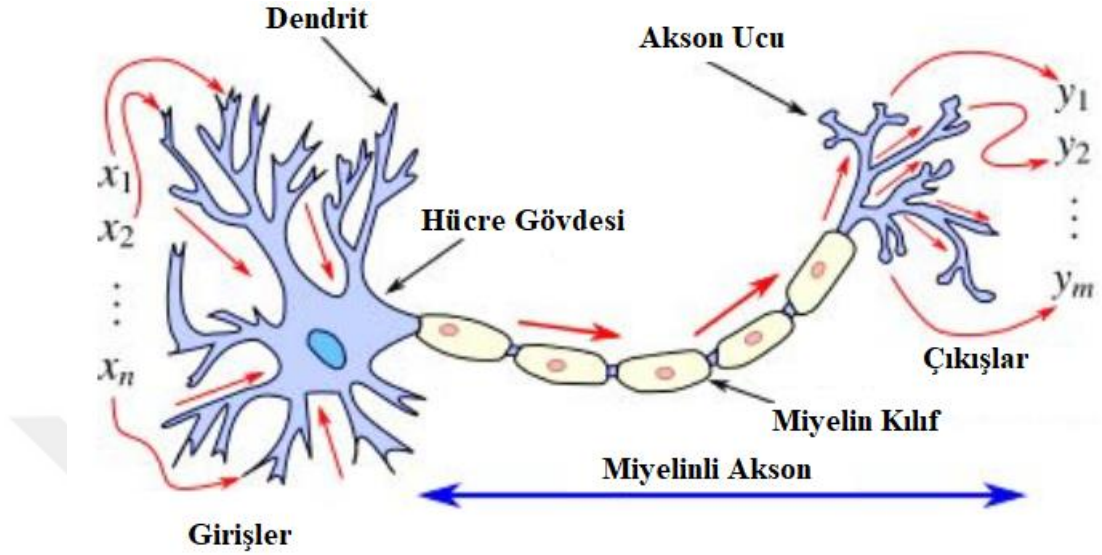
### 3. YAPAY SİNİR AĞLARI

Yapay sinir ağı, birbirine bağlı nöronlar olarak bilinen çok sayıda sinir hücresinden oluşan insan beynindeki sinir ağlarının matematiksel temsilleridir. Yapay nöron, yapay sinir ağındaki bir bağlantı noktasıdır. İnsan beyninin içinde yaklaşık 10 milyar nöron ve 60 trilyon sinaps vardır. Beyin, birkaç nöronu aynı anda kullanarak amacını dünyadaki en hızlı bilgisayarlardan bile daha hızlı gerçekleştirebilir. Her nöronun çok temel bir yapısı olmasına rağmen büyük bir bilgi işlem gücüne sahiptir. YSA'lar insan beyninin öğrenme süreçlerini kopyalamaya çalışır. İlk YSA teorileri, insan davranışını ve düşünme sürecini insan beynini modellemek için çalışan araştırmacılar tarafından açıklanmıştır (Gurney, 2018; Kröse vd., 1993). YSA'larında dört farklı güncel araştırma alanı vardır (Gültepe ve Duru, 2018; Tan, 1997):

- ✓ İnsan beyni ve işlevlerini anlamak için biyolojik ağı modellemek için YSA'lar kullanılmaktadır.
- ✓ Geleneksel Yapay Zeka (YZ) metodolojilerinin ve bilgisayar algoritmalarının çözmekte zorlandığı karmaşık görevlerin nasıl çözüleceğini anlamak için YSA'ları bir eğitim aracı olarak kullanılmaktadır.
- ✓ Çeşitli ticari uygulamalardaki gerçek dünya problemlerini çözmek için YSA'lar kullanılmaktadır. Bu alandaki birçok araştırmacı, YSA ile ilgili olanlar dışındaki alanlarda geçmişe sahiptir. Bir YSA kullanmanın cazibesi, onu bir araç olarak kullanmanın basitliği ve rapor edilen YSA tabanlı ticari uygulama başarılarıdır.
- ✓ YSA algoritmalarının iyileştirilmesi alanındaki araştırmacılar, daha verimli 'öğrenebilen' veya modelleyebilen daha iyi YSA algoritmaları oluşturmakla ilgilenmektedir; yani daha hızlı eğitim süreleri ve/veya daha doğru sonuçlar elde edilir.

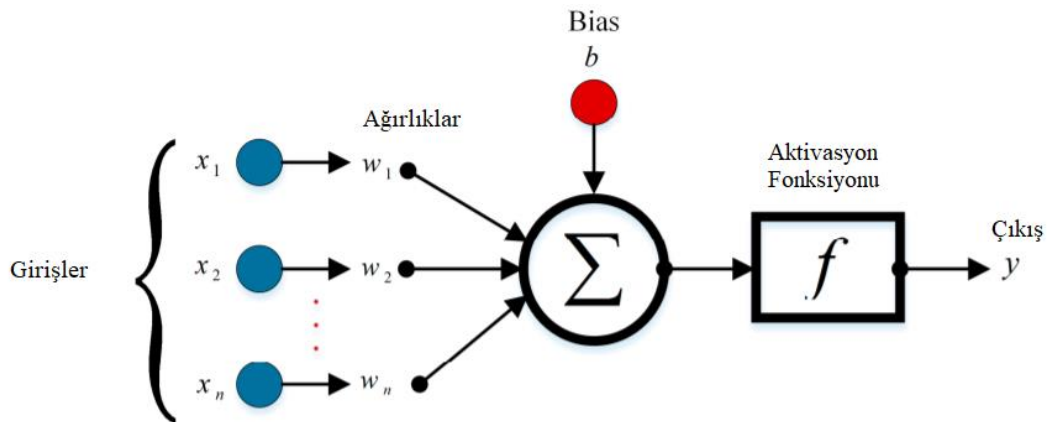
Şekil 3.1'de, biyolojik sinir hücresi gösterilmiştir (Krosche vd., 1993). Sinir hücreleri (nöronlar), dendritler adı verilen kapsamlı süreçlere sahiptir. Dendritler, giriş

sinyallerini nörona iletir ve bir akson, çıkış sinyalini sinapslar üzerinden diğer nöronların dendritlerine doğru iletir.



Şekil 3.1 Biyolojik sinir hücresi

Yapay sinir ağlarında oluşturulan basit bir hücrenin yapısı Şekil 3.2'deki gibidir. Bu basit yapı; girdiler, ağırlık değerleri, çıktılar, toplam fonksiyon ve aktivasyon fonksiyonundan oluşur. Veriler hücreye diğer hücrelerden veya dış ortamlardan geldiğinden giriş kesindir. Bunlar, sistemin öğrenmek istediği örnekler tarafından güçlendirilir. Ağırlıklar, bir sonraki işlemde girişteki veya önceki katmandaki hücrenin sonucunu gösteren tanımlanmış değerlerdir. Girişler, toplam fonksiyon aracılığıyla ağırlık değeri ile birleştirilir ve bir sonraki hücreye bağlanır.



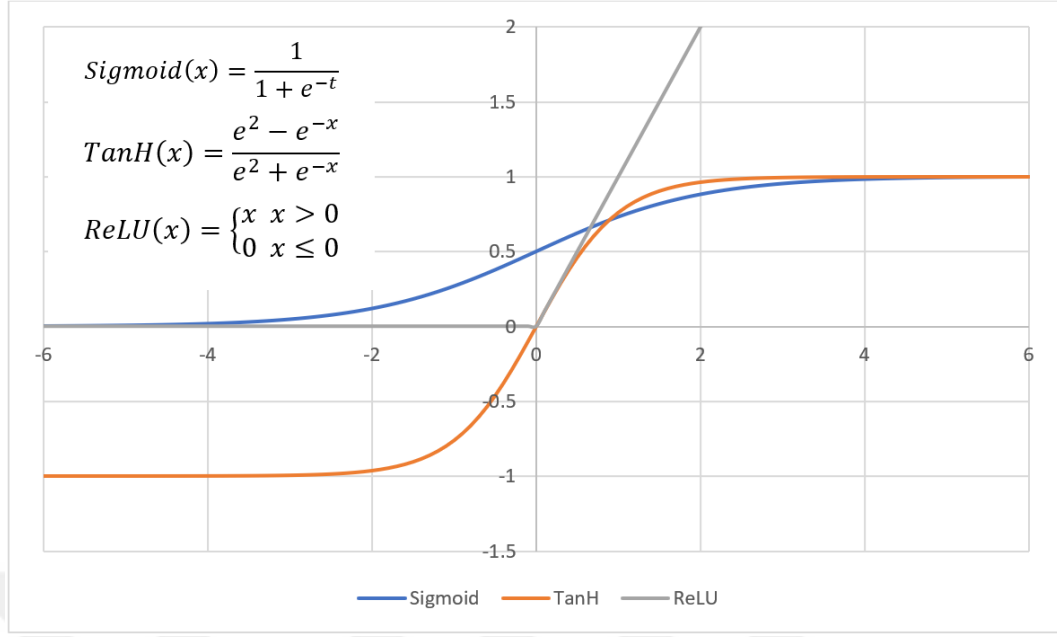
Şekil 3.2 Yapay sinir hücresi

Girdiler ( $x_1, x_2, \dots, x_n$ ), hücreye diğer hücrelerden veya dış ortamdan giren bilgiler şeklindedir. Bunlar ağıın öğrenmesi istenen örnekler tarafından belirlenir. Ağırlıklar ( $w_1, w_2, \dots, w_n$ ), her bir özellik ile ilişkilendirilen ve o özelliğin nihai değeri tahmin etmedeki önemini anlatan gerçek değerlerdir. Toplama fonksiyonu, işleme elemanlarından net girdinin hesaplanmasını sağlayan fonksiyondur. Ayrıca var ise nörona ait bir eşik değeri (bias ( $b$ )) eklenir. Transfer fonksiyonu toplama fonksiyonu tarafından sağlanan net girdiye karşılık üretileceği çıktıyı belirlemesini sağlar. Denklem 3.1’de toplam fonksiyonu matematiksel olarak gösterilmiştir.

$$T(x) = \sum_{i=1}^n w_i x_i + b \quad (3.1)$$

Aktivasyon fonksiyonu, girdinin veya girdi kümesinin çıktısını tanımlar veya başka bir deyişle, girdilerde verilen düğümün çıktısının düğümünü tanımlar. Aktivasyon fonksiyonları olmadan çıkış sinyalleri sadece basit lineer fonksiyonlardır. Doğrusal işlevlerin karmaşıklığı sınırlıdır ve verilerden karmaşık işlev eşlemelerini öğrenme yeteneği düşüktür.

Yapay sinir ağlarında en yaygın olarak kullanılan aktivasyon fonksiyonları ve grafikleri Şekil 3.3’de verilmiştir (Karlık ve Olgac, 2011; Wan vd., 2013; Xu vd., 2015). Bunlar şu şekildedir: Sigmoid, Hiperbolik Tanjant (TanH) ve Rektifiye Doğrusal Birimdir (ReLU). Adından da anlaşılacağı gibi sigmoid fonksiyonunun eğrisi S şeklindedir. Sigmoid, 0 ile 1 aralığındaki değerleri dönüştürür. Tanh Aktivasyon fonksiyonu, Sigmoid Aktivasyon fonksiyonundan daha üstündür çünkü bu aktivasyon fonksiyonunun aralığı sigmoid aktivasyon fonksiyonundan daha yüksektir. Tanh, -1 ile 1 aralığındaki değerleri dönüştürür ReLU, girdi pozitif ise  $x$  diyelim, çıktı  $x$  olacak, aksi halde sıfır çıktı veren parçalı lineer bir fonksiyondur (Ren vd., 2016; Sharma vd., 2020; Xu vd., 2015).



Şekil 3.3 Nöronlar için aktivasyon fonksiyonları

### 3.1 Yapay Sinir Ağlarının Sınıflandırılması

Yapay sinir ağlarını yapılarına göre sınıflandırmak mümkündür. Bu bölümde yapay sinir ağlarının yapılarına göre sınıflandırılması açıklanmıştır. Yapıları açısından sinir ağları iki türe ayrılabilir: İleri beslemeli ağlar ve tekrarlayan ağlar.

