

KOCAELİ ÜNİVERSİTESİ * FEN BİLİMLERİ ENSTİTÜSÜ

GÜRÜLTÜLÜ GÖRÜNTÜLERİN AKILLI BİR YÖNTEM İLE ONARIMI

YÜKSEK LİSANS TEZİ

Bilgisayar Müh. Sergey TSOY

Anabilim Dalı: Bilgisayar Mühendisliği

Danışman: Doç. Dr. Yaşar BECEREKLİ

KOCAELİ, 2011

KOCAELİ ÜNİVERSİTESİ * FEN BİLİMLERİ ENSTİTÜSÜ

GÜRÜLTÜLÜ GÖRÜNTÜLERİN AKILLI BİR YÖNTEM İLE ONARIMI

YÜKSEK LİSANS TEZİ
Bilgisayar Müh. Sergey TSOY

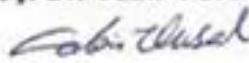
Tezin Enstitüye Verildiği Tarih: 10 Ocak 2011

Tezin Savunulduğu Tarih: 17 Şubat 2011

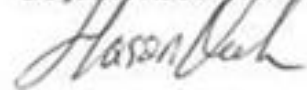
Tez Danışmanı
Doç. Dr. Yaşar BECEREKLİ



Üye
Doç. Dr. Cabir VURAL



Üye
Doç. Dr. Hasan OCAK



KOCAELİ, 2011

ÖNSÖZ VE TEŞEKKÜR

Gürültü giderme problemi en eski görüntü işleme problemlerinden olmasına rağmen, günümüzde de ilgi çeken, yeni cevapları arayan bir alandır.

Bu çalışmada dalgacık bölgesinde yapay sinir ağı yardımı ile gürültü giderme algoritması önerilmiştir.

Bu yüksek lisans tezinin ortaya çıkması sürecindeki önerileriyle ve gösterdiği özenle yardımı, emeği ve temel katkıları bulunan değerli danışmanım, hocam sayın Doç. Dr. Yaşar BECEREKLİ çok teşekkür ederim.

İÇİNDEKİLER

ÖNSÖZ VE TEŞEKKÜR	i
İÇİNDEKİLER	ii
ŞEKİLLER DİZİNİ	iii
TABLolar DİZİNİ	iv
ÖZET	v
İNGİLİZCE ÖZET	vi
1. GİRİŞ	1
2. TEMEL GÖRÜNTÜ İŞLEME KAVRAMLAR	4
2.1. Sayısal Görüntüler	4
2.2. Görüntü Onarımının Uygulanması	4
2.3. Gürültülü Görüntü Modeli	5
2.4. Görüntünün Gürültü Giderme Yöntemlerin Değerlendirme	8
3. YAPAY SİNİR AĞLARI	9
3.1. Biyolojik Nöron	10
3.2. YapayNöronun Yapısı ve Özellikleri	11
3.3. Sigmoid Transfer Fonksiyonu	12
3.4. YSA'nın Türleri ve Onların Özellikleri	13
3.5. İleri Beslemeli YSA	15
3.6. YSA'larınÖğretimi	15
3.7. Geri Yayılım Algoritması	17
4. DALGACIK DÖNÜŞÜMÜ	21
4.1. Fourier Dönüşümü	21
4.2. Kısa Süreli Fourier Dönüşümü	22
4.3. Dalgacık Dönüşümü	22
4.4. Hızlı Dalgacık Dönüşümü	24
4.5. Dalgacık Dönüşümünün KullanımAlanları	27
5. GÜRÜLTÜ GİDERME YÖNTEMLERİ	28
5.1. Klasik Gürültü Giderme Yöntemleri	28
5.2. Dalgacık Eşikleme	29
6. KENAR BULMA	32
7. GÜRÜLTÜ GİDERME ALGORİTMALARI	36
7.1. Görüntü Onarımı Amaçlı Yapay Sinir Ağ	36
7.2. Görüntü Hazırlama	39
7.3. YSA'larınYapıları	40
7.4. Kenar Korunan Gürültü Giderme Algoritması	41
8. SONUÇLAR	44
9. TARTIŞMA VE ÖNERİLER	50
KAYNAKLAR	51
ÖZGEÇMİŞ	53

ŞEKİLLER DİZİNİ

Şekil 2.1: Görüntü bozunum modeli	5
Şekil 2.2: Sadece toplamsal gürültüçeren görüntü bozunum modeli	6
Şekil 2.3: $\mu = 0$ ve $\sigma^2 = 1$ için Gauss (normal) dağılım.	7
Şekil 2.4: Gauss gürültülü Cameraman görüntüsü ($\mu = 0, \sigma^2 = 0.02$)	7
Şekil 2.5: İmpuls gürültülü Cameraman görüntüsü	8
Şekil 3.1: Biyolojik nöronlar arasındaki bağlantı (Ostrovsiy, 2002)	10
Şekil 3.2: Yapay nöronun yapısı (Ostrovsiy, 2002)	11
Şekil 3.3: YSA'ların ana yapıları. a - tam ilişkili, b - çok katlı, c - zayıf ilişkili (Ostrovsiy, 2002)	14
Şekil 3.4: İleri beslemeli yapay sinir ağının yapısı (Kruglov, 2002)	15
Şekil 3.5: YSA'nın öğretim işlemi (Ostrovsiy, 2002)	16
Şekil 3.6: Tek nöronlu YSA(Ostrovsiy, 2002)	18
Şekil 4.1: Bir işaretin farklı frekanslı sinüzoidal bileşenlerine ayrıştırılması.	21
Şekil 4.2: Bir işaret zaman uzayı gösteriliminin frekans uzayındaki eşdeğer	21
Şekil 4.3: Kısa zamanlı Fourier dönüşümü	22
Şekil 4.4: Dalgacık dönüşümü	23
Şekil 4.5: Sinüzoid ve dalgacık arasındaki fark.	23
Şekil 4.6: Dalgacık dönüşümü	24
Şekil 4.7: HDD'nün hesaplanması	26
Şekil 5.1: Katı ve yumuşak eşikleme	30
Şekil 7.1: Gürültü giderme işleminin şematik gösterilimi	38
Şekil 7.2: Pencerenin görüntünün (2, 2) konumundaki yerleşimi	39
Şekil 7.3: Merkezi Y_5 olan 3*3 boyutlu pencere	39
Şekil 7.4: YSA'nın yapısı	40
Şekil 7.5: Orijinal ve gürültülü görüntülere karşılık gelen yatay ve dikey alt bant katsayıları.	42
Şekil 7.6: Kenar korunan gürültü giderme algoritmasının şeması	43
Şekil 8.1: $\sigma^2 = 0.05$ gürültülü görüntünün iki yöntemle onarılması halleri. a) YSA+kenar bulma. PSNR=33.61. b) YSA. PSNR=31.69	45
Şekil 8.2: $\sigma^2 = 0.01$ gürültülü görüntünün iki yöntemle onarılması halleri. a) YSA+kenar bulma. PSNR=36.12. b) YSA. PSNR=36.94	45

TABLULAR DİZİNİ

Tablo 3.1: Aktivasyon fonksiyonları	12
Tablo 4.1. Ölçekleme ve dalgacıklar fonksiyonları	24
Tablo 5.1: Ortalama filtreleri	28
Tablo 5.2: Sıra istatistiği filtreleri	29
Tablo 7.1: Aktivasyon fonksiyonların karşılaştırılması	41
Tablo 8.1: Onarım sonuçları ($\sigma^2 = 0.01$)	44
Tablo 8.2: Onarım sonuçları ($\sigma^2 = 0.05$)	44
Tablo 8.3: Gürültü giderme yöntemlerin görsel görüntü kalitesi cinsinden karşılaştırılması	46
Tablo 8.4: Gürültü giderme yöntemlerin PSNR cinsinden karşılaştırılması	47
Tablo 8.5: 'Man' görüntüsünün gürültü giderme yöntemlerin görsel görüntü kalitesi cinsinden karşılaştırılması	48
Tablo 8.6: 'Man' görüntüsünün gürültü giderme yöntemlerin PSNR cinsinden karşılaştırılması	49

GÜRÜLTÜLÜ GÖRÜNTÜLERİN AKILLI BİR YÖNTEM İLE ONARIMI

Sergey TSOY

Anahtar Kelimeler: Gürültü giderme, dalgacık dönüşümü, yapay sinir ağları.

Özet: Bu tez çalışmasında, toplamsal beyaz Gauss gürültüsü ile bozulmuş görüntülerin yapay sinir ağlarının yardımı ile onarımı çalışılmıştır. Yapılan çalışmada görüntü verisi olarak, geliştirilen algoritmaların test edilmesi için siyah-beyaz görüntüler kullanılmıştır. Uygulamalarda geliştirilmiş algoritmaların klasik algoritmalara göre gürültü gidermesi daha başarılı olduğu gösterilmiştir.

Gürültü giderme işleminin uygulanması için bir görüntüler ilk önce dalgacık tabanına çevrilmiştir. Bir gürültülü görüntünün ve onun orijinal halinin dalgacık alt bantları yapay sinir ağlarının eğitimi için kullanılmıştır. Eğitilmiş yapay sinir ağlarının girişine diğer gürültülü görüntünün dalgacık alt bantları verilerek çıkışında bu alt bantların onarılmış halleri elde edilip, bu alt bantlara ters dalgacık dönüşümü uygulanarak onarılmış gürültüsüz görüntü elde edilmektedir.

Bu algoritma geliştirilerek, kenara duyarlı kenar bulma yapısı eklenmiştir. Çok ölçekli dalgacık kenar bulma tekniği ile gürültülü görüntünün kenarları bulup, kenarlara uygun olmayan dikey ve yatay katsayıların değerleri önceden seçilmiş bir parametre ile çarpılmaktadır. Bu yöntemle kenarlar da dikkate alınarak algoritma daha iyi hale getirilmiştir.

IMAGE DENOISING WITH AN INTELLIGENT METHOD

Sergey TSOY

Keywords: Image de-noising, wavelet transformation, neural networks.