#### 3.1.1 İleri Beslemeli Yapay Sinir Ağları

İleri beslemeli bir sinir ağı, en basit yapay sinir ağı mimarisidir. Bilgi, giriş düğümlerinden, gizli düğümler (varsa) boyunca ve çıkış düğümlerine doğru, yalnızca bir yönde hareket eder. Ağda döngü yoktur. Böyle bir ağın en yaygın eğitim yöntemi, düğümler arasındaki bağlantıların ağırlıklarının ayarlamak için çıktı sonucunu geriye doğru kullanmaktır (Gültepe, 2019; Moldovan vd., 2020).

İleri beslemeli sinir ağlarında, nöronlar katmanlar halinde dağıtılır, burada her katmandaki nöronların çıktıları bir sonraki katmandaki nöronların girdilerine bağlanır. Bu ağlarda giriş, çıkış ve gizli katman olmak üzere üç tür katman vardır. Girdi katmanı, dış dünyadan gelen girdilerin bağlandığı ilk katmandır ve bu katmandaki nöron sayısı sinir ağının girdi sayısına eşitken, çıktı katmanındaki nöron sayısı ise sistemden istenen çıktı sayısı birbirine eşittir. Böylece, bu katmanlardaki nöronların sayısı, sinir

ağının uygulandığı göreve uyması için gereken uygulama tarafından kontrol edilir (Nureddin vd., 2020; Sheinfeld vd., 2017). İleri beslemeli sinir ağı örnekleri olarak; çok katmanlı algılayıcı (MLP) (Sağıroğlu vd., 2003), radyal temel fonksiyon (RBF) ağı (Broomhead ve Lowe, 1988), öğrenme vektörü niceleme (LVQ) ağı (Kohonen, 1987), serebellar model artikülasyon kontrolü (CMAC) ağı (Albus, 1975), grup yöntemiyle bilgi işleme (GMDH) yöntemi (Hecht-Nielsen, 1990) sıralanabilir.

### **3.1.2 Geri Beslemeli Yapay Sinir Ağları**

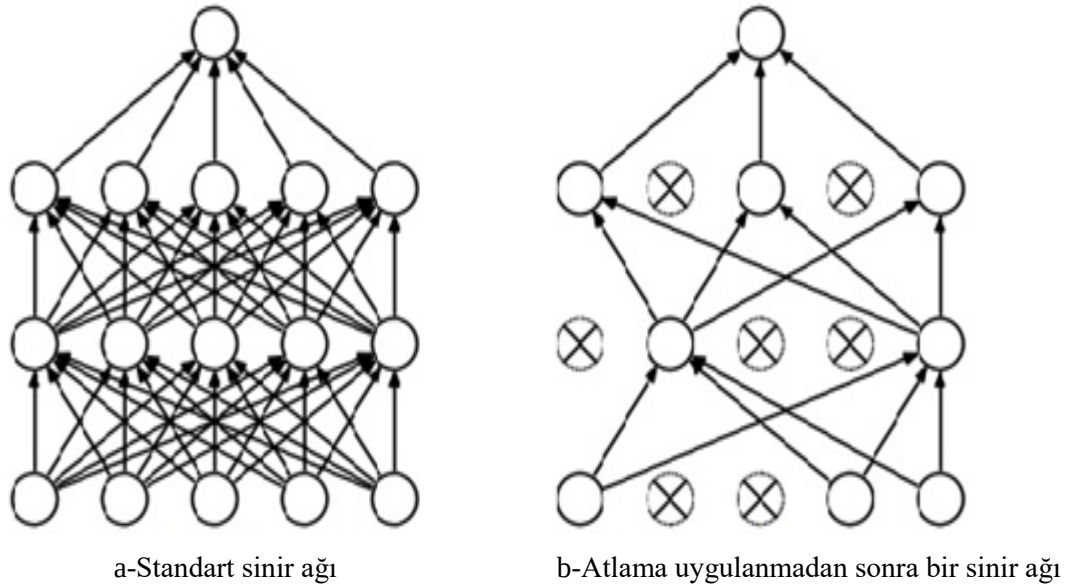
Geri beslemeli sinir ağının ana işlevi, ağ bilgilerini aşamalar halinde tekrarlamaktır. İlk bilgi girildikten sonra, bilgi durumu katman katman aktarılabilir, böylece tüm ağ durumu dinamik bir dengeye ulaşabilir. Aynı zamanda, geri besleme ağının gerçek zamanlı geri bildirimi yoluyla, bilgi çeşitli alanlara iletilebilir ve bilgilerin içeriği veri biçiminde çıktılanabilir ve son olarak çıktı terminali aracılığıyla entegre çıktı alınabilir. Bu yapısı ile geri beslemeli YSA, doğrusal olmayan dinamik bir davranış gösterir (Sun vd., 2021).

### **3.2 Derin Sinir Ağları**

Sinir ağlarının performansını artırmak için giriş ve çıkış katmanları arasına daha fazla katman eklenir. Bu katmanlar, sistemin giriş ve çıkışlarına bağlı olan giriş ve çıkış katmanlarının aksine dış dünya tarafından görülmediklerinden gizli katmanlar olarak bilinirler. Gizli katmanlardaki nöronların sayısı herhangi bir dış faktör tarafından kısıtlanmaz, ancak bir gizli katmandaki nöronların sayısını veya gizli katmanların sayısını artırmak, bir girdi için bir sınıfı tahmin etmek için gereken hesaplamaların karmaşıklığını artırır, bu tahmin için gereken süre de artar. Ayrıca, gizli bir katmandaki nöron sayısını artırmak, o katmanda algılanabilecek özelliklerin sayısını, yani gizli katmandaki nöronları ateşleyen kombinasyonların sayısını artırırken, katman sayısını da artırır. Böylelikle ağın derinliği, daha derin katmanlarda algılanabilen özelliğin karmaşıklığını artırır. Bu nedenle, birden fazla gizli olan ağlar daha karmaşık özellikleri tespit edebilen çok katmanlı yapay sinir ağlarına derin öğrenme denir (Ahmed vd., 2021b; Goodfellow vd., 2016; Zarbega ve Gültepe, 2020).



Çoğu derin öğrenme yöntemi, sinir ağı mimarilerini kullanır, bu nedenle derin öğrenme modellerine genellikle derin sinir ağları denir. Derin sinir ağlarının karşılaştığı ana zorluklardan biri, tahminlerin sinir ağındaki belirli özelliklere dayandığı ve bu tahminlerin bu özelliklerle çok sınırlı hale geldiği aşırı uyum olgusudur. Bu nedenle, o sınıfa ait olabilecek ancak bu özelliklere karşılık gelen nöronları tetiklemeyen herhangi bir yeni girdi, büyük olasılıkla yanlış sınıflandırılacaktır. Böyle bir sorunun üstesinden gelmek için, gizli bir katmandaki nöronların önceden tanımlanmış bir oranı, eğitim aşamasının her yinelenmesinde rastgele düşürülür, böylece sinir ağı aynı tahmin için alternatif yollar bulmaya ve belirli özelliklere bağımlılığı azaltmaya zorlanır. Bu yaklaşım Atlama (Dropout) olarak bilinir ve sinir ağları tarafından sağlanan tahminlerde iyi bir gelişme göstermiştir ve Şekil 3.4'te gösterilmiştir (Ab-BelKhair vd., 2020; Srivastava vd., 2014).



Şekil 3.4 Sinir ağı modelleri

### 3.3 Yapay Sinir Ağları Eğitimi

Biyolojik sinir ağının topolojisinin ve sinapsların iletkenliğinin beyin tarafından alınan kararları tanımladığı insan beynine benzer şekilde, YSA'ları da gerekli kararı vermek için nöronların dağılımına ve aralarındaki ağırlıklara güvenir. İki özdeş sinir ağı, nöronları arasında farklı ağırlık değerleri kullanılarak farklı kararların alındığı

tamamen farklı görevlerde kullanılabilir. İki nöron arasındaki ağırlık değeri, etkinin tipini, önceki katmandaki nöronun sonraki katmandaki nörona göre çıktısını ve bu etkinin sonraki katmandaki nöronun çıktısı üzerindeki önemini tanımlar (Sethi ve Jain, 2014).

Geri yayılım, sinir ağlarının popülaritesinde kilit bir role sahiptir, çünkü bu teknik, ağırlıklarının değerini güncellemek için kullanıldığında bu ağların performansı önemli ölçüde iyileşir. YSA'nın  $w$  ağırlıklarını güncellemek için, geri yayılım, Denklem 3.2'de gösterildiği gibi, güncellenen kayba  $\partial O/\partial w$  göre ağırlık çıktısının değişim oranı olan üç değer gerektirir; ağırlık çıktısı ve ondan gerçekten gerekli olanı ve öğrenme oranı  $L$ 'dir (Hecht-Nielsen, 1992).

$$\hat{w} = w - \frac{\partial O}{\partial w} \times E \times L \quad (3.2)$$

Çapraz entropi ve Ortalama Karesel Hata (MSE) işlevleri gibi sinir ağı tarafından kullanılan hata işlevinin türü ne olursa olsun, bu işlevler mevcut ağırlık değerlerini ve ağırlıklardan istenen değerleri kullanarak sinir ağının çıktısı arasındaki farkı temsil eden tek bir değer hesaplar. Sinir ağının çıktısı, sinir ağının ileri geçişi kullanılarak eğitim veri setinden bir grup örnek girdi işlenerek, gerçek çıktılar ise doğrudan eğitim veri setinden veya girdilerin önceden tanımlanmış fonksiyonlar kullanılarak işlenmesiyle toplanır. Hesaplanan hata değeri daha sonra geri yayılımda tarafından kullanılır. Ancak, büyük hata değerleri büyük delta değerleri üretebileceğinden, ağırlık güncellemeleri için daha düşük aralıklarda delta değerlerini kontrol etmek için bir öğrenme oranı kullanılır. Delta değerlerinin bu kontrolü, patlayan ağırlık değerlerinden kaçınılmasını sağlar, böylece minimum hatayı üreten ağırlık değerleri keşfedilebilir (Mäkeläinen vd., 2019).

Çıkış hatasının ağırlık değerlerine göre değişim oranı hesaplanarak, üç olası değer üretilir (Çilimkovic, 2015):

- ✓ Ağırlık değerini artırmanın hatayı artırdığını gösteren pozitif bir değer. Bu nedenle, sinir ağının çıktıları ile gerekli olanlar arasındaki farkı azaltmak için ağırlık değeri hesaplanan delta değeri kadar azaltılmalıdır.

- ✓ O ağırlığın değeri artırılarak hatanın azaldığını gösteren negatif bir değeri. Bu nedenle, hata değerini azaltmak ve daha doğru çıktılar üretmek için mevcut ağırlık değeri hesaplanan delta değeri kadar artırılmalıdır.
- ✓ Ağırlığın mevcut değerinde herhangi bir değişiklik gerekmediğini gösteren sıfır değeri kullanılır.

Bu olası değerlere göre ve Denklem 3.2’de gösterilen formül kullanılarak, yapay sinir ağının tahminleri ile YSA’nın görevini yerine getirmek için gereken gerçek çıktı arasındaki farkı azaltmak için sinir ağındaki ağırlıkların değerleri güncellenebilir.