Abstract: The purpose of this work is to perform an image de-noising of images corrupted by additive white Gaussian noise by means of neural networks. For testing of obtained algorithms grayscale images were used. Comparing with the classic image de-noising algorithms the proposed algorithm has given a better performance.

In order to apply de-noising algorithm images are first transferred to wavelet domain. The wavelet sub-bands of corrupted image and its original (without noise) version are used in learning of artificial neural networks. By giving of any another, different from ones, used in training, images sub-bands to the well trained neural networks, the restored versions of these sub-bands are obtained. In order to get a restored image, the obtained sub-bands are passed through inverse discrete wavelet transform.

To improve this algorithm, the edge adaptive de-noising stage was added. Using the multi-scale wavelet edge detection algorithm edges of corrupted image are detected and then wavelet coefficients that don't match image edges are multiplied by some previously selected parameter. Taking in account the edge information by means of this method the image de-noising algorithm has been improved.

1. GİRİŞ

Gürültü giderme problemi görüntü işlemede en eski problemlerden biridir(Kharlamov, Podlozhnyuk, 2007). Gürültü giderme algoritmaları genelde uzaysal veya frekanssal bölgelerde uygulanır. Weiner filtresi gibi bazı yöntemler hem frekanssal hem de dalgacık bölgesinde uygulanabilirler (Jacob, Martin, 2004).Gürültü olarak genelde beyaz Gauss gürültü kullanılmaktadır. Bunun sebebi beyaz gürültünün modelleme kolaylığı ve başka tür gürültülere bezerliğidir.

Bu çalışmada görüntünün kenarlarını koruyandalgacık bölgesinde bir gürültü giderme algoritması önerilmiştir. Dalgacık analizi ortaya çıktıktan sonra görüntü işleme problemleri kısmı dalgacık bölgesine uyarlanmıştır. Bunun sebebi görüntülerin düşük ve yüksek frekanslı bileşenlere ayırabilmedir. Bu çalışmada da onarım işlemi dalgacık bölgesinde gerçekleştirilmektedir.

Yapay sinir ağları sadece görüntü işleme ve gürültü giderme değil çok farklı alanda kullanılmıştır. YSA'ların çok farklı kabiliyetleri olmasına rağmen belirsiz bir fonksiyonun yaklaşımı en çok kullanılanlardan birisidir. Bu çalışmada gürültülü ve gürültüsüz görüntü katsayılarının arasındaki ilişkiyi bulmak için çok katmanlı YSA kullanılmıştır.

Genel olarak önerilen algoritmanın çalışma prensibi şöyledir: Gürültülü bir görüntünün kenarları bulunur. Sonra kenarların koordinatları ayrı bir matriste saklanıp görüntünün bir seviyeli ayırık dalgacık dönüşümü hesaplanır. Bu aşamada, görüntünün bir düşük frekanslı ve üç tane yüksek frekanslı bileşeni elde edilir: Yaklaşım katsayılar matrisi ve yatay-dikey ve diyagonal katsayılar matrisleri. Kenarları saklamak amacıyla kenarlara uygun olmayan dikey ve yatay katsayıların değerleri önceden seçilmiş bir ($0 < \alpha < 1$) parametre ile çarpılırlar. Bu aşamada ön işlem kısmı bitip gürültü giderme algoritması uygulanmaktadır.

Gürültü giderme aşaması tamamen YSA'larının yardımı ile gerçekleştirilir. Bu amaçla uygun şekilde öğretilmiş dört YSA kullanılır. Bu ağların çıkışlarına onarılmış dalgacık katsayılar verilmektedir. Bu katsayılarla ters dalgacık dönüşümü uygulandıktan sonra gürültülü görüntünün onarılmış versiyonu elde edilmektedir.

Bu çalışmada şukitap ve makalelerden yararlanılmıştır:

Görüntü işleme ve onun MATLAB'ta uygulanması ayrıntılı olarak R. Gonzales'in 'Sayısal görüntülerin işlenmesi'(Gonzales, 2005) ve 'MATLAB'ta sayısal görüntülerin işlenmesi' (Gonzales, 2006) kitaplarında verilmiştir. Bu kitaplarda görüntü işlemenin temelleri, görüntü onarımı, gürültü giderme, görüntü sıkıştırması ve bölütlenmesi, görüntülerle dalgacık bölgesinde çalışma ve nesne tanıma gibi konular bulunmaktadır.

Yapay sinir ağların kullanabilmesi için ilk önce MATLAB dokümantasyon kullanılmıştır. 'Neural Network Toolbox User's Guide' hem teorik hem de pratik bilgileri kapsamaktadır. Bu kaynakta ağların türleri, özellikleri ve bu ağların modellenmesi anlatılmıştır. Ayrıca standart ve evrensel ağların modellenmesiyle birlikte özel mimariye ve özel aktivasyon fonksiyonlara sahip olan ağların tasarlanması betimlenmiştir.

'Bulanık mantık ve yapay sinir ağları' (Kruglov, 2001) isimli kitapda YSA'ların çalışma mantığı ve öğretim aşaması ayrıntılı olarak anlatılır.

Dalgacık analizi hakkında genel bilgiler yine MATLAB dokümantasyondan alınmıştır. Dalgacık teorisinde en önemli çalışmalardan birisi: 'Çok çözünürlüklü işaretin ayrışması teorisi' (Mallat, 1989) Stephane Mallat tarafından 1989 yılında yapılmıştır. Bu çalışmada, yazar hızlı dalgacık dönüşümü algoritmasını anlatmaktadır. Bu algoritma hızlı Fourier dönüşümüne alternatif olup yüzlerce yeni çalışmaların temeli olmuştur. Çalışmanın ana fikri, bir fonksiyonu dalgacık dönüşüme çevirebilmek için bu fonksiyonun uygun dalgacık fonksiyonlarıyla konvolüsyona (convolution) tabi tutulması gerekir. Bu çalışmada dalgacık dönüşümü olarak Mallat'ın MATLAB ortamında gerçekleştirilen hızlı dalgacık dönüşümünü kullanılmaktadır.

'Dalgacıkların yardımı ile teklıkların bulunması ve işlenmesi' (Mallat, 1992) isimli makalede, sürekli dalgacık dönüşümün matematiksel temelleri, dalgacık kenar bulma tekniği ve Lipschitz üsü (exponent) kullanan bir gürültü giderme yöntemi anlatılmıştır.

'Yumuşak eşikleme ile bir görüntünün gürültü gidermesi'(Donoho, 1995) D. Donoho tarafından 1995'te önerilmiştir. Bu çalışmada eşikleme yöntemi kullanarak belli bir eşik altındaki değerleri sıfırlayarak bir gürültü giderme yöntemi anlatılır. Bu makalenin yayınlanmasından sonra dalgacık bölgesinde gürültü giderme

yöntemlerinin sayısı oldukça hızlı artmıştır.Bu yöntemlerin çoğu ya bu yöntemin iyileştirilmesi yada başka yöntemlerle birleştirilmesi olarak önerilmiştir.

'YSA'ya dayalı doğrusal olmayan bir filtre ile dalgacık bölgesinde gürültü gidermesi'(Zhang,2005) isimli makalede önerilen algoritma bu çalışmada bir esas olarak kullanılmıştır.Önerilmiş yöntemde gürültülü görüntünün dalgacık bölgesine dönüştürdükten sonra dalgacık katsayıları YSA'ların yardımı ile temizlenirler. Gürültü giderme aşamasından sonra elde edilmiş katsayıları ters dalgacık dönüşümünden geçirilerek onarılmış görüntü elde edilir.

'Dalgacık bir kenar bulma yöntemidir'(Li, 2003) bir yüksek lisans tez çalışmasıdır. Yazar tarafından gürültülü görüntünün bir kenar bulma tekniğinin iyileştirilmesi (Mallat, 1992)için önerilmiştir.Bu tez çalışmada bu algoritma kenar saklama amacı ile kullanılmaktadır.

Bu çalışma aşağıdaki bölümlerden oluşmuştur.

Bölüm 2'de sayısal görüntü, gürültü ve gürültü giderme gibi genel kavramlar açıklanmıştır.

3. bölümde YSA'lar hakkında uygulama alanları, türleri, biyolojik ve yapay nöronların yapıları gibi bazı bilgiler verilmiştir.Tek nöronlu ağın öğrenme örneği ile geri yayılım algoritması incelenmiştir.

4. bölüm dalgacık dönüşümü hakkında temel bilgileri içermektedir.Hızlı dalgacık dönüşümü algoritması incelenmiştir.

5. bölümde eşikleme gibi bazı gürültü giderme yöntemleri verilmiştir.

6. bölümde dalgacık kenar bulma tekniği tartışılmıştır.

Son bölümde önerilen gürültü giderme algoritması açıklanmıştır. Algoritmanın çalışma sonuçları ve onların karşılaştırması verilmiştir.

2. TEMEL GÖRÜNTÜ İŞLEME KAVRAMLAR

2.1. Sayısal Görüntüler

Renkli olmayan her hangi görüntü iki boyutlu $f(x, y)$ fonksiyonu olarak temsil edilebilir. Burada x ve y yuzamsal koordinatlarıdır. Her x ve y için $f(x, y)$ fonksiyonun değeri bu koordinattaki noktanın parlaklık değerine eşittir. Bir görüntünün x ve y koordinatları sürekli olabilirler. O zaman bu görüntüyü ifade eden fonksiyon değerleri aynı şekilde sürekli olabilir. Bu tür görüntüleri sayısal hale çevirebilmek için hem koordinatların hem de fonksiyon örneklenmesi gereklidir. Sayısal bir görüntü matris şeklinde ifade edilebilir. Eğer matris M satıra ve N sütüne sahipse o zaman bu matrise $M \times N$ boyutlu matris denir ve şu şekilde gösterilebilir:

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

Matrisinin her bir elemana piksel denir. Genellikle renksiz görüntüler için pikselin değeri 0'dan 255'e kadar değişir.