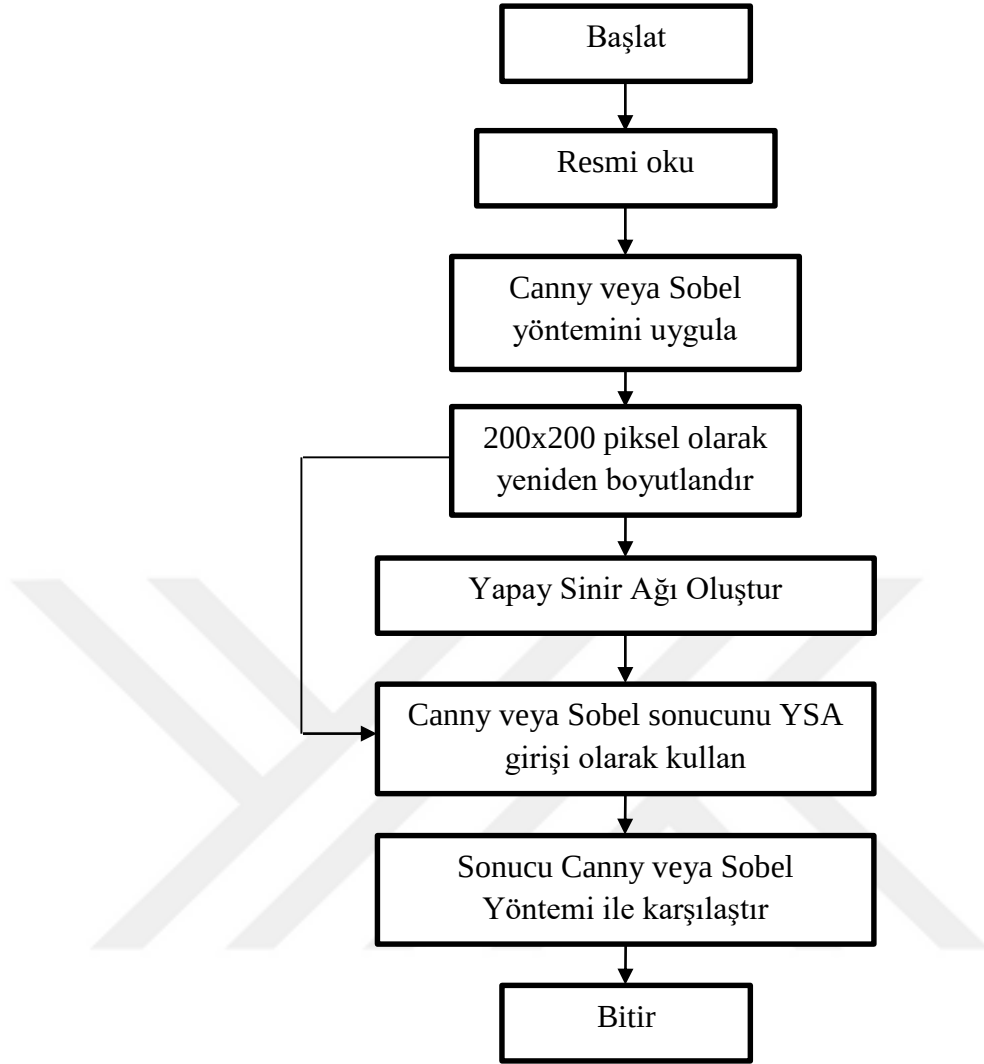
Ancak öğrenme hızının ihtiyacına göre ağırlık değerlerini güncellemek için kullanılan delta değerini azaltmak için, ağırlıkların güncellenmesi yoluyla hatayı en aza indirerek üretilen sinir ağının optimal performansı, optimal ağırlık değerlerinin hesaplanması birden fazla iterasyon gerektirir (Schmidhuber, 2015).

## 4. SİMÜLASYON SONUÇLARI

Yapay sinir ağları, biyolojik nöronların verileri nasıl işlediğinden ilham almaktadır. Yapay sinir ağları, makine öğrenimi alanında önemli başarılar yaratmıştır. Görüntü işleme, otonom sürüş, konuşma tanıma ve karakter tanıma gibi alanlarda büyük başarılarla imza atmıştır.

### 4.1 Metot

Kenar algılama, görüntü içindeki uzamsal süreksizlikleri tespit etmek ve bulmak için kullanılan bir işlemdir. Kenar, görüntü kontrastının veya yoğunluğunun büyük ölçüde değiştiği sonlu alan olarak tanımlanır. Kenar Algılama, yoğunlukta yüksek kontrastların olduğu sonlu alanları bulur. Kenar algılama, bir görüntüde şekil, konum, ölçek, ayrıntı keskinleştirme ve görüntüde bulunan bozulmaların iyileştirilmesi gibi ayrıntıların toplanmasına yardımcı olur. Bu tez çalışmasında kenar algılama için en yaygın olan Sobel ve Canny kenar algılama yöntemine dayalı modeli eğitmek için yapay sinir ağını kullanılmıştır. Önerilen yöntemin akış şeması, Şekil 4.1’de gösterilmiştir. Akış şeması, genel olarak altı modülden oluşmaktadır. Sırasıyla, veri kümesinden resimleri okuma, Canny veya Sobel yönteminin uygulanması, veri setindeki resimlerin yeniden boyutlanması, yapay sinir ağının oluşturulması, uygulanan yönteminin sonucunu YSA giriş olarak kullanılması ve elde edilen sonucun Canny veya Sobel yöntem sonuçları ile karşılaştırılmasıdır. Bu teknikler için MATLAB kullanılmıştır. MATLAB, geleneksel ve son teknoloji görüntü işleme teknikleri ve uygulamalarına kıyasla çeşitli avantajlar sağlar. Örneğin, MATLAB, yeniden derleme olmaksızın kolay uygulama ve algoritmik testler ve kapsamlı veri analizi ve görselleştirme ile kolay hata ayıklama sağlar. Ayrıca MATLAB’ın hesaplama kodları, hem durağan hem de video görüntülerinin simülasyonlarını işlemek için oluşturmak için kolayca geliştirilebilir ve kullanılabilir (Buksh vd., 2014; Nagi vd., 2008).



Şekil 4.1 Görüntü kenarı algılama için önerilen yöntemin akış şeması

Bu tezde, 64 bitlik bir damga 512x512 boyutlarında gri tonlamalı test görüntülerinden oluşan standart veri seti kullanılmıştır. Bu veri seti <https://ccia.ugr.es/cvg/CG/base.htm> adresinde yer almaktadır. Bu veri setinde toplam 49 görüntü bulunmaktadır.

Görüntü kenarı algılama için önerilen yöntemi değerlendirmek için Şekil 4.2’de gösterildiği şekilde; Cameraman, Forest, Boy, Pepper, Alpha ve Gorilla olarak toplam altı deneysel görüntü kullanılarak deneyler gerçekleştirilmiştir ve yapılan deneyleri sonuçları çok umut vericidir.



Şekil 4.2 Örnek görüntüler

Yapay sinir ağları, görüntü üzerinde kenar algılama konusunda klasik yöntemlere göre daha az çalışma yükü sağlamış ve gürültünün etkisini azaltmak için daha fazla avantajlar sağlamıştır; ve akademik çalışmalarda sıkça kullanılmaya başlanmıştır. Görüntünün eğitimi aşamasında birden fazla girdi ve çıktı kullanılabilmesine olanak tanıdığından yapay sinir ağı modern zamanlarda en kullanışlı ve güçlü yapay zeka yöntemlerinden biridir.

Tez çalışmasında kurulan YSA'na toplam 22 adet nöron sunulmuştur. Tek gizli katmana sahip bir YSA modelinin problemin çözümü için yeterli olduğu belirlenmiştir. Gizli katmana ait aktivasyon fonksiyonu sigmoid transfer fonksiyonları kullanılmıştır. YSA parametrelerini belirlemek için çok sayıda deneme yanılma yöntemi kullanılmıştır ve çok sayıda test yapılmıştır. Giriş katmanında 9 adet nöron, gizli katman için 12 ve çıkış katmanı için ise 1 adet nöron kullanılmıştır.

YSA, ileri yayılma sinir ağı kullanan bir MLP yapısına sahiptir.

- ✓ MLP'nin bir girdi katmanı, bir veya birden çok gizli katmanı ve bir çıktı katmanı vardır.

- ✓ Bilgi girdi katmanına girer ve tüm hücrelerde işlenir ve çıktı katmanına gönderilir.
- ✓ Sınıflandırma, karakter tanıma ve fonksiyon hesaplama vs. MLP'nin kullanım alanlarıdır.

Parametrelerin güncellenmesi işleminde ise geri yayılım (BP) yöntemi kullanılmıştır.

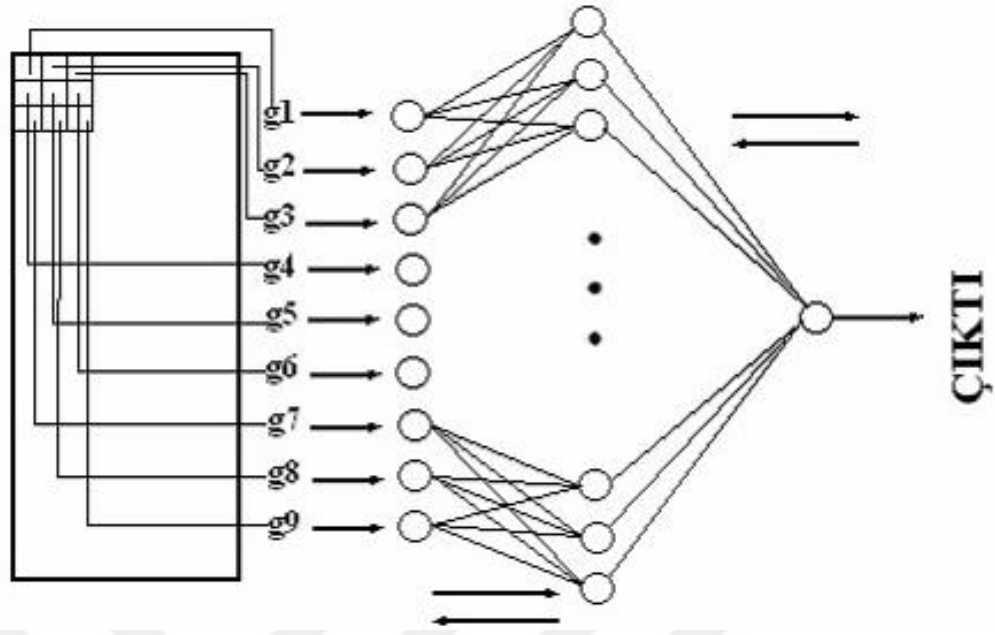
- ✓ BP, en çok kullanılan yöntemdir.
- ✓ BP'nin yöntemi, zamanında maliyet fonksiyonu ile parametreleri düşürmeye dayanmaktadır.

Geri yayılım algoritması, performansı etkileyen hataları en aza indirmek için ağırlıkların nasıl ayarlanacağını belirlemek için performans fonksiyonunun gradyanını kullanır.

#### **4.2. Simülasyon Sonuçları**

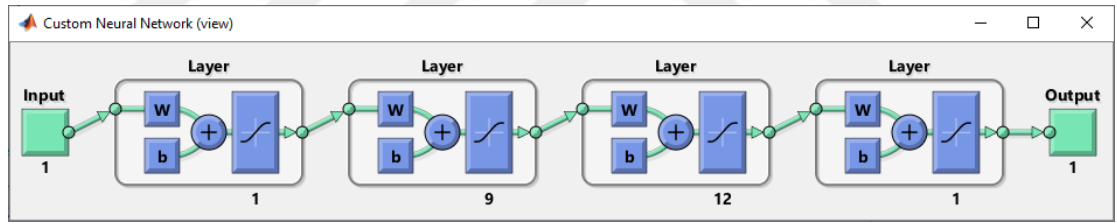
Bölüm 4.1'de açıklanan çok katmanlı algılayıcı ağ yapısına görüntülerin olduğu veri kümesi MATLAB'ın sağladığı eğitme ve aktivasyon algoritmaları ile uygulanmıştır.

Bu tezde, her bir düğümün aktivasyon fonksiyonu bir sigmoid fonksiyonu kullanır,  $f(x) = 1/(1 + e^{-x})$ . Sigmoid işlevi (0, 1) arasında değerler üretir, bu nedenle değerler ağ girişi yapılmadan önce normalleştirilir ve; 0 ve 1 arasına düşürülür. Şekil 4.3'te bu tez için tanımlanan MLP'nin mimarisini gösterilmektedir.



Şekil 4.3 Sistem için tasarlanan YSA modeli

Nöron mimarisi, Şekil 4.4'te gösterilmiştir.