2.2. Görüntü Onarımının Uygulanması

Görüntü onarımı problemi, ilk olarak astronomik görüntü işleme alanında ortaya çıkmıştır. Hızlı değişen atmosfer koşullarından dolayı yeryüzü görüntüsü, algılama sistemleriyle algılandıktan sonra bazen bulanık şekilde elde edilmiştir. Dünya'nın ve başka gezegenlerin görüntülerini algılayan ilkkameraların çekme hızı oldukça yavaştı. Ayrıca bu kameraların bulunduğu uzay araçlarının hızları yüksek olduğundan, elde edilen görüntüler hareket bulanıklıkları ile bozulmuş olarak oluşmaktaydı. Ayrıca, görüntüler farklı gürültü türlerinden etkilenmektedir. Mesela ışık kaynaklarının zayıf olmasından bazı görüntüler işaret bağılı Poisson gürültüsünden etkilenmektedir (Banham, 1997). Gürültülerin en yaygın türü Gauss gürültüsüdür. Gauss gürültü genelde elektronik parçaların çalışmasından

dolayıortaya çıkmaktadır. Buna hem algılama aletlerinin hem de veri aktarma mekanizmasındaki elemanların çalışması neden olabilir(Sahraeian 2007).Astronomik görüntülerin işlenmesi, görüntü onarımının temel uygulama alanlardan birisidir(White, 1991), (Nunez, 1995).

Aynı zamanda görüntü onarımı tıp sektöründe de başarıyla uygulanmıştır(Kwon, 2010).Toplamsal Poisson gürültülü, göğüs manyetik rezonans röntgen görüntülerimim temizlenmesi örnek olarak verilebilir. (Kwon, 2010)(Zhang, 2008)(Slump, 1992).

2.3. Gürültülü Görüntü Modeli

Klasik bozma modeli içerisinde iki kavramı ihtiva eder: Bulanık ve toplamsal gürültülü bir görüntünün matematiksel modeli aşağıdaki gibi verilebilir.(Paik, 1992):

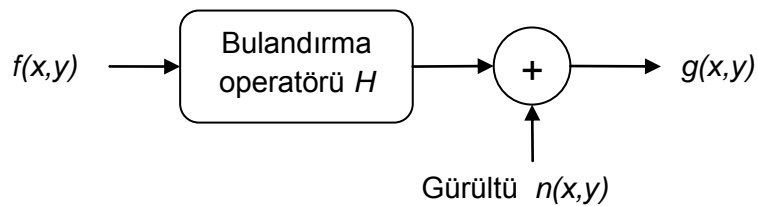
$$g(x, y) = \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} h(x, y; k, l) f(x, y) + n(x, y) \quad (2.2)$$

Burada $f(x, y)$ orijinal ve $g(x, y)$ bozulmuş $M*N$ boyutlu görüntüleri ifade etmektedir. $n(x, y)$ ise gürültüdür. Kaynaklarda gürültü olarak genelde sıfır ortalamalı Gauss dağılımına sahip beyaz gürültü kullanılır. $h(x, y; k, l)$ görüntü sisteminin iki boyutlu nokta yayılım fonksiyonu(point-spread function)dur. Bulanıklaştırma operatörü (h) ötelemede bağımsız ise, denklem (2.2) aşağıdaki gibi verilir.

$$g(x, y) = h(x, y) * f(x, y) + n(x, y) \quad (2.3)$$

Burada *sembolü katlama(konvolüsyon) operatörüdür.

Görüntü bozunum modelişematik olarak Şekil2,1'de gösterilmiştir:

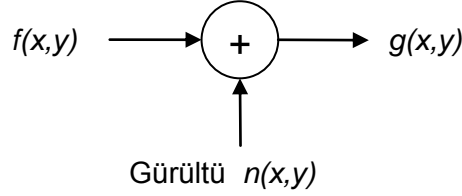


Şekil 2.1: Görüntü bozunum modeli

Bu tezin hedefi sadece gürültülü görüntülerin onarımıdır. Böyle görüntülerin matematiksel modeli:

$$g(x,y) = f(x,y) + n(x,y) \quad (2.4)$$

Şeklinde alıp şematik olarak aşağıda gösterilmiştir:

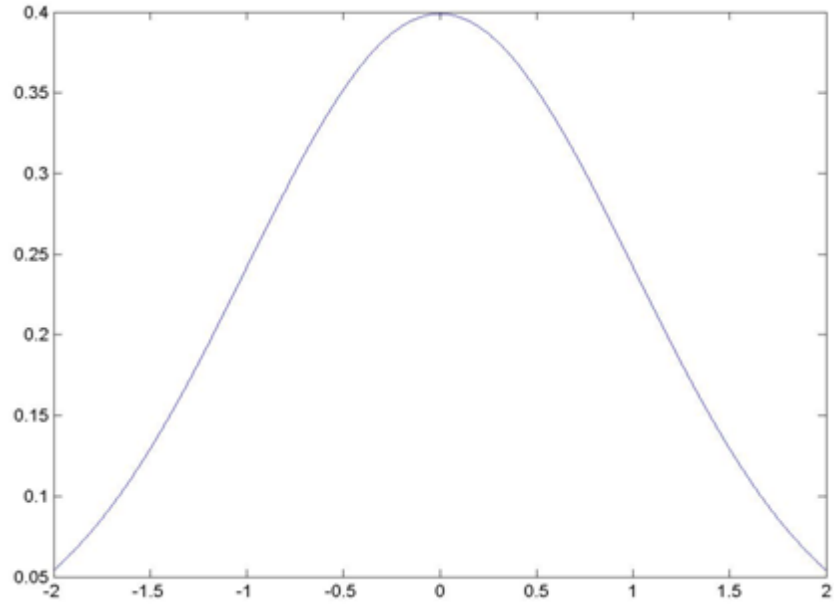


Şekil 2.2: Sadece toplamsal gürültü içeren görüntü bozunum modeli

Yukarıda belirttiği gibi gürültü genelde görüntünün algılandığında yada taşındığında eklenmektedir. Bir görüntünün gürültüsünün giderme problemlerinde genellikle Gauss dağılımına sahip olduğu varsayılmaktadır. Gauss gürültüsünün olasılık dağılımı aşağıdaki formül ile hesaplanır:

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(z-\mu)^2/2\sigma^2} \quad (2.5)$$

Burada z - parlaklık değeri, μ rastgele z değerinin ortalama değeri, σ standart sapmasıdır. Standart sapmasının karesine varyans denir. Bu fonksiyonun grafiği Şekil 2.3' de ve Gauss gürültüsü eklenmiş bir görüntü Şekil 2.4 verilmiştir.



Şekil 2.3: $\mu = 0$ ve $\sigma^2 = 1$ için Gauss (normal) dağılım.



Şekil 2.4: Gauss gürültülü Cameraman görüntüsü ($\mu = 0, \sigma^2 = 0.02$)

Tuz ve biber gürültüsü daha az kullanılan bir gürültü türüdür. Bu gürültüye impuls gürültüsü de denilmektedir. Elektrik şebekesindeki hatalardan dolayı oluşur (Kalinkina, 2005). Görüntüde beyaz veya siyah noktalar olarak görünmektedir. Bu tür gürültünün giderilmesi için ortanca filtresi kullanılmaktadır. Şekil 2.5'te İmpuls gürültülü görüntü verilmiştir.



Şekil 2.5: İmpuls gürültülü Cameraman görüntüsü

2.4. Görüntünün Gürültü Giderme Yöntemlerin Değerlendirme

Farklı gürültü giderme yöntemlerin değerlendirme genelde şu şekilde gerçekleştirilir: Orijinal görüntüye bir gürültü eklenmektedir. Elde edilmiş görüntü bir gürültü giderme yöntemi ile onarılarak elde edilen görüntü orijinal görüntü ile karşılaştırılmaktadır.

Orijinal ve gürültülü görüntünün karşılaştırılması genelde PSNR (peak signal-to-noise ratio) ölçütü kullanılır. PSNR yöntemi aşağıdaki formül ile hesaplanır:

$$PSNR(x, y) = 20 * \log_{10} \frac{255}{\sqrt{\frac{1}{N} \sum_{i=1..N} d(x_i, y_i)^2}} \quad (2.6)$$

Burada x_i, y_i orijinal ve onarılmış görüntülerin i . pikselleri, N her görüntüdeki toplam piksel sayısı, $d(x_i, y_i)$ orijinal ve onarılmış görüntülerin piksel değerleri arasındaki farkı belirtmektedir. Onarılmış görüntü orijinal görüntüye ne kadar yakınsa, PSNR değeri ve algoritmanın verimliliği o kadar yüksektir.

PSNR ve başka ölçütleri görüntüler arasında karesel ortalama hatayı hesaplamaktadır. Bu nedenle PSNR ölçütü insan görme sistemiyle uyumlu değildir.

3. YAPAY SİNİR AĞLARI

Yapay sinir ağları (YSA), insan beyinin çalışmayla ilgili basit biyolojik işlemleri modelleyen bir hesaplama sistemidir. Olumlu veya olumsuz etkileri analiz ederek YSA'lar öğrenebilen sistemlerdir. YSA'larda temel eleman olarak sinir hücresi kullanılır.