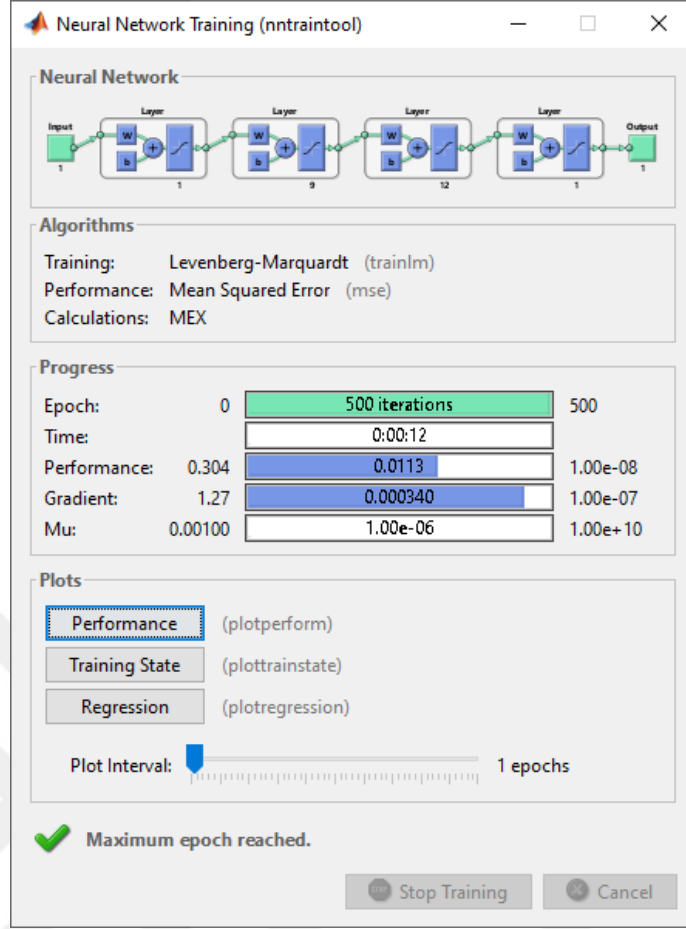


Şekil 4.4 Katmanlar ve nöronlar

Bu şekilde görüldüğü gibi giriş sayısı 1'dir ve bu eğitim için 1 piksel seçtiğimiz anlamına gelir. Sistemi eğitmek için üç gizli katman kullanılır. Tüm nöron sayısı ve katmanları Şekil 4.4'te gösterilmiştir.

Sinir ağı eğitim aracı, Şekil 4.5'te gösterilmiştir.

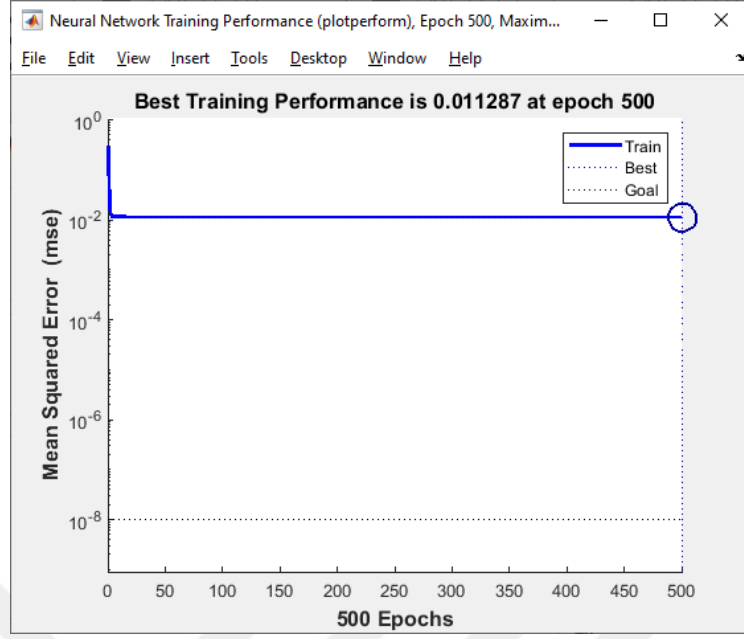




Şekil 4.5 Sinir ağı araçları

Bu şekilde görüldüğü gibi, YSA araçlarında epoch değerleri ve işlem süresi ve performansı açıkça gösterilmektedir. Giriş sayısı 1'dir ve bu, eğitim için 1 piksel seçtiğimiz anlamına gelir. Sistemi eğitmek için üç gizli katman kullanılır. Tüm nöron sayısı ve katmanları Şekil 4.5'te gösterilmiştir.

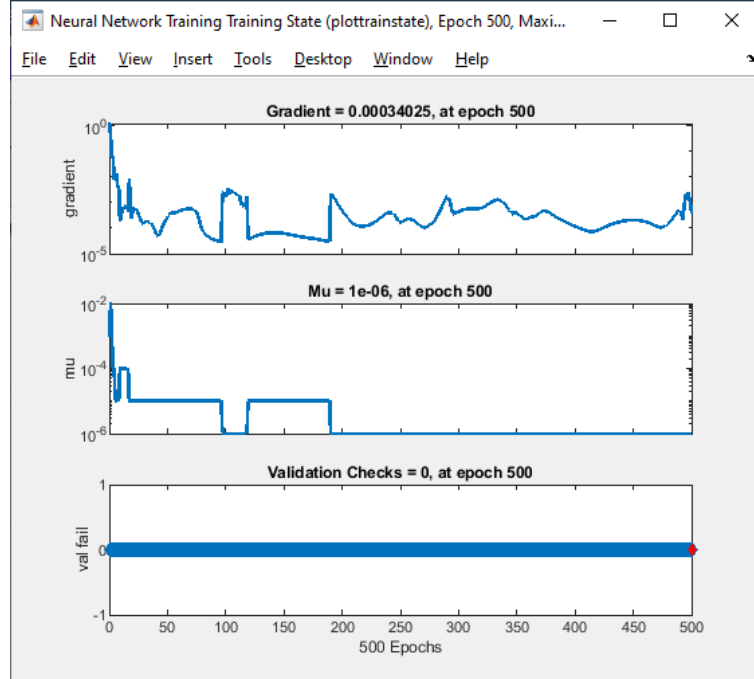
Sinir ağı eğitim performansı, Şekil 4.6'da gösterilmektedir.



Şekil 4.6 Sinir ağı eğitim performansı

Bu şekilde görüldüğü gibi, en iyi eğitim performans değeri, epoch 500'de 0,011287 olarak elde edilmiştir.

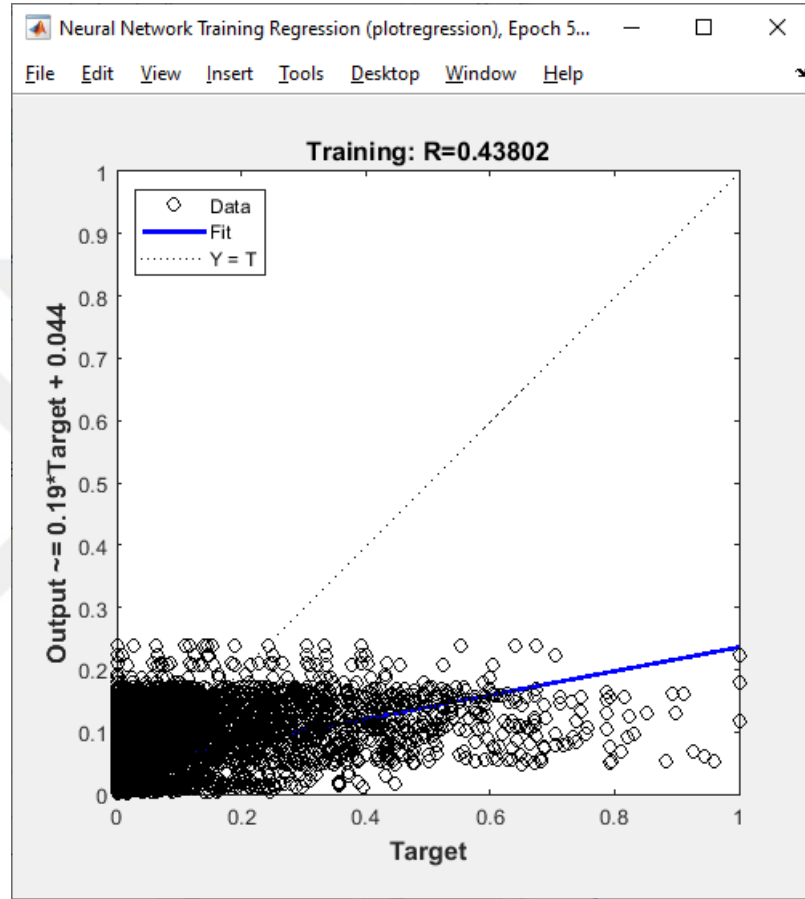
Epochlara karşı sinir ağı eğitim durumu, Şekil 4.7'de gösterilmektedir.



Şekil 4.7 Epochlara karşı sinir ağı eğitim durumu

Şekil 4.7’de, 500 epoch’taki Gradient, Mu ve Validation’ı göstermektedir. Gradient değeri, Mu ve Validation değeri sırasıyla 0,000334025, 0,000006 ve 0 olarak elde edilmiştir.

YSA’nın eğitim regresyon grafiği, Şekil 4.8’de gösterilmektedir.



Şekil 4.8 Eğitim regresyon grafiği

Bu sonuçta görüldüğü gibi regresyon, eğitim sisteminin doğruluğunu göstermektedir. Bu şekilden sinir ağının iyi eğitildiğini ve regresyon değeri 0,43802 olarak elde edilmiştir.

Bu tezde, tüm deneysel görüntüler, Sobel kenar algılama adı verilen geleneksel bir kenar algılama ile karşılaştırılmıştır. Simülasyon sonuçları Şekil 4.9 ile Şekil 4.13 arasında gösterilmiştir. Bu tez çalışmasında farklı bir sahneden 6 görsel seçilmiştir. Bu şekillerde, sol taraftaki resimler orijinal resimleri, ortadaki resimler klasik yönteme

dayalı sonuçları ve sütunun sağ tarafındakiler ise yapay sinir ağının sonucu gösterilmiştir.

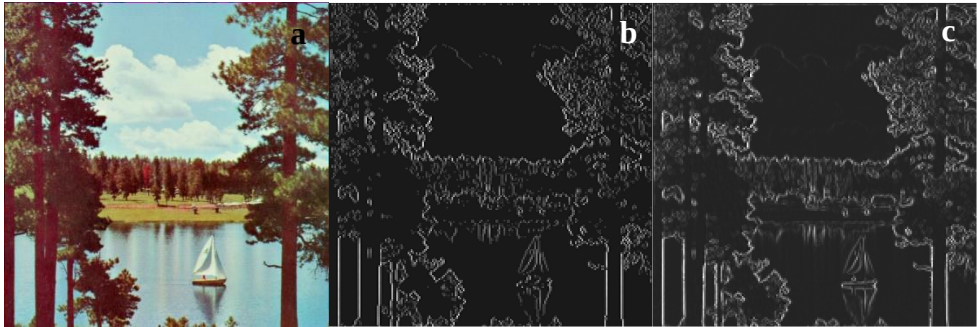
“Kameraman” görüntüsünün sonucu Şekil 4.9’da gösterilmiştir.



Şekil 4.9 a) Kameraman orijinal görüntüsü, b) Keyfi eşikleme ile Sobel kenar algılama, c) Kenar algılama yöntemi birleşik YSA

Şekil 4.9’da görüldüğü gibi, Sobel kullanılarak kenar tespit edilen görüntü iyi bir kenar iyileştirmesine sahip değildir. Örneğin Sobel kenar tespitinde bir kişinin yüzü net değildir ancak önerilen yöntemde bu sorun ortadan kaldırılmış ve başarılı sonuçlar elde edilmiştir.

“Göl” görüntüsünün sonucu Şekil 4.10’da gösterilmektedir.



Şekil 4.10 a) Göl orijinal görüntüsü, b) Sobel kenar algılama, c) Kenar algılama yöntemi birleşik YSA

Bir göl olan bu görüntüde önerilen yöntemde akarsu kenarları Sobel yöntemine göre çok daha iyidir.

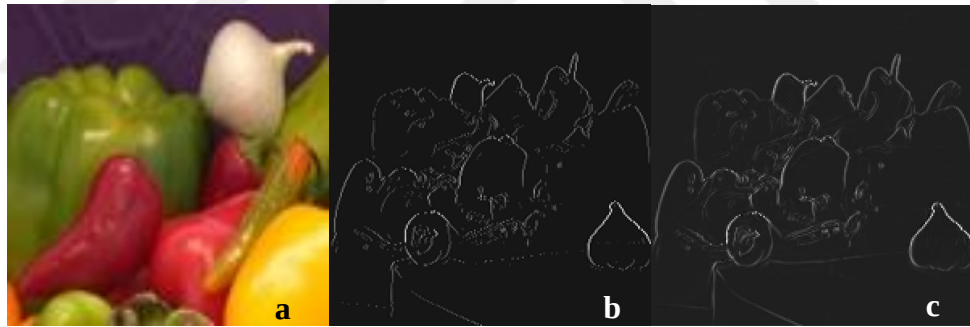
“Julei” görüntüsünün sonucu Şekil 4.11’de gösterilmektedir.



Şekil 4.11 a) Julei orijinal görüntüsü, b) Sobel kenar algılama, c) Kenar algılama yöntemi birleşik YSA

Şekil 4.11’de Julei isimli görüntüde bir kişinin yüz görüntüsü verilmiştir. Bu görüntüdeki kişinin dudaklarına dikkat etmenizın yanı sıra yüz ve burun kenarlarının ve yüzün diğer kısımlarının doğru bir şekilde belirlendiği yöntemimizde, kenarları tanıyabilmektedir.

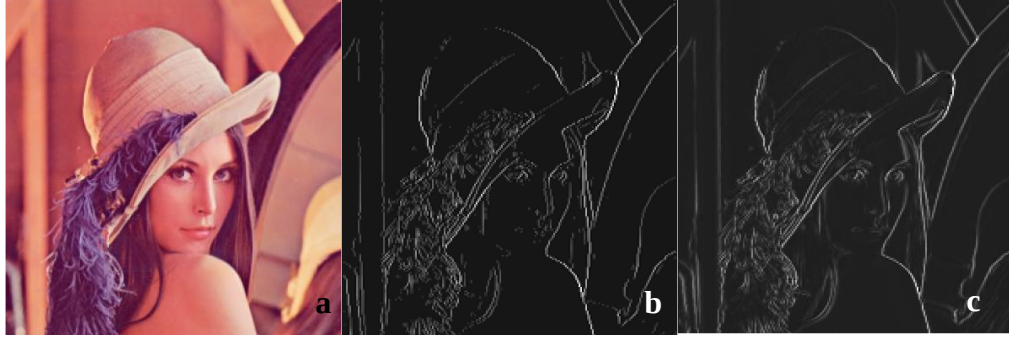
“Biber” görüntüsünün sonucu Şekil 4.12’de gösterilmektedir.



Şekil 4.12 a) Biber orijinal görüntüsü, b) Sobel kenar algılama, c) Kenar algılama yöntemi birleşik YSA

Bu şekilde dikkat ederseniz görüntünün farklı yerlerinde var olan eğrileri fark edeceksiniz ve bu eğrilerin görüntünün ortasında kalan bazı kenarları bizim yöntemimizde daha iyi tespit ediliyor.

“Lenna” görüntüsünün sonucu, Şekil 4.13’te gösterilmektedir.



Şekil 4.13 a) Lenna orijinal görüntüsü, b) Sobel kenar algılama, c) Kenar algılama yöntemi birleşik YSA

Önerilen yöntemde, kenarların detayının bulunmasında ve kenar algılama yönteminin çözümünde olumlu rol oynayan birleştirilmiş yapay sinir ağı ve klasik yöntem kullanılmıştır. Birleşik yöntemi çözmenin genel süreci, Şekil 4.1’de sunulmuştur. Bu yöntemde önce klasik kenar algılama yöntemi ile bir dizi cevap oluşturulmakta ve daha sonra bu cevapların bilgileri yapay sinir ağı için faydalı olacak şekilde elde edilmektedir. Bu cevaplar kullanıcının hedef rolünü oynamaktadır. Bu adımın sonunda yapay sinir ağına anlatılan yöntemle üretilen toplam cevap sayısına göre kenar güncellenir ve adımlar tekrarlanır.

Yukarıdakiler göz önüne alındığında, nöronların davranışları şu şekilde uygun bir açıklama yapılabilir: Her yinelemede, önce mevcut nöron soruna cevaplar bulur ve daha sonra çiftleşme sonucunda, yavrular, bu yavrulara tepkileri çıktı tepkilerinin özelliklerine sahip olan bu erken nesil nöronlardan doğar. Algoritmanın uygulanması, nöronların cevaplarının genellikle YSA kullanılmadan verilen cevaplardan daha iyi olduğunu göstermiştir. Bu algorithmada meydana gelen atlamalar ve kesişmeler nedeniyle, mutlak optimale ulaşmadan cevapların yakınsaması engellenir. Her iterasyonun sonunda, üretilen nöron ağırlıklarının özellikleri ve etkileri, nöronun yanıtının etkileri kullanılarak bir sonraki adıma iletilir.

Diğer görüntüler için elde edilen sonuçlar; a) orijinal görüntü, b) Sobel kenar algılama ve c) kenar algılama yöntemi birleşik YSA olarak Ekler kısmında Ek-A olarak verilmiştir.

## 5. SONUÇ VE ÖNERİLER

Kenar algılama alanındaki araştırmalar günümüzde önemli bir konu haline gelmiştir ve uygulamaları çok geniştir, görüntü işlemede montaj makinelerinin otomatik izlenmesi, tıbbi görüntülerde algılama ve tümseklerin tanımlanması vb. gibi geniş uygulamalar bulmuştur. Ancak kenar bulma şu anda en zor işlemlerden biridir.

Bir görüntü bir insan tarafından görüldüğünde, insan istemeden görüntünün kenarlarını belirler. Şu anda bir görüntüden herhangi bir kenar türünü başarıyla algılayabilen tek bir kenar algılama algoritması yoktur. Son yıllarda, bu önemli konuda, bazıları Roberts, Sobel, Parvit, Robinson ve Krish gibi maskeler kullanılarak evrişim işlemleri, diğerleri ise türev operatörleri kullanılarak gerçekleştirilen birçok çalışma yapılmıştır. İkinci mertebe laplasin ve iki yönlü bir türev gibi davranır. Gerçekleştiren bu maskeler, türetme ile eşdeğerdir ve pratik olarak kenarların algılanmasına yol açar. Öte yandan, bu işlemin zaman alanında gerçekleştirilmesi, yüksek geçiren filtre işlemleri ile frekans alanı tarafından yorumlanabilir. Bu nedenle, frekans alanındaki bu kenar algılamanın bir kısmı, Kenny, Mar-Hildorth gibi Gauss kenar dedektörleri ve kum azaltma gibi zaman alanındaki konveksiyona eşdeğer olan farklı yüksek geçiren filtreler tasarlanarak da yapılabilir.

Bu tezde, görüntü kenarı tanımayı geliştirmek için yapay sinir ağı yöntemleri kullanılmıştır. Başarılı bir şekilde geliştirilen önerilen yöntemlerde, simülasyon sonuçlarında gösterildiği gibi mevcut kenar bulma algoritması için daha iyi performans elde edilmektedir. Kenar görüntüleri, klasik maskelerden elde edilen kenar görüntülerinden daha düşük kalitededir ve bu maskeleri YSA kenar görüntülerine uygulamanın hiçbir etkisi yoktur. Önerilen yöntemde, görüntünün kenar algılamasını iyileştirmek için moment denetimli öğrenme yöntemiyle yapay sinir ağı kullandık. Burada, YSA kenar algılama görüntüsü yapay sinir ağı için bir öğretmen olarak kullanılmaktadır. Deneysel olarak elde edilen sonuçlar, sinir ağlarının kullanımının görüntü kenarında önemli bir gelişmeye sahip olduğunu göstermektedir. Bu tezde, optimizasyon problemlerinin hızının arttırılmasında ve çözümünde olumlu rol oynayan kombine optimizasyon yöntemlerini kullandık. Bu yöntemde öncelikle Sobel

veya Canny gibi klasik görüntü kenar algılama yöntemleri ile bir dizi yanıt oluşturulmakta ve daha sonra bu yanıtların bilgileri YSA için faydalı olacak şekilde elde edilmektedir. Ayrıca, önerilen hibrit algoritmamız, araştırma çalışmalarında hesaplama kapasitesini daha da azaltmak için kullanılabilir. Eski tarihi ve önemli resimlerin güncel akademik arşive de eklenmesinde yararlı olacaktır.

Gelecekte, görüntülerin kenarlarını iyileştirmek için Bat algoritması veya Crow arama algoritması gibi Metasezgisel yöntemleri kullanabiliriz. Ayrıca, iyi sonuçlar elde etmek için görüntülerin kenarlarını sınıflandırmak için derin öğrenme yöntemini kullanabiliriz.





## KAYNAKLAR

- Ab-BelKhair, A., Rahebi, J., & Nureddin, A. A. M. (2020). A study of deep neural network controller-based power quality improvement of hybrid PV/Wind systems by using smart inverter. *International Journal of Photoenergy*, <https://doi.org/10.1155/2020/8891469>.
- Abo-Zahhad, M., Gharieb, R. R., Ahmed, S. M., & Donkol, A. A. E. (2014). Edge detection with a preprocessing approach. *Journal of Signal and Information Processing*, 5, 123-134.
- Ahmed, S., Frikha, M., Hussein, T.D.H., & Rahebi, J. (2021a). Face recognition system using histograms of oriented gradients and convolutional neural network based on with particle swarm optimization. *2021 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, 1–5.
- Ahmed, S., Frikha, M., Hussein, T.D.H., & Rahebi, J. (2021b). Optimum feature selection with particle swarm optimization to face recognition system using gabor wavelet transform and deep learning. *BioMed Research International*, <https://doi.org/10.1155/2021/6621540>
- Albus, J.S. (1975). A new approach to manipulator control: the cerebellar model articulation controller (CMAC). *Journal of Dynamic Systems, Measurement and Control*, 97(3), 220-227.
- Alkım, Ö. (2005). TMS320 kullanılarak görüntüdeki kenarların belirlenmesi. Yüksek Lisans Tezi, *İstanbul Üniversitesi Fen Bilimleri Enstitüsü*, İstanbul.
- Arce, G. R. (2005). *Nonlinear Signal Processing: A Statistical Approach*. John Wiley & Sons.
- Argyle, E. (1971). Techniques for edge detection. *Proc. IEEE*, 59, 285-286.
- Arias-Castro, E., & Donoho, D. L. (2009). Does median filtering truly preserve edges better than linear filtering? *The Annals of Statistics*, 37(3), 1172-1206.
- Balq, A. A. A. (2019). Görüntü işleme tekniklerini kullanarak optik işaret tanıma sistemi. Doktora Tezi, *Kastamonu Üniversitesi Fen Bilimleri Enstitüsü*. Kastamonu.
- Bovik, A. C. (2010). *Handbook of Image and Video Processing*. Academic Press.
- Broomhead, D. S., & Lowe, D. (1988). *Radial Basis Functions, Multi-Variable Functional Interpolation and Adaptive Networks*. Royal Signals and Radar Establishment Memorandum 4148, Copyright @ Controller HMSO, London.
- Buksh, B., Routh, S., Mitra, P., Banik, S., Mallik, A., & Gupta, S. (2014). MATLAB based image editing and color detection. *International Journal of Scientific and Research Publications*, 4(1), 1-6.
- Canny, I. (1986). A computational approach to edge detection. *IEEE Trans. on Pattern*