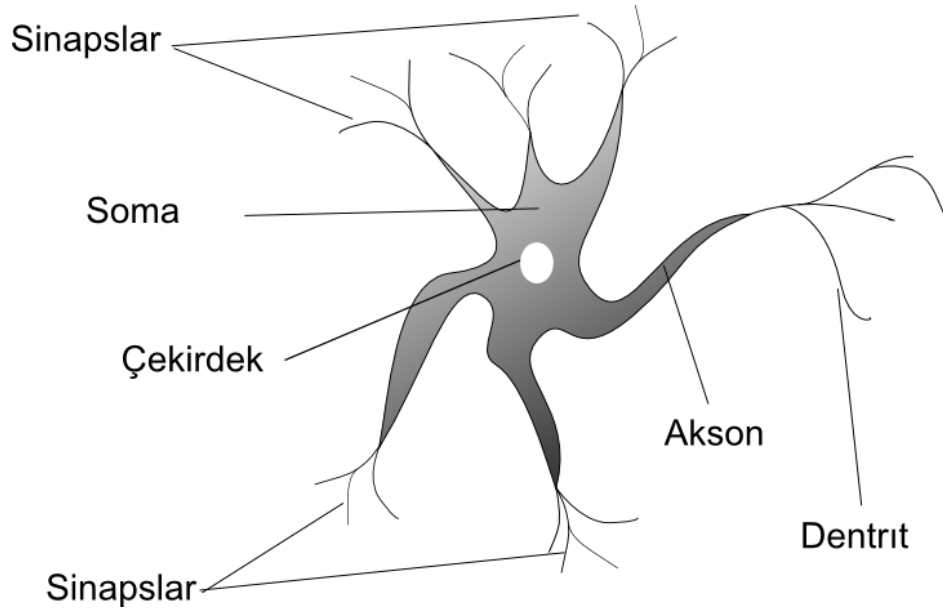
YSA'ların yardımıyla çözülebilen problemler(Kruglov, 2001):

- Sınıflandırma:Vektör şeklinde gelen giriş verilerinin(el yazması harf yada konuşma sinyali) önceden belirlenmiş sınıflara ait olup olmadığı problemini çözer. Uygulamalar: metin tanınması, ses tanınması, kan hücrelerin sınıflandırması v.b.
- Kümeleme (öğretmensiz sınıflandırma): Bu tür problemlerde kümeleme işlemi öğrenme örnekleri kullanmadan gerçekleştiriyor. Algoritma verilerinin benzerliğine dayanıyor ve bir birine yakın örnekleri bir kümeye koyarak işe başlamaktadır.
- Fonksiyonların yaklaşımı: Bilinmeyen bir fonksiyonla üretilmiş giriş ve çıkış değerleri $((x_1, y_1), (x_2, y_2), \dots, (x_n, y_n))$ göre bilinmeyen $F(n)$ fonksiyonunuyaklaşık modellemesidir. Fonksiyonların yaklaşımımühendislik ve bilimsel modelleme problemlerde çok kullanılır.
- Tahmin:Art arda gelen zaman aralıklarına (t_1, t_2, \dots, t_k) ait n tane ayrık değer $\{y(t_1), y(t_2), \dots, y(t_k)\}$ vardır. Hedef $y(t_k)$ değerinin tahmin edilmesidir.
- Optimizasyon: Optimizasyon algoritmanın hedefi,belirli kısıtlar altında uygun bir ölçüt fonksiyonunun maksimumu yada minimumunu bulmaktır. Genellikle borsave hava tahmini gibi uygulamalarda kullanılır.Çok sayıdaki matematiksel, istatistiksel, tıbbi ve ekonomiksel problemler optimizasyon problemi olarak düşünebilir.
- Çağrışımlı bellek: Belleğinin içeriği tamamlanmamış ya da hatalı içeriğine göre çağırılabilir. Bu tür uygulamalar, mültimedya veri tabanlarının tasarımında kullanılabilir.

3.1. Biyolojik Nöron

İnsan beyni ve sinir sistemi nöronlardan ve nöronları bağlayan sinir uçlarından oluşuyor. Sinir uçları nöronlarının arasında elektrik sinyalleri gönderirler. Nöron, bilgileri işleyen bir biyolojik hücredir. Gövdeden ve sinir uçları uzantılarından oluşur. Uzantıların iki türü vardır: dentritler ve aksonlar. Hücre birkaç tane dentrite sahip olabilir ve bu dentritler üzerinden itkiyi (impuls) alır. Aynı zamanda hücre sadece bir tane aksona (çıkış) sahip olup bu akson üzerinden itkiyi gönderebilir (Osovskiy, 2002).

Hücresinin gövdesi kalıtsal bilgileri kapsayan çekirdekte ve nöron için gerekli malzemeleri üreten plazmadan oluşur. Ücündeki akson birkaç ucla dallanır. Uçların ucunda ise itki gücünden etkilenen sinapslar vardır. Nöronların yapısı Şekil 3.1 gösterilmektedir (Bu şekil (Osovskiy, 2002) kaynaktan çizilmiştir).

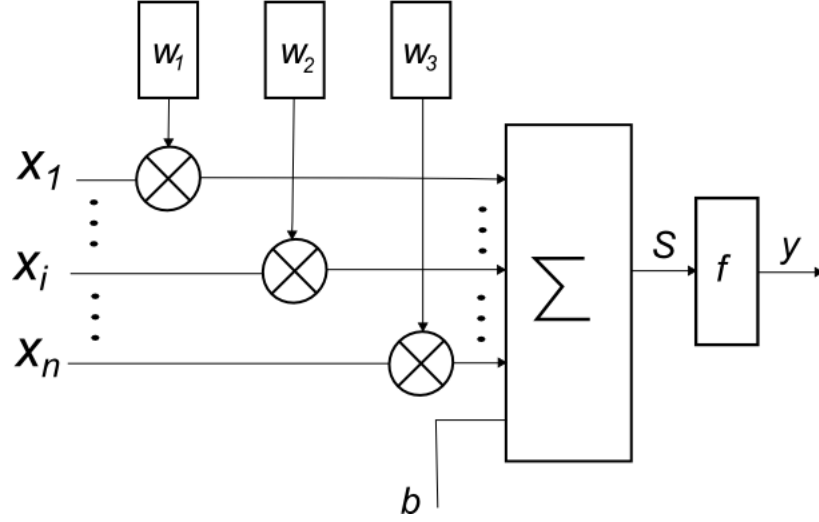


Şekil 3.1: Biyolojik nöronlar arasındaki bağlantı (Ostrovsiy, 2002)

Sinaps bir nöronun aksonu başka bir nöronun dentriti ile bağlanan bir düğüm olarak düşünülebilir. Bir itki sinapsın ucuna ulaştığında bu sinapsla bağlanmış nöronu etkileyen kimyasal maddeler kurtulur. Sinapsın tipine göre bu maddelerden etkilenen nöronun elektrik itkileri üretme olasılığı artar ya da azalır. Sinapslar katıldığı işlemlerin aktifliğine göre öğrenilebilirler. Başlangıçta bağılı bir bellek olarak çalışırlar. Sinapsların ağırlıkları zamanla değişip ilgili nöronun davranışına etki yaparlar.

3.2. YapayNöronun Yapısı ve Özellikleri

Nöron – YSA'nın bileşik elamanıdır. Şekil 3.2'de nöronun yapısı gösterilmektedir(Rutkovskaya, 2006).Yapay nöron; sinaps, toplama ve transfer elemanı içerir. Sinapslar nöronlar arasındaki bağlantıyı gerçekleştirirler ve giriş sinyalinin bağlantı gücü olan bir sayıyla çarpırlar. Bu tür sayılara sinapsın ağırlığı denir.



Şekil 3.2: Yapay nöronun yapısı (Ostrovsiy, 2002)

Toplama elemanı başka nöronların sinapslarından ve her hangi dış sinyal kaynaklardan gelen sinyalleri toplar. Transfer elemanı doğrusal olmayan bir fonksiyonu gerçekleştirir.Bu fonksiyon toplama elemanının çıkışını alırki bu fonksiyona transfer ya da etkinleştirme fonksiyonu denir. Nöronun matematiksel modeli şu şekilde verilir:

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

$$y = f(S) \quad (3.2)$$

Buradaw_i – sinapsın ağırlığı (i=1, ...,n);b- kayma değeri; S – toplamanın sonucu; x_i– giriş sinyali (i=1, ...,n);y- nöronun çıkış sinyali;n – nöron girişlerin sayısı;f- transfer (aktivasyon) fonksiyonudur.

Çıkış y transfer fonksiyonu ile bulunur.Şu halde bir nöron kendi ağırlıklarıyla ve transfer fonksiyonu ile tanımlanabilir. Bir vektörü giriş olarak alan transfer fonksiyonu çıkışta herhangi bir sayısını verir. Bazı transfer fonksiyonların (etkinleştirme fonksiyonları) matematiksel ifadeleri Tablo 3.1 verildi.

Tablo 3.1: Aktivasyon fonksiyonları

İsim	Matematik İfadesi	Değer Alanı
Keskin Sınırlayıcı	$f(S) = \begin{cases} 0, S < \theta \\ 1, S \geq \theta \end{cases}$	(0,1)
İşaretsel	$f(S) = \begin{cases} 1, S > 0 \\ -1, S \leq 0 \end{cases}$	(-1,1)
Sigmoid	$f(S) = \frac{1}{1 + e^{-aS}}$	(0,1)
Yarı-Doğrusal	$f(S) = \begin{cases} S, S > 0 \\ 0, S \leq 0 \end{cases}$	(0, ∞)
Doğrusal	$f(S) = S$	(-∞,∞)
Gaussff	$f(S) = e^{-S^2}$	(0,1)
Hiperbolik Tanjant (Sigmoid)	$f(S) = \frac{e^S - e^{-S}}{e^S + e^{-S}}$	(-1,1)
Üçgensel	$f(S) = \begin{cases} 1 - S , S \leq 1 \\ 0, S > 1 \end{cases}$	(0,1)

3.3. Sigmoid Transfer Fonksiyonu

Sigmoid en çok kullanılan transfer fonksiyonlardan birisidir(Haykin, 1999).

$$f(S) = \frac{1}{1 + e^{-aS}} \quad (3.3)$$

a parametresi azaltılınca f sigmoid fonksiyonu daha eğik olur.a sıfıra eşitse fonksiyon 0.5 değerindeki yatay çizgiye dönüşür.a artınca sigmoid θ eşikli keskin sınırlayıcı fonksiyona yaklaşır.Sigmoid şu özelliklere sahiptir: Sigmoidin türevi oldukça basittir.

$$f'(S) = af(S)(1 - f(S)) \quad (3.4)$$

Fonksiyon bütün apsis ekseninde ayırt edilebilir. Bu özellik bazı öğretim algoritmalarında kullanılır. Zayıf sinyalleri güçlü sinyallerden daha çok artırır ve güçlü sinyalleri doyurmaktan önler.