- Analysis and Machine Intelligence*, 8(1), 679-698.
- Chidiac, H., & Ziou, D. (1999). Classification of image edges, Vision Interface'99, Troise-Rivieres, Canada, 17-24.
- Gelb, A., & Hines, T. (2011). Detection of edges from nonuniform Fourier data. *Journal of Fourier Analysis and Applications*, 17(6), 1152-1179.
- Gelb, A., & Song, G. (2014). A frame theoretic approach to the nonuniform fast Fourier transform. *SIAM Journal on Numerical Analysis*, 52(3), 1222-1242.
- Gonzalez, R. C., & Richard, E. E. (1993). Digital image processing, Addison-Wesley Publishing Company Inc., USA.
- Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). *Deep Learning* (Vol. 1). MIT press Cambridge.
- Gupta, B., & Shailendra, S. N. (2013). Image denoising with linear and non-linear filters: A review. *International Journal of computer Science Issues*, 10(6), 149-154.
- Gurney, K. (2018). *An Introduction to Neural Networks*. CRC press.
- Gültepe, Y., & Duru, A. E. (2018). Daily SO<sub>2</sub> air pollution prediction with the use of artificial neural network models. *International Journal of Computer Applications*, 181(34), 36-40.
- Gültepe, Y. (2019). Makine öğrenmesi algoritmaları ile hava kirliliği tahmini üzerine karşılaştırmalı bir değerlendirme. *Avrupa Bilim ve Teknoloji Dergisi*, 16, 8-15.
- Zarbega, T. S. A., & Gültepe, Y. (2020). Semantic segmentation of cell nuclei in breast cancer using convolutional neural network. *International Journal of Computer Applications*, 176(22), 1-8.
- Ha, P. S., & Shakeri, M. (2016). Licence plate automatic recognition based on edge detection. *Artificial Intelligence and Robotics*, 170-174.
- Hall, M. (2007). Smooth operator: Smoothing seismic interpretations and attributes, *The Leading Edge*, 26(1), 16-20.
- Hecht-Nielsen, R. (1990). *Neurocomputing*. Addison-Wesley, Reading, MA.
- Hecht-Nielsen, R. (1992). Theory of the backpropagation neural network. In *Neural networks for perception* (pp. 65-93). Elsevier.
- Hines, E. L., & Watson, P.A. (1983). Application of edge detection techniques to detection of the bright band in radar data. *Image and Vision Computing*, 1(4), 221-226.
- Huang, T., Yang, G., & Tang, G. (1979). A fast two-dimensional median filtering algorithm. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 27(1), 13-18.
- Jähne, B. (2013). *Digitale bildverarbeitung*. Springer-Verlag.

- Jiang, J., Trundle, P., & Ren, J. (2010). Medical image analysis with artificial neural networks. *Computerized Medical Imaging and Graphics*, 34: 617-631.
- Karlik, B., & Olgac, A. V. (2011). Performance analysis of various activation functions in generalized MLP architectures of neural networks. *International Journal of Artificial Intelligence and Expert Systems*, 1(4), 111-122.
- Kohonen, T. (1987). State of the art in neural computing. *IEEE First International Conference on Neural Networks*.
- Kröse, B., Krose, B., van der Smagt, P., & Smagt, P. (1993). *An Introduction to Neural Networks*. The University Amsterdam, Netherlands.
- Lakshmi, S., & Sankaranarayanan, V. (2010). A study of edge detection techniques for segmentation computing approaches, IJCA Special Issue on “Computer Aided Soft Computing Techniques for Imaging and Biomedical Applications.
- Lindeberg, T. (1998). Edge detection and ridge detection with automatic scale selection. *International Journal of Computer Vision*, (30), 117-156.
- Maini, R., & Aggarwal, H. (2009). Study and comparison of various image edge detection techniques. *International Journal of Image Processing*, 3(1).
- Miikkulainen, R., Liang, J., Meyerson, E., Rawal, A., Fink, D., Francon, O., vd. (2019). Evolving deep neural networks. In *Artificial intelligence in the age of neural networks and brain computing* (pp. 293-312). Elsevier.
- Moldovan, A., Caatron, A. & Andonie, R. (2020). Learning in feedforward neural networks accelerated by transfer entropy. *Entropy*, 22(102), 1-19.
- Muthukrishnan, R. (2011). Edge detection techniques for image segmentation. *International Journal of computer Science and Information Technology*, 3(6), 259-267.
- Nadernejad, E., Sharifzadeh, S., & Hassanpour, H. (2008). Edge detection techniques: Evaluations and comparisons. *Applied Mathematical Sciences*, 2(31), 1507-1520.
- Nagi, J., Ahmed, S. K., & Nagi, F. (2008). A MATLAB based face recognition system using image processing and neural networks. 4th International Colloquium on Signal Processing and Its Applications, 2: 83-88.
- Nayak, D. R. (2014). Edge detection using fuzzy double gradient morphology. *Bonfring International Journal of Advances in Image Processing*, 4(01), 1-4.
- Nureddin, A. A. M., Rahebi, J., & Ab-BelKhair, A. (2020). Power management controller for microgrid integration of hybrid PV/fuel cell system based on artificial deep neural network. *International Journal of Photoenergy*, <https://doi.org/10.1155/2020/8896412>.
- Rajasekaran, A., & Senthilkumar, P. (2014). Image denosing using median filter with edge detection using canny operator. *International Journal of Science and Research*, 30-34.

- Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6), 1137-1149.
- Rosenfeld, A., & Kak, A.C. (1982). *Digital Image Processing*. New York: Academic Press.
- Savant, S. (2014). A review on edge detection techniques for image segmentation. *International Journal of Computer Science and Information Technologies*, 5(4), 5898-5900.
- Sağıroğlu, Ş., Beşdok, E. & Erler, M. (2003). *Mühendislikte yapay zeka uygulamaları I: Yapay sinir ağları*, Kayseri: Ufuk Kitap Kirtasiye-Yayıncılık.
- Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61, 85-117.
- Senthilkumaran, N., & Rajesh, R. (2008). Edge detection techniques for image segmentation-A survey. *Proceedings of the International Conference on Managing Next Generation Software Application*, 749-760.
- Sethi, I. K., & Jain, A. K. (2014). *Artificial Neural Networks And Statistical Pattern Recognition: Old And New Connections*. Elsevier.
- Shah, A., Bangash, J. I., Khan, A. W., Ahmed, I., Khan, A., Khan, A., & Khan, A. (2020). Comparative analysis of median filter and its variants for removal of impulse noise from gray scale images. *Journal of King Saud University-Computer and Information Sciences*.
- Sharma, P., Diwakar, M., & Choudhary, S. (2012). Application of edge detection for brain tumor detection. *International Journal of Computer Applications*, 58(16), 21-25.
- Sharma, S., Sharma, S., & Athaiya, A. (2020). Activation functions in neural networks. *International Journal of Engineering Applied Sciences and Technology*, 4(12), 310-316.
- Sheinfeld, M., Levinson, S., & Orion, I. (2017). Highly accurate prediction of specific activity using deep learning. *Applied Radiation and Isotopes*, 130, 115-120.
- Shrivakshan, G. T., & Chandrasekar, C. (2012). A comparison of various edge detection techniques used in image processing. *International Journal of computer Science Issues*, 9(5), 269-276.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1), 1929-1958.
- Sudarshan, M., Paidi, G., & Gangashetty, S. (2011). Optimized edge detection algorithm for face recognition. *The International Conference on Security and Management*, Worldcomp.
- Sun, L., Liang, F., & Cui, W. (2021). Artificial neural network and its application

- research progress in chemical process. *ArXiv Preprint* ArXiv: 2110.09021.
- Tan, C. N. W. (1997). Artificial neural networks: A financial tool as applied in the Australian market, Doctorate Thesis, *Bond University*, Robina QLD 4226, Australia
- Teutsch, M., & Beyerer, J. (2012). Noise resistant gradient calculation and edge detection using local binary patterns. *Asian Conference on Computer Vision*, 1-14. Springer.
- Umbaugh, S. E. (2010). *Digital Image Processing and Analysis: Human and Computer Vision Applications with CVIP Tools*. CRC press.
- Rezai-Rad, G., & Aghababaei, M. (2006). Comparison of SUSAN and Sobel edge detection in MRI images for feature extraction. *Information and Communication Technologies*.
- Robert, J. S. (1989). *Digital image processing and computer vision*, John Wiley & Sons.
- Vincent, O. R. (2007). Optimization of network bandwidth using image compression. *Proceedings of the National Conference on advanced Data Computing Communications and Security*, Kadi, India.
- Wan, L., Zeiler, M., Zhang, S., Le Cun, Y., & Fergus, R. (2013). Regularization of neural networks using dropconnect. *International Conference on Machine Learning*, 1058-1066. PMLR.
- Wang, H. K., Ye, X. Q., & Gu, W. K. (2000). Training a neural network for moment based image edge detection. *Journal of Zhejinag University SCIENCE*, 1(4), 398-401.
- Xiang, Y., Wang, F., Wan, L., & You, H. (2017). SAR-PC: Edge detection in SAR images via an advanced phase congruency model. *Remote Sensing*, 9(3), 209.
- Xin, G., Ke, C., & Xiaoguang, H. (2012). An improved Canny edge detection algorithm for color images. *Industrial Informatics*, 113-117.
- Xu, B., Wang, N., Chen, T., & Li, M. (2015). Empirical evaluation of rectified activations in convolutional network. *ArXiv Preprint* ArXiv:1505.00853.
- Xueli, Y. (1996). *Neural Network and Example*. Learning Press of Railway of china, Beijing.
- Yim, Y. U., & Oh, S.-Y. (2003). Three-feature based automatic lane detection algorithm (TFALDA) for autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, 4(4), 219-225.
- Zhao, Z. C., Mehrdadi, B., & Freeman, J.M. (2000). A face image edge-detection method using fuzzy if-then rules. *Institute of Measurement and Control*, 33(10), 303-306.
- Zwaag, B. J. D., Slumo, K., & Spaanenburg, L. (2002). Analysis of neural networks

for edge detection, Proceedings of the ProRISC Workshop, Veldhoven, the Netherlands (2002), 580-586.

URL-1. Jacobs, D. Image Gradients, [www.cs.umd.edu/~djacobs/CMSC426/ImageGradients.pdf](http://www.cs.umd.edu/~djacobs/CMSC426/ImageGradients.pdf), Eriřim Tarihi; 01/012/2021.



# EKLER



## EK A-Diğer Görüntüler İçin Elde Edilen Sonuçlar

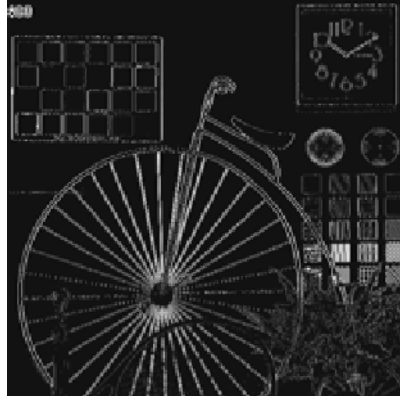
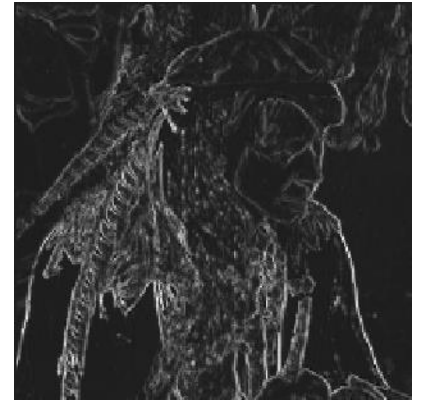
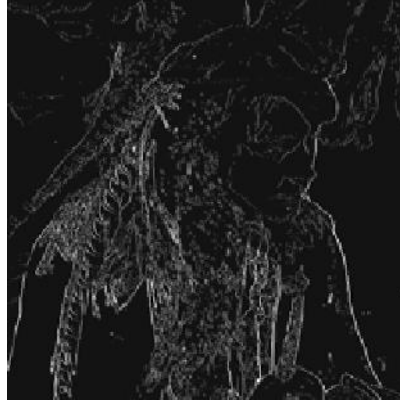
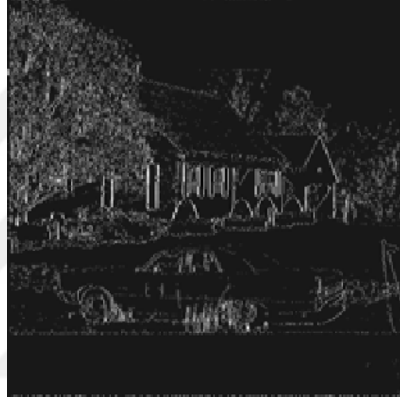
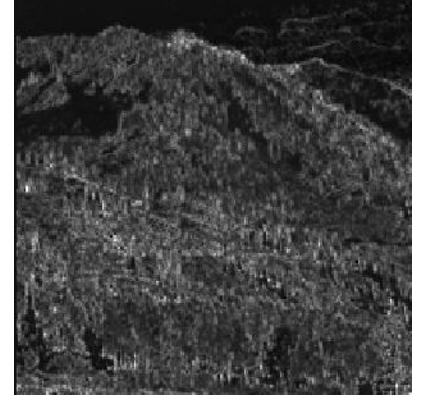
a) Orijinal görüntü