3.4. YSA'nın Türleri ve Onların Özellikleri

Her hangi bir YSA'nın görevi giriş X vektörünün çıkış Y vektörüne dönüştürmesidir. Bu dönüştürme ağın ağırlığına dayalıdır.

YSA'nın yardımıyla bir problemin çözülmesi:

1. YSA mimarisinin (yapısının) seçilmesi.
 - Nöron yapısının seçilmesi (Giriş sayısı, transfer fonksiyon).
 - Nöronlar arasında bağlantıları belirtmesi.
 - Ağın girişinin ve çıkışının belirtmesi.
2. YSA'nın öğrenmesi. Bu aşamanın amacı ağın çalışmasını doğru şekilde sağlayan nöronlarının ağırlıklarını bulmaktır. Nöron ağırlıklarının gerekli şekilde ayarlanması farklı YSA yapıları için farklı öğrenme algoritmalarını gerektirir.

Nöronlar ağın yapısında görevlerine göre üç gruba ayrılabilirler (Kruglov, 2002).

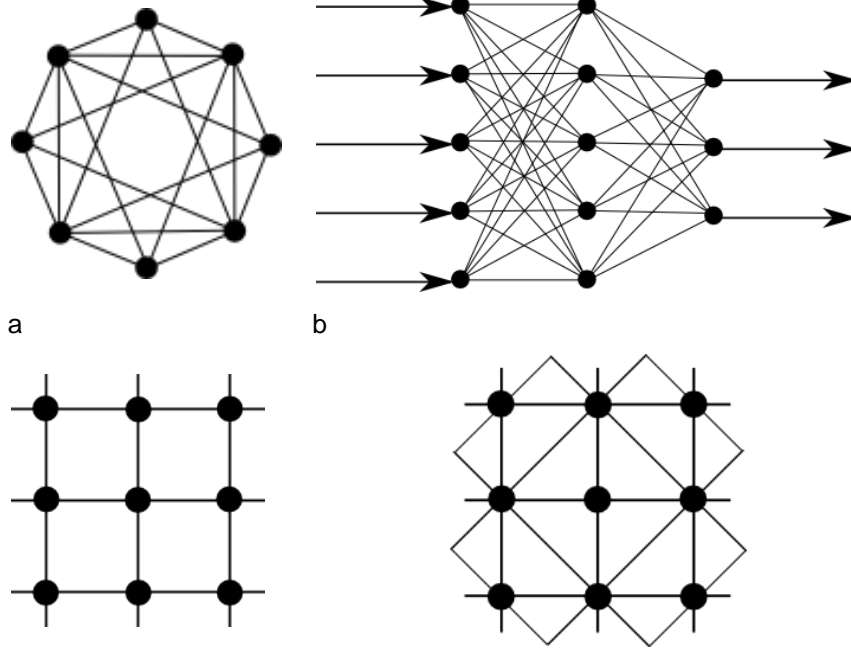
- Giriş nöronları: Bu tür nöronlara dış etkileri ya da dış sisteminin durumunu saklayan giriş vektörü verilmektedir.
- Çıkış nöronları: Bu nöronların çıkışları ağın çıkışını sağlamaktadır.
- Ara nöronları: YSA'nın temeli olan nöronlardır.

Dönüşümünü biçimlendiren koşullar:

1. Nöronlarının özellikleri (etkinleştirme fonksiyonu v.b)
2. Mimarisinin (yapısının) özellikleri:
 - a. Nöronlar arasındaki bağlantıların topoloji.
 - b. Belirli giriş ve çıkış nöronların seçilmesi.
 - c. YSA'nın öğretilmesi.
 - d. Nöronlar arasında rekabetin olup olmadığı.

Topolojisine göre ağlarının üç ana türü vardır:

- Tam ilişkili ağlar (Şekil 3.3 a)
- Çok katlı ağlar (Şekil 3.3 b)
- Zayıf ilişkili ağlar (Şekil 3.3 c)



Şekil 3.3: YSA'ların ana yapıları. a - tam ilişkili, b - çok katlı, c - zayıf ilişkili (Ostrovsiy, 2002)

Tam ilişkili YSA'ların tüm nöronları birbirine bağlıdır, yani bir nöronun çıkışı diğer bütün nöronlarla ve kendi girişi ile bağlıdır. Ağın çıkış sinyali olarak ağın bütün nöronlarının çıkışı ya da sadece bazı nöron çıkışları kullanılabilir.

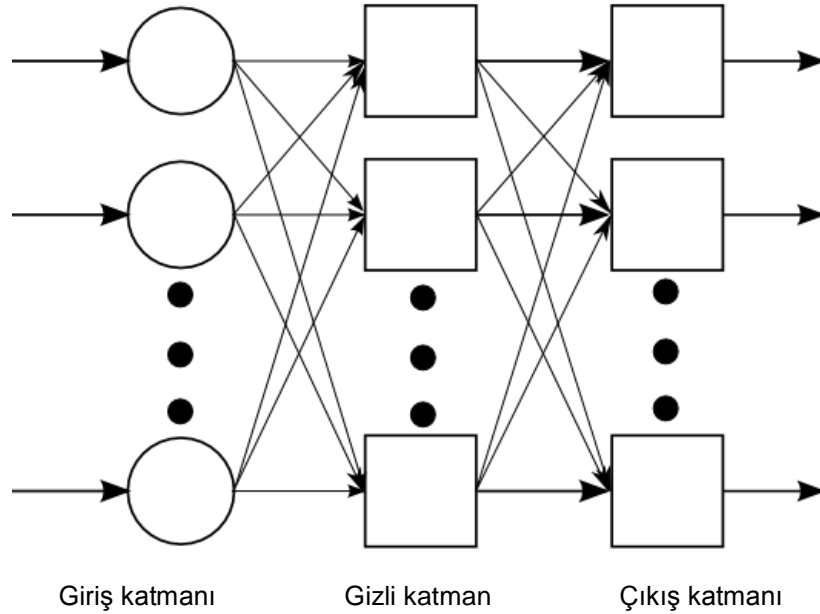
Çok katmanlı ağlarda nöronlar katmanlarla birleştirilirler. Her hangi bir katmandaki nöronların sayısı diğer katmanlardaki nöronların sayısından bağımsızdır. Genel olarak bu tür bir ağ Q katmandan oluşur. Giriş sinyalleri 1. katmandaki nöronlarının girişlerine verilir. Ağın çıkışı olarak son katmandaki nöronların çıkışları kullanılır. Giriş ve çıkış katmanları hariç ağda bir ya da daha fazla gizli katman vardır.

Katmalı ağlar da şu üç türe ayrılır:

- Monoton ağlar.
- İleri beslemeli ağlar.
- Geri beslemeli ağlar.

3.5. İleri Beslemeli YSA

Genellikle bu tür ağlarda sinyal ilk önce giriş katmana verilir. Giriş sinyali dönüştürdükten sonra ilk gizli katmana verilir. Ama, bazen sinyal giriş katmana verilmeden gizli katmana verilebilir. İlk gizli katmanın çıkışı sonraki bir katmana verilir. Bu işlem sinyalinin çıkış katmana ulaşacağına kadar tekrarlanmaktadır. Genel olarak bir gizli katmanın her çıkış sinyali, sonraki katmanın her bir nöronun girişine verilir. Ama bazen katmanın çıkışı sonraki katmandan direk başka katmanlara gönderilebilir. Şematik olarak ileri beslemeli ağının şeması Şekil3.4'de gösterilmektedir (Bu şekil (Kruglov, 2002) kaynaktan çizilmiştir).



Şekil 3.4: İleri beslemeli yapay sinir ağının yapısı (Kruglov, 2002)

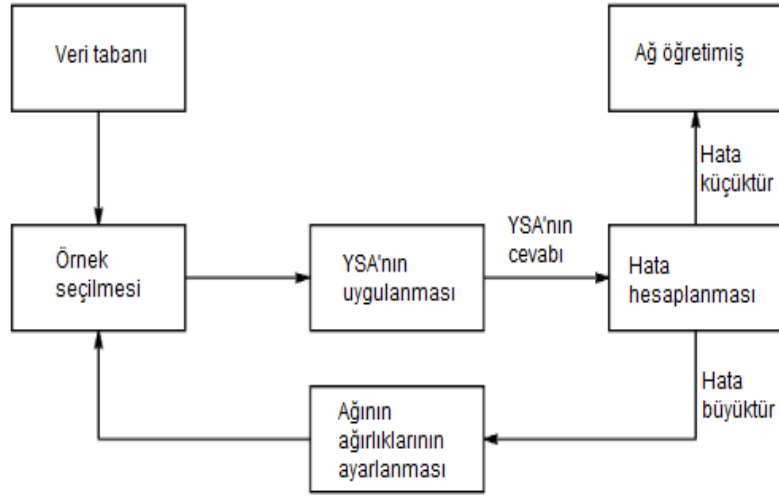
3.6. YSA'ların Öğretimi

Her hangi bir YSA'nın eğitimi için örnekler içeren bir veritabanı kullanılmaktadır. Ağın girişine verileri verip, çıkışından bir cevap alınır. Bu cevap yanlış olabilir. Doğru (beklenen) cevap ve alınmış cevaplar arasındaki fark hesaplanarak hata vektörü elde edilir. Öğretim algoritması, hata vektörüne göre nöronların ağırlıklarını düzeltebilen formüllerin setidir.

Aynı giriş sinyali YSA'a birkaç kez uygulanmaktadır. Bu şekilde öğretim işlemi defalarca uygulanıp ağın ağırlıkları stabilize edilir. Bundan sonra veritabandaki bütün örnekler ağ doğru cevapları verebilir. Bu durumda, ağa öğrenmiş ağ denir. YSA'ları modelleyen programlarda öğretim işlemi boyunca hata fonksiyonun

azaltılması gösterilmektedir. Hata fonksiyonu sıfıra ya da her hangi gerekli bir seviyeye ulaşınca öğretim işlemi bitirilir ve bu ağ öğretilmiş sayılır. Öğretme prosedürü Şekil 3.5 gösterilmiştir.