b) Sobel kenar algılama

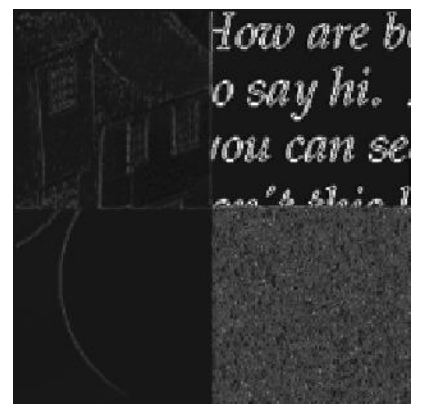
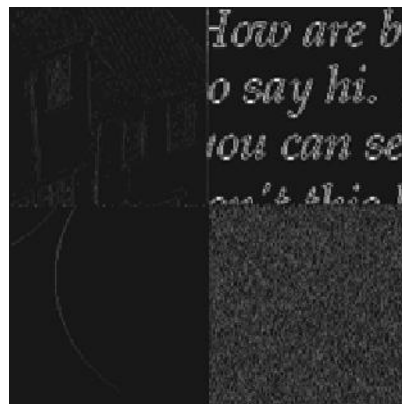
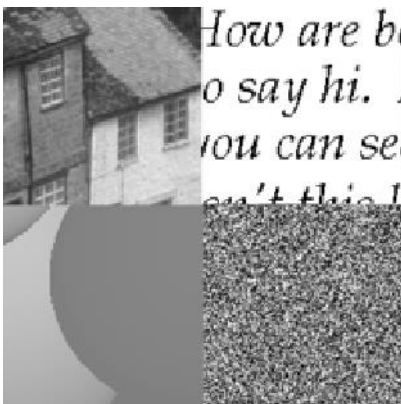
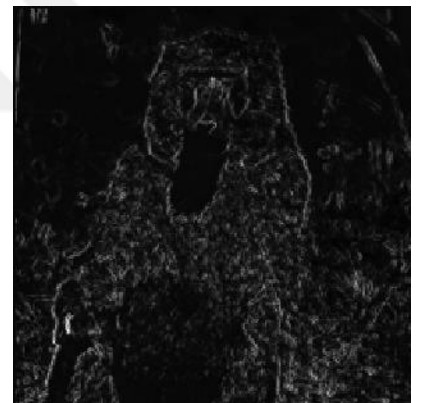
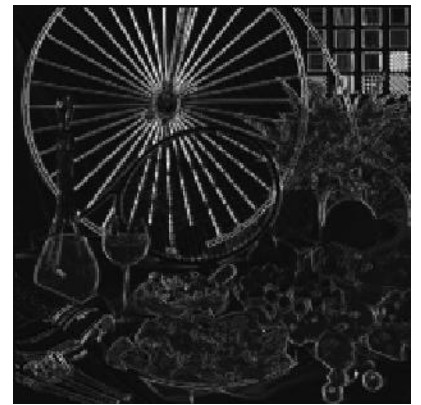
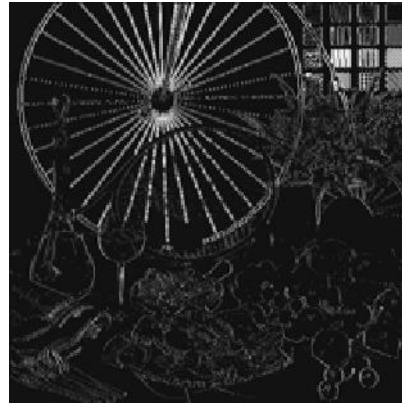


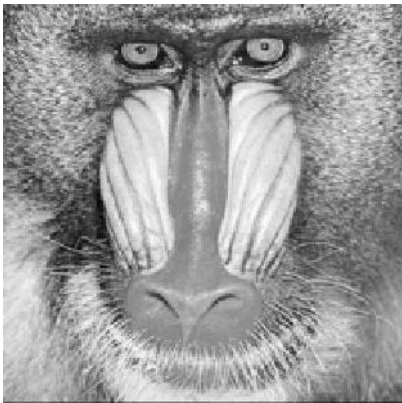
c) Kenar algılama yöntemi  
birleşik YSA

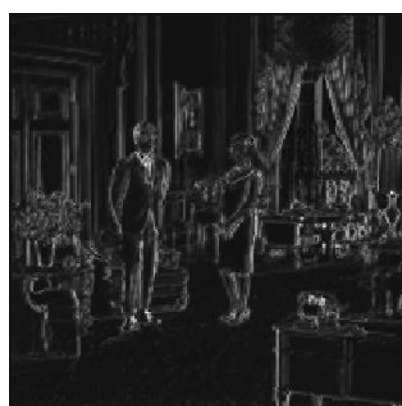
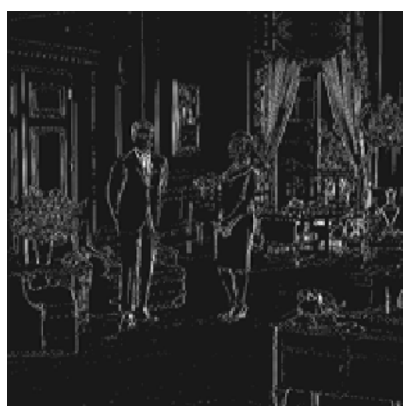
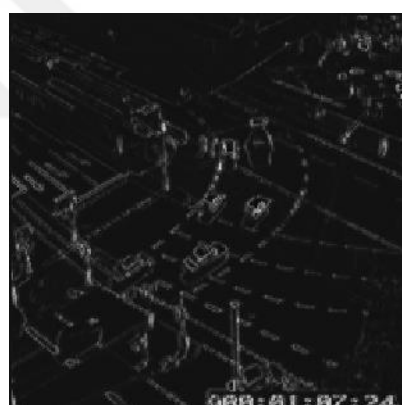
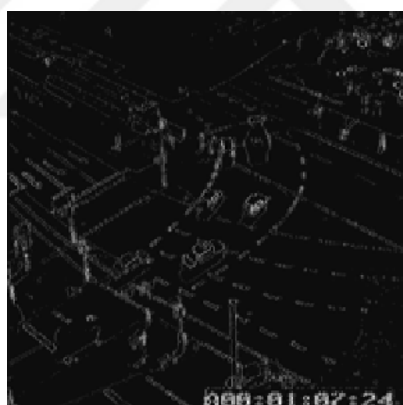
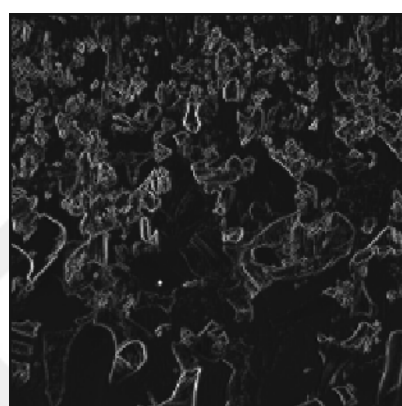
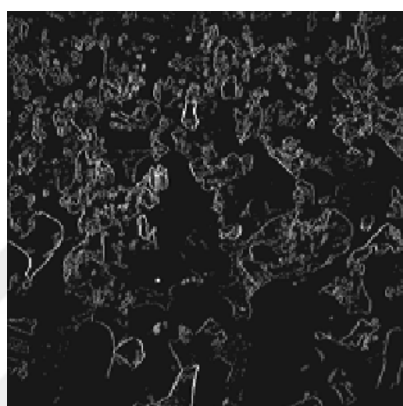
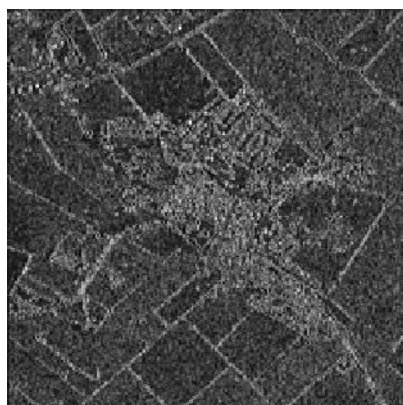




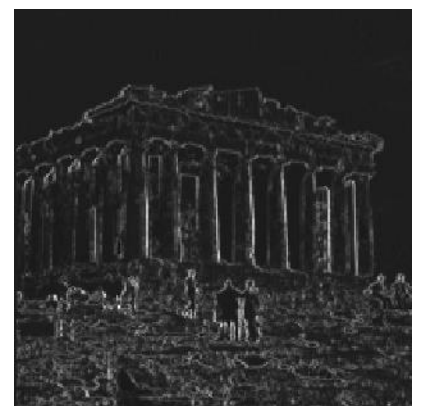
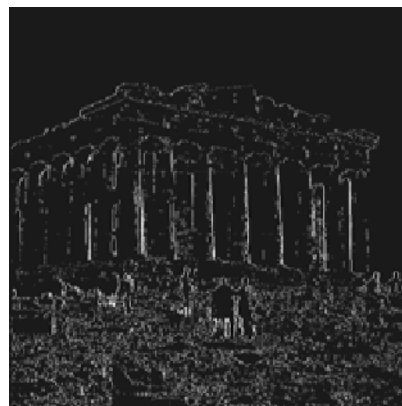
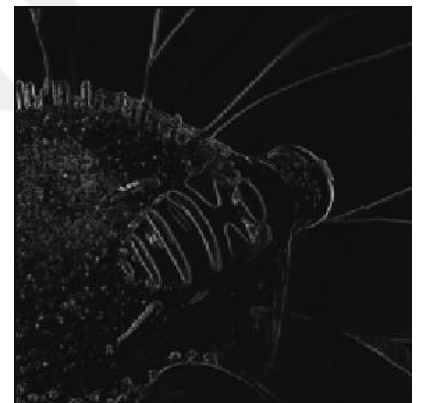
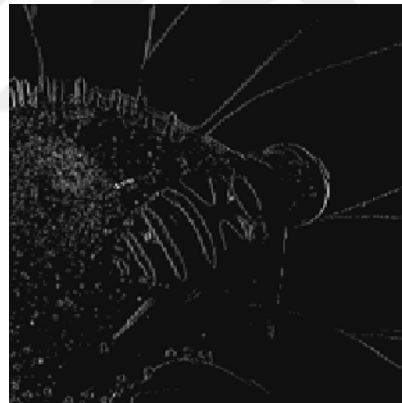
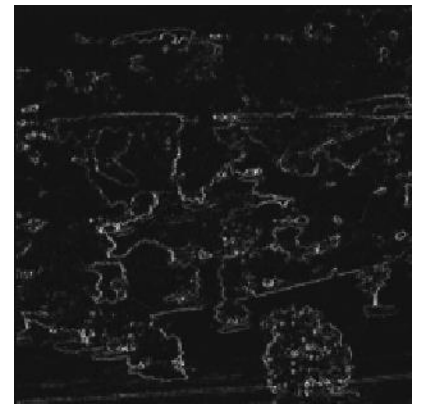
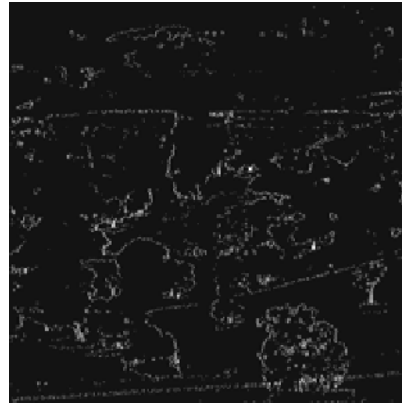