Öğretim kalitesi doğrudan doğruya örnekler sayısına ve bu örneklerin problemi ne kadar iyi betimlediğine bağlıdır. Eksiksiz bir öğretim için örneklerin sayısı yeteri kadar olmalıdır (Mesela birkaç yüzden fazla).



Şekil 3.5: YSA'nın öğretim işlemi (Ostrovsiy, 2002)

Matematiksel olarak öğretim işlemi şu şekilde gösterilir: Çalışma sürecinde YSA $Y=G(x)$ bir fonksiyonu gerçekleştirerek X girişi için uygun Y çıkışı verir. Sinapsların ağırlıkları ve kaymalar G fonksiyonunun şeklini belirtir.

$Y=F(x)$ fonksiyonu her hangi bir problemin çözümüdür. YSA'nın öğretimin amacı F fonksiyona benzer yani hata fonksiyonu az olan bir G fonksiyonu bulmaktır. Bu şekilde YSA'nın öğretimi çok boyutlu optimizasyon problemine dönüşülür. F fonksiyonu serbest şekilli olduğundan ağın öğretimi çok ekstremumlu bir optimizasyon problemidir.

Bu problemin çözülmesi için şu algoritmalar kullanılır (Kruglov, 2002):

- 1. derecen kısmi türevi hesaplanan lokal optimizasyon algoritmaları (Gradyan algoritması, eşleştirmeli Gradyan yöntemi v.b.)

- 2. derecen kısmi türevi hesaplanan lokal optimizasyon algoritmaları (Newton yöntemi, quasi Newton yöntemi, Gauss-Newton yöntemi, Levenberg–Marquardt algoritması v.b.)
- Stokastik optimizasyon algoritmaları (Monte Carlo benzetimi algoritması, benzetilmiş tavlama algoritması, genetik algoritma)
- Global optimizasyon algoritmaları

3.7. Geri Yayılım Algoritması

Geri yayılım algoritması en yaygın kullanılan algoritmalarından birisidir. Çok katmanlı YSA'nın gerçek ve beklenen çıkışları arasında ortalama karesel sapmayı optimize eden yinelemeli bir gradyan algoritmasıdır. Bu algoritma ileri beslemeli ağları öğretmek için kullanılır.

Hata fonksiyonu olarak gerçek ve beklenen çıkışları arasında fark karelerinin toplamı alınır. Gradyan vektörü hesaplanırken sigmoid aktivasyon fonksiyonların türevleri kullanılmaktadır. Algoritma yinelemeli olarak çalışır ve her yinelemeye epok denir. Her epokta ağın girişine bütün öğretim örnekleri sırasıyla verilirler. Çıkış değerleri hedef değerlerle karşılaşıp hata hesaplanır. Hatanın değeri ve hataların gradyanı, ağırlıkların düzeltilmesi için kullanılır. Ondan sonra bütün işlemler tekrarlanır. Ağın ilk konfigürasyonu rastgele bir yöntem ile seçilir. Öğretim işlemini bitirmek için üç koşul vardır: çağların sayısı belli bir değere ulaşması; hatanın uygun bir seviyeye ulaşması; hatanın azalmaması.

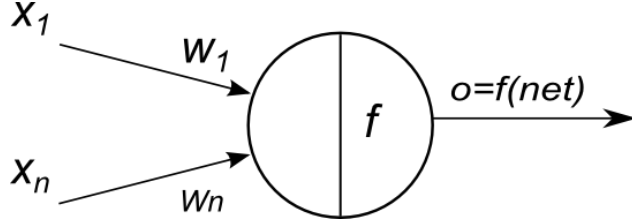
Algoritmanın betimlenmesi:

1. Ağ ağırlıklarına ilk değerler verilir.
2. Öğretim setinden olağan çift örnek (X, Y) seçilir. X vektörü ağın girişine verilir.
3. Ağın çıkışı hesaplanır.
4. Beklenen (hedef, Y) ve gerçek (hesaplanmış) ağ çıkışları arasındaki fark hesaplanır.
5. Hatayı minimize etmek amacıyla ağın ağırlıkları düzeltilirler.
6. Örneklerin bütün seti için hata uygun seviyeye ulaşacağına kadar, 2 - 5 adımlar her çift örnek için tekrarlanırlar.

Adımlar 2 ve 3 ağı normal çalışmasında kullanılırlar.

Ağıdaki hesaplamalar katmanlı olarak gerçekleştirilir. 3. adımda ağı her bir çıkışı hatayı hesaplamak amacıyla hedef vektörün uygun bir bileşiminden çıkarılır.

Tek nöronlu ağı (Şekil 3.6) öğretimi:



Sekil 3.6: Tek nöronlu YSA(Ostrovsiy, 2002)

Tek nöronlu ağı çıkışı o sigmoid tipi $f(net)$ transfer fonksiyonuna bağlı olsun. Bu fonksiyonu şu şekilde veririz:

$$o = o(w^T x) = \frac{1}{1 + e^{-w^T x}} \quad (3.5)$$

Burada $w^T = (w_1, w_2, \dots, w_n)$ nöron ağırlıklar vektörü ve $x^T = (x_1, x_2, \dots, x_n)$ giriş sinyaller vektörüdür.

Ağı öğretimi için sonraki örnekler kullanılsın:

$$x^1 = (x_1^1, x_2^1, \dots, x_n^1)^T, y^1,$$

$$x^2 = (x_1^2, x_2^2, \dots, x_n^2)^T, y^2,$$

...

$$x^N = (x_1^N, x_2^N, \dots, x_n^N)^T, y^N$$

(3.6)

Burada y^k – beklenen çıkış değeri.

k .örnek için hata fonksiyonu olarak gerçek ve beklenen çıkış değerleri arasında farkın karesinin değeri alınsın.

$$E_k = \frac{1}{2}(y^k - o^k)^2 = \frac{1}{2}(y^k - o^k(w^T x^k))^2 = \frac{1}{2}\left(y^k - \frac{1}{1 + e^{-w^T x^k}}\right)^2 \quad (3.7)$$

Daha sonra bütün örnekler için toplam hata aşağıdaki gibi hesaplanır.

$$E = \sum_{k=1}^N E_k \quad (3.8)$$

Hem E hem de E_k ağırlıklarına bağlı fonksiyonlardır. O zaman öğretmenin amacı böyle bir ağırlıklar vektörü bulup, E değerini minimuma ulaştırmaktır. Bu optimizasyon problemi aşağıdaki oranı kullanarak gradyan yöntemi ile çözülebilir.

$$w := w - \eta \dot{E}_k(w) \quad (3.9)$$

Burada $:=$ atama operatörüdür, \dot{E}_k vektörün gradyanı, η - bir sabit değeridir. Sigmoid transfer fonksiyonun türevini alarak (2.2) $\dot{E}_k(w)$ fonksiyonu şu şekilde yazılabilir:

$$\dot{E}_k(w) = \frac{d}{dw} \left(\frac{1}{2} \left(y^k - \frac{1}{1 + e^{-w^T x^k}} \right)^2 \right) = -(y^k - o^k) o^k (1 - o^k) x^k \quad (3.10)$$

Ağırlıkların düzeltilme algoritması şu şekilde verilebilir:

$$w := w + \eta (y^k - o^k) o^k (1 - o^k) x^k = w + \eta \delta_k x^k \quad (3.11)$$

Burada:

$$\delta_k = (y^k - o^k) o^k (1 - o^k) \quad (3.12)$$

Ele alınmış matematiksel ifadeler ile öğretim algoritması tamamen tanımlanır. Algoritmayı şu şekilde yazabiliriz:

1. Sabit değeri η ($0 < \eta < 1$), hatanın maksimum kabul edilebilir değeri E_{max} ve rastgele sinops ağırlıkları w_i seçilir.
2. $k = 1$ ve $E = 0$ verilir.

3. Sırası gelen örnek çifti (x^k, y^k) girilir. $x := x^k$ veya $y := y^k$ atamaları uygulanırlar ve ağırlık çıkış değeri hesaplanır:

$$o = o(w^T x) = \frac{1}{1 + e^{-w^T x}} \quad (3.5)$$

4. Ağırlıklar düzeltilir:

$$w := w + \eta(y - o)o(1 - o)x \quad (3.13)$$

5. Hatanın değeri hesaplanır:

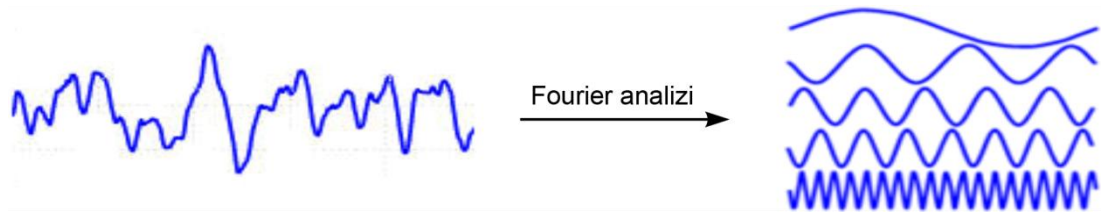
$$E := E + \frac{1}{2}(y - o)^2 \quad (3.14)$$

6. Eğer $k < N$, o zaman $k := k + 1$, ve algoritma 3. adımından devam edilir. Aksi halde algoritma 7. adıma geçilir.
7. Bir öğretim döngüsünün sonu. Eğer $E < E_{max}$, o zaman öğretim işlemi bitirilir. Eğer $E \geq E_{max}$ o zaman yeni öğretim döngüsü başlanır ve algoritma 2. adımdan devam edilir.

4. DALGACIK DÖNÜŞÜMÜ

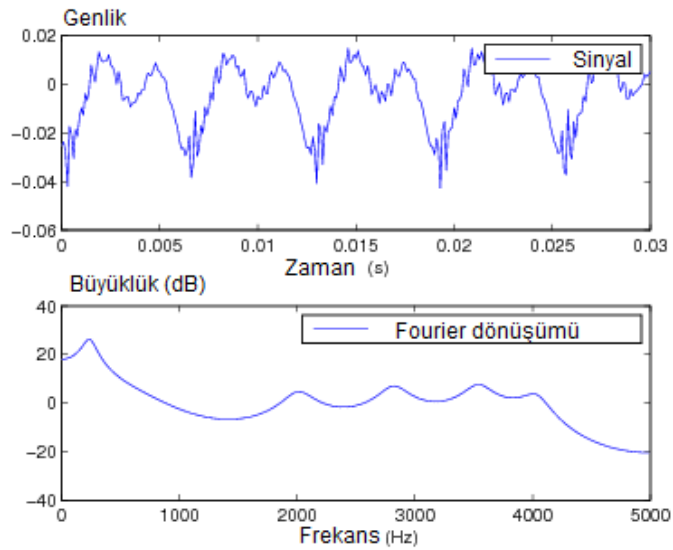
4.1. Fourier Dönüşümü

Dalgacık dönüşümü, Fourier dönüşümü ne göre sahip olduğu farklılardan yola çıkılarak anlatılabilir. Fourier analizi, işaretifarklı sinüzoidal birleşenlerine ayıran bir analiz işlemidir(Şekil 4.1).



Şekil 4.1: Bir işaretin farklı frekanslı sinüzoidal bileşenlerine ayrıştırılması.

Diğer bir ifadeyle Fourier dönüşümü, bir işaretin zaman uzayındaki gösteriliminden frekans uzayı gösterilimine geçiştir (Smolentsev, 2005). Zaman uzayında bir işaret $f(t)$ bir fonksiyon olarak düşünülebilir. Burada zaman uzayının bağımsız değişkenidir. Frekans uzayının bağımsız değişkeni w olmak üzere, bir işaret Fourier dönüşümünü uygulanarak $F(w)$ fonksiyon elde edilir.



Şekil 4.2: Bir işaret zaman uzayı gösteriliminin frekans uzayındaki eşdeğeri

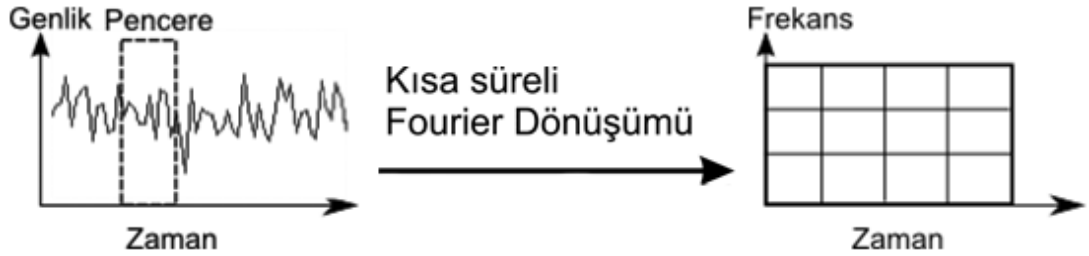
4.2. Kısa Süreli Fourier Dönüşümü

Fourier dönüşümünün önemli bir eksikliği: frekans uzayına dönüşüm yapılarak zamanla ilgili bilginin kaybetmesi. Bu nedenle Fourier dönüşümünde işaretin sahip olduğu frekans bileşenleri hakkında bilgi edinilebilir. Ancak bu frekanslar hangi anlarda olduğu tespit edilemez.

Eğer işaret özellikleri zamanla çabuk değişmezse, o zaman bu eksiklik çok önemli değildir. İncelenen işaretler genellikle zamanla değişen özelliklere sahip olmaktadır. Genelde bu özellikler sinyalin en önemli kısmını taşırlar ve Fourier analizi ile anlaşılır.

Bu eksikliği gidermek amacı ile Dennis Gabor tarafından kısa süreli Fourier dönüşümü geliştirilmiştir. Bu yöntem belli zaman sürecindeki işaret küçük kısımlarının gösteren pencere tekniği kullanır. Fourier tekniği işareti iki boyutlu zaman ve frekans fonksiyona dönüştür.

Kısa süreli Fourier dönüşümü bir işaretin frekans ve zaman çözünürlüğü arasında bir ödünleşime imkân verir. Bu dönüşüm, bir işaretin hangi zamanda hangi frekansa sahip olduğunu gösterir (Şekil 4.3). Ancak bu bilgilerin kesinliği kullanılan pencerenin büyüklüğüne bağlı olduğundan bu bilgiler belli bir doğrulukla alınır.

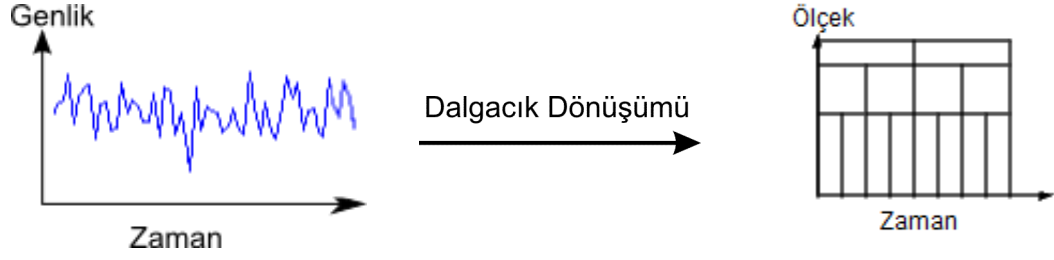


Şekil 4.3: Kısa zamanlı Fourier dönüşümü

Ama sinyallerin çoğu frekansı ve zamanı daha kesin analize edebilmek için pencerenin büyüklüğünü değiştirebilen daha esnek bir yönteme ihtiyaç duyarlar.

4.3. Dalgacık Dönüşümü

Dalgacık analizi, büyüklüğü değişen bir pencere tekniğidir (Şekil 4.4). Eğer düşük frekanslı işaret incelenecekse uzun süreli pencere, eğer yüksek frekanslı işaret incelenecekse kısa süreli pencere kullanılır.



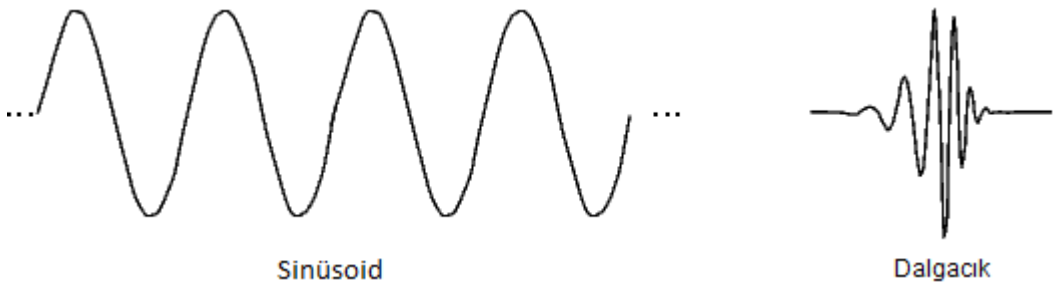
Şekil 4.4: Dalgacık dönüşümü

Dalgacık dönüşümünün avantajlarından biri yerel analiz gerçekleştirebilmesi, yani büyük bir işaretin belirli bir kısmını analiz edebilmesidir. Dalgacık analizi, başka dönüşümlerle belirlenemeyen işaret özellikleri yakalayabilmektedir. Örneğin: eğilimler, dönüm noktaları, yüksek türevlerin süreksizlikleridalgacık analiziyle kolayca belirlenebilir.

Dalgacık, uzunluğu sınırlı dalga şeklinde bir işarettir. Dalgacığın ortalama değeri sifıra eşittir. Yani,

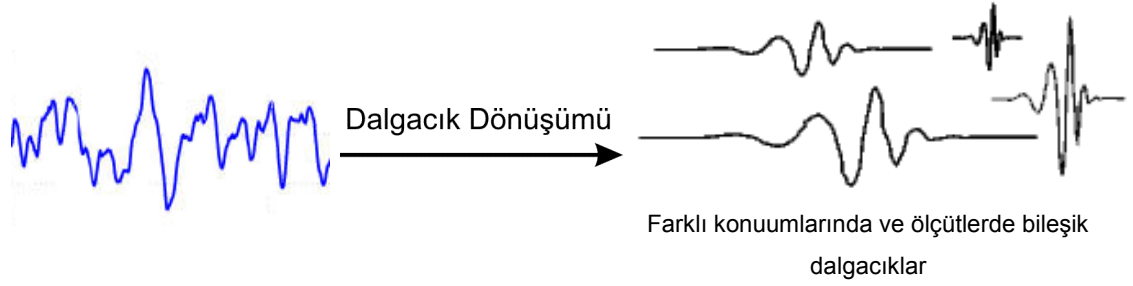
$$\int_{-\infty}^{+\infty} \psi(x) dx = 0 \quad (4.1)$$

Dalgacık ve sinüzoidal bir işaret arasında şufarklar vardır: Sinüzoidin tanım kümesi eksi sonsuz artı sonsuz aralığındadır(Şekil 4.5). Dalgacık ise yerel bir fonksiyonudur. Sinüzoidyumuşak değişen ve öngürebilen bir işaret olup dalgacık ise dengesiz ve asimetriktir.



Şekil 4.5: Sinüzoid ve dalgacık arasındaki fark.

Dalgacık analizi, bir işareti orijinal dalgacığın kaymış ve ölçeklenmiş biçimlerine ayrıştırır (Smolentsev, 2005). Bu işlem sekil 4.6'da gösterilmiştir.



Şekil 4.6: Dalgacık dönüşümü

Sayısal görüntüleri farklı çözünürlüklerde analize etmek gerekirse ayırık dalgacık dönüşümün (ADD) kullanılması uygundur(Dyakonov, 2004).

4.4. Hızlı Dalgacık Dönüşümü

Bir işaretin ayrıştırılması için ilk önce seviye seçilir. Ayrıştırma seviyesi sayısı j ile gösterilsin.