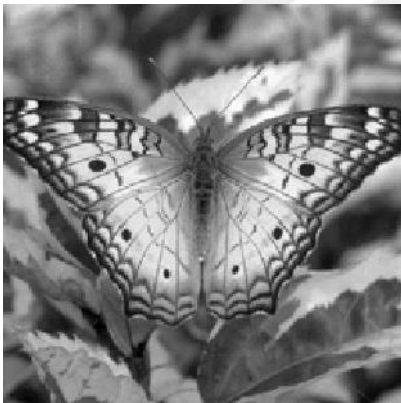


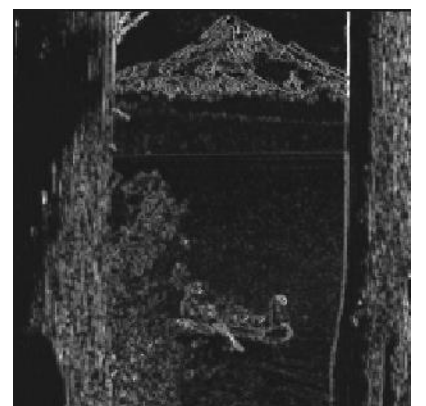
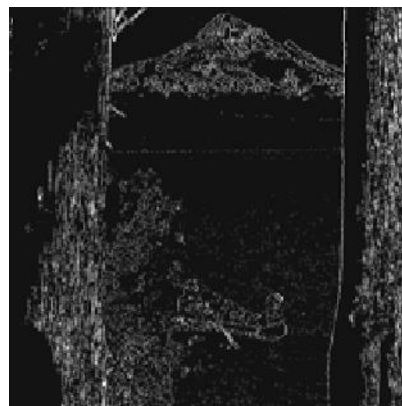
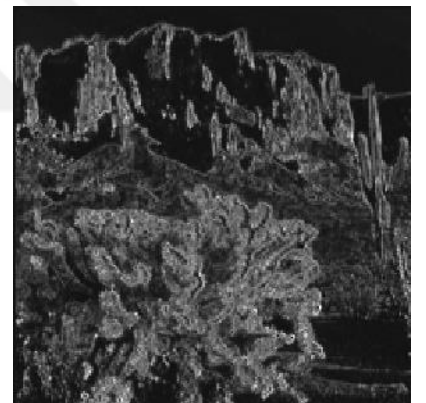
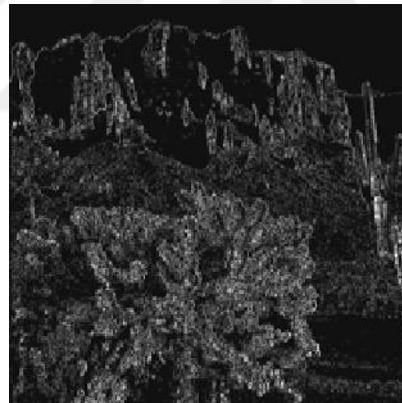
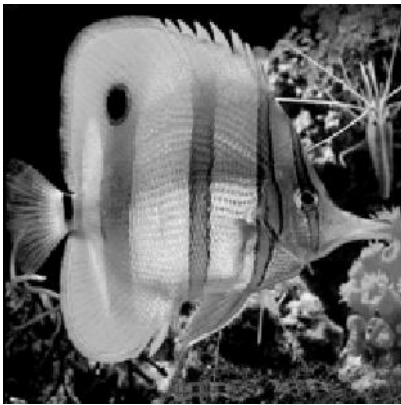
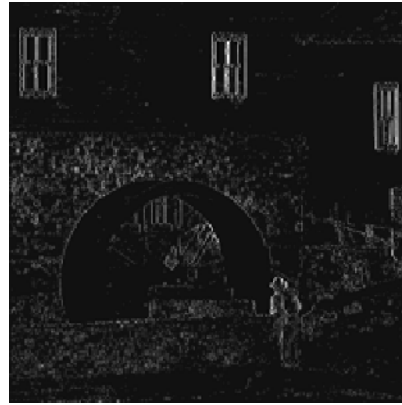


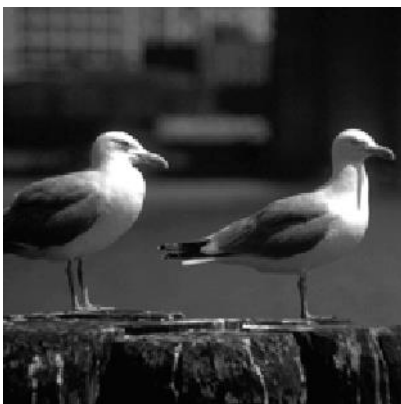
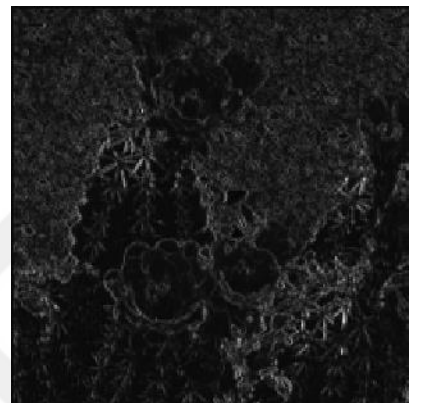
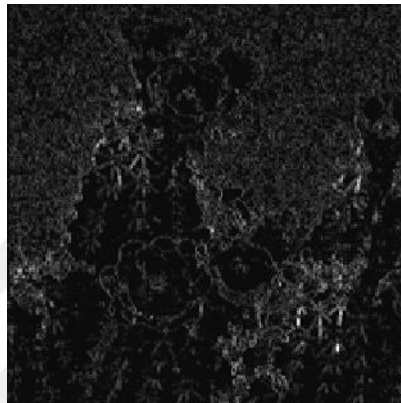




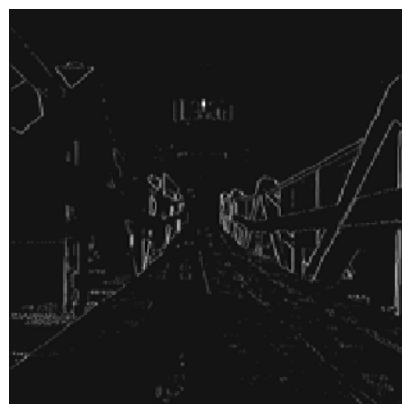
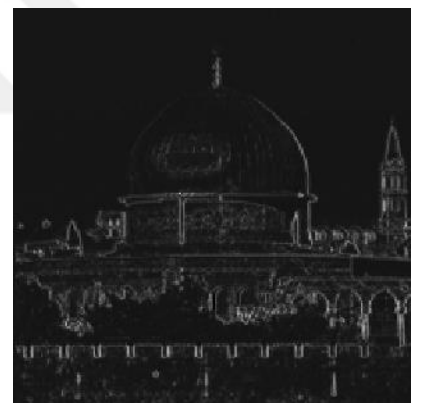
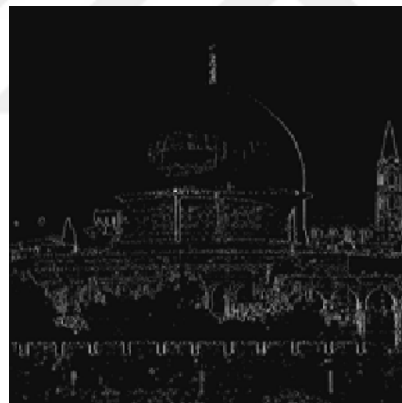
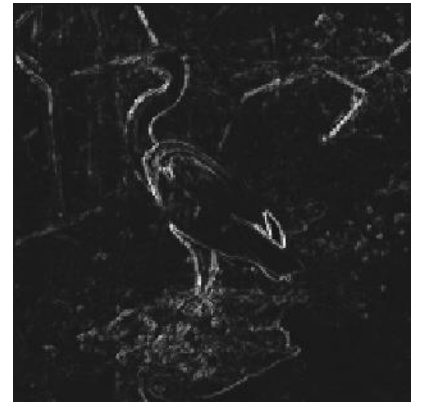
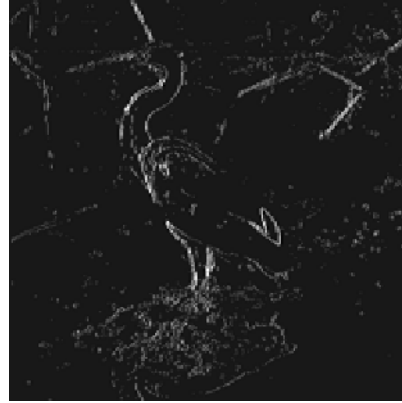


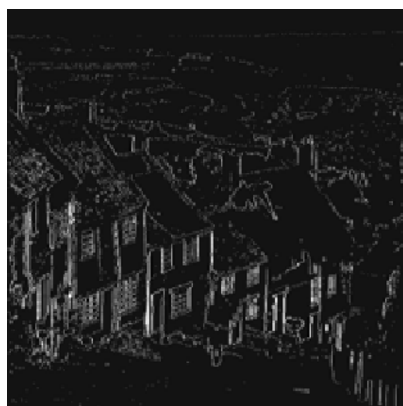
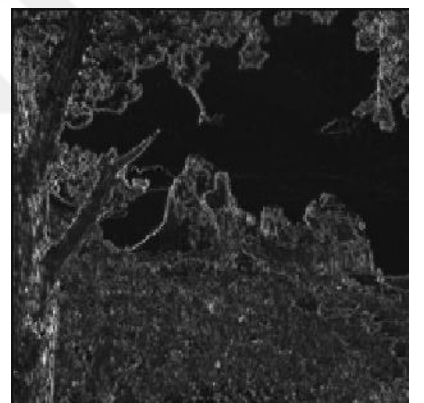
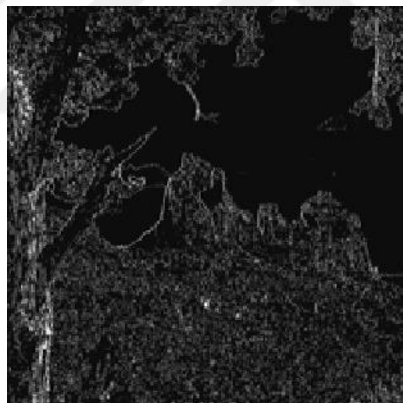


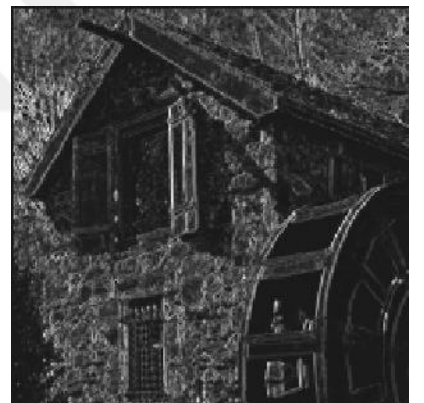


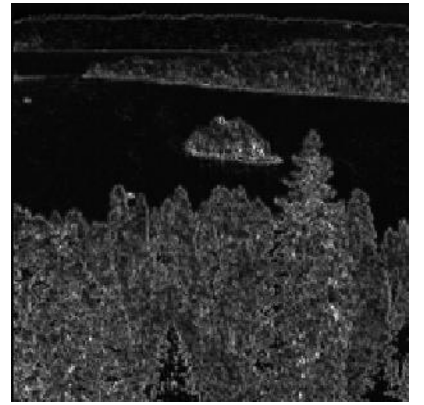
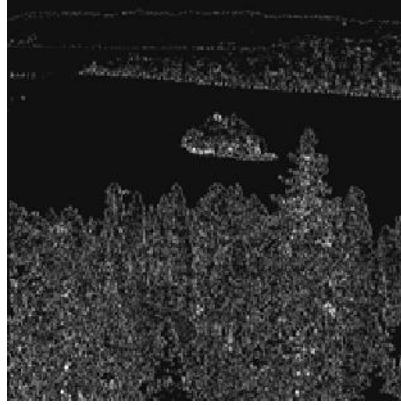












0