Tek boyutlu işaretin dalgacık analizi, bir ölçekleme fonksiyonuna (φ) ve bir dalgacığa (ψ) dayalıdır. Bir dalgacık fonksiyonun amacı incelenen bir fonksiyonun detaylarını belirlemek, ölçekleme fonksiyonu ise kaba bir yaklaşıklığını belirlemektir. İki boyutlu bir işaretin dalgacık dönüşümünü gerçekleştirmek için bir ölçekleme fonksiyondan $\varphi(x, y)$ ve 3 tane dalgacık gereklidir (Mallat, 1989) (Tablo 4.1). İki boyutlu dalgacık fonksiyonları bir boyutlu uygun dalgacık fonksiyonlarının çarpımından oluşturulur.

Tablo 4.1. Ölçekleme ve dalgacıklar fonksiyonları

$\varphi(x, y) = \varphi(x)\varphi(y)$	Ölçekleme fonksiyonu
$\psi^H(x, y) = \psi(x)\varphi(y)$	Yatay dalgacık
$\psi^V(x, y) = \varphi(x)\psi(y)$	Dikey dalgacık
$\psi^D(x, y) = \psi(x)\psi(y)$	Diagonal dalgacık

Bir boyutlu dalgacık fonksiyonu bir ana dalgacıktan oluşur. Bir ailedeki dalgacıklar iki parametreye bağlıdır: konum ve ölçüt. Dalgacığın konumu ana dalgacıktan b kadar uzaktaysa bu dalgacığın konumu, $\psi(x-b)$ ana dalgacık olmak üzere

$$\psi_b = \psi(x-b), (x \in R) \quad (4.2)$$

şeklinde belirlenir. Ölçek değişikliği ise şu şekilde ifade edilir:

$$\psi_a = \frac{1}{\sqrt{a}} \psi\left(\frac{x}{a}\right), (x \in R) \quad (4.3)$$

Hem konum hem de ölçüt değişikliği aşağıdaki gibi temsil edilir:

$$\psi_{a,b} = \frac{1}{\sqrt{a}} \psi\left(\frac{x-b}{a}\right), (x \in R) \quad (4.4)$$

Genellikle dalgacıkların tanımlanmasında aşağıda verilen ayırık değerler kullanılır:

$$a = 2^j, b = k2^j = ka, (j, k) \in Z^2 \quad (4.5)$$

Aynı değerler için dalgacık ve ölçekleme fonksiyonu:

$$\psi_{j,k} = 2^{j/2} \psi(2^j - k) \quad (4.6)$$

$$\varphi_{j,k} = 2^{j/2} \varphi(2^j - k) \quad (4.7)$$

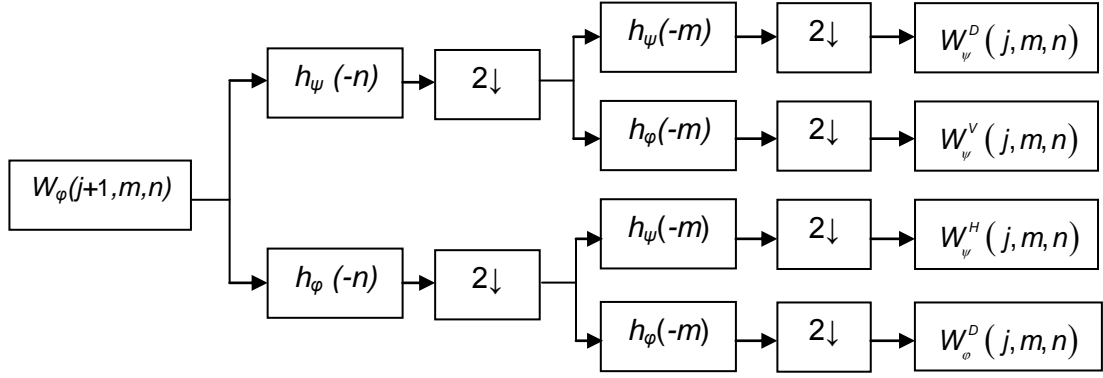
şeklinde yazılabilir. Burada, j ve k parametreleritamsayılarıdır. k parametresi bir boyutlu fonksiyonların x eksene göre konumunu ve j parametresi de onların genişliğini belirler. $2^{j/2}$ çarpanı ise bu fonksiyonların genliğini kontrol eder.

Ölçekleme ve dalgacık fonksiyonları aşağıda verilen eşitlikleri sağlar (Gonzales, 2006):

$$\varphi(x) = \sum_n h_\varphi(n) \sqrt{2} \varphi(2x - n) \quad (4.8)$$

$$\psi(x) = \sum_n h_\psi(n) \sqrt{2} \psi(2x - n) \quad (4.9)$$

Burada, $h_\psi(n)$ ve $h_\varphi(n)$ katsayılarına ölçekleme ve dalgacık vektörler denir. Bu katsayılar hızlı dalgacık dönüşümünde (HDD) filtreleme katsayıları olarak kullanılırlar (Mallat, 1989). İki boyutlu bir işaret için HDD'nin iteratif hesaplanması Şekil 4.7'de gösterilmiştir.



Şekil 4.7: HDD'nün hesaplanması

Burada $W_\varphi(j, m, n)$ ve $\{W_\psi^i(j, m, n), i = H, V, D\}$ j . ölçekteki HDD katsayılarıdır. Ölçekleme $h_\varphi(-n)$ ve dalgacık katsayıları $h_\psi(-m)$ alçak ve yüksek geçiren filtrelerdir. $2\downarrow$ sembolü 2 ile alt örnekleme göstermektedir.

Özetle, yukarıda verilen Şekil 4.7'nin çalışması şöyle açıklanabilir: Görüntüye dalgacık dönüşümünün uygulanması sonucunda dört alt band elde edilir. Bu işlem, görüntüye düşük ve yüksek frekanslı filtreler uygulayarak gerçekleştirilir.

Matematiksel olarak tüm alt bandlar aşağıdaki şekilde yazılabilir:

$$W_\psi^H(j, m, n) = h_\psi(-m) * (h_\varphi(-n) * W_\varphi(j+1, m, n)|_{n=2k, k \geq 0})|_{m=2k, k \geq 0} \quad (4.10a)$$

$$W_\psi^D(j, m, n) = h_\psi(-m) * (h_\psi(-n) * W_\varphi(j+1, m, n)|_{n=2k, k \geq 0})|_{m=2k, k \geq 0} \quad (4.10b)$$

$$W_\psi^V(j, m, n) = h_\varphi(-m) * (h_\psi(-n) * W_\varphi(j+1, m, n)|_{n=2k, k \geq 0})|_{m=2k, k \geq 0} \quad (4.10c)$$

$$W_\varphi(j, m, n) = h_\varphi(-m) * (h_\varphi(-n) * W_\varphi(j+1, m, n)|_{n=2k, k \geq 0})|_{m=2k, k \geq 0} \quad (4.10d)$$

W_φ katsayıları düşük frekanslı filtre görüntünün ilk önce satırlarına daha sonra sütunlarına uygulanarak elde edilir. Bu yüzden bu katsayılara yaklaşıklık katsayıları denir. $\{W_\psi^i(j, m, n), i = H, V, D\}$ katsayılarına yatay, dikey ve diyagonal detay katsayıları denir. İlk iterasyonda algoritmanın girişine $W_\varphi(j+1, m, n)$ uygulanır: en yüksek çözünürlüklü işaret olarak görüntüsünün kendisi kullanılır.

4.5. Dalgacık Dönüşümünün Kullanım Alanları

Deneysel verilerin yorumlanması. Dalgacık dönüşümü (DD) deneysel verilerin yorumlanması amacıyla günümüze kadar başarıyla kullanılmıştır. DD bir deneyimin sonuçlarını daha görsel ve anlaşılabilir bir şekilde sunabilmektedir.

DD özellikle tıp, tahvilat borsaların analize ve başka alanlarda karşılaşılan durağan olmayan (non-stationary) işaretlerin işlenmesi için de kullanılmaktadır.

Görüntü işleme. DD yardımı ile bir görüntüdeki gürültünün giderilmesi, bir görüntünün küçültülmesi ya da büyütülmesi gibi işlemler yapılabilir.

DD, veri sıkıştırma amacıyla da başarıyla kullanılmıştır.

5. GÜRÜLTÜ GİDERME YÖNTEMLERİ

5.1. Klasik Gürültü Giderme Yöntemleri

Gürültü onarımı oldukça eski bir problemidir. Çeşitli onarım yöntemleri geliştirilmiştir. Gürültü giderme, işlemin hangi uzayda yapıldığına bağlı olarak ana iki sınıfa ayrılır:

1. Uzamsal (piksel) uzayı yöntemleri.
2. Frekans uzayı yöntemleri.
3. Dalgacık bölgesinde gürültü giderme.

Piksel uzayı gürültü giderme yöntemleri, adında da anlaşılacağı üzere görüntüyü parlaklık seviyelerini uygulanan bir görüntünün uzayında uygulanırlar. Bu tür algoritmalar aşağıdaki filtreleri kullanmaktadır: Ortalama filtreleri, sıra istatistiği filtreleri ve uyarlanır filtreler. Tablo 5.1 ve 5.2'de ortalama ve sıra istatistiği filtrelerin özellikleri ve matematik modelleri özetlenmiştir (Gonzales 2006):

Tablo 5.1: Ortalama filtreleri

Geometrik ortalama	Geometrik ortalamanın bulunması üzerinde çalışan filtre. Düzeltile pikselin değerini bulmak için önce S_{xy} komşuluğundaki piksellerin değerleri çarpılır. Sonra, çarpımın $n*m$ dereceden kökü alınır. ($m*n$: S_{xy} komşuluğundaki piksellerin sayısı).	$\hat{f}(x, y) = \left[\prod_{(s,t) \in S_{xy}} g(s, t) \right]^{\frac{1}{mn}}$
Aritmetik ortalama	Düzeltile pikselinin (x, y) konumundaki değeri S_{xy} komşuluğundaki piksellerin aritmetik ortalamasına eşittir.	$\hat{f}(x, y) = \frac{1}{mn} \sum_{(s,t) \in S_{xy}} g(s, t)$
Harmonik ortalaması	Tek kutuplu beyaz gürültü gidermesi için uygulanır.	$\hat{f}(x, y) = \frac{mn}{\sum_{(s,t) \in S_{xy}} \frac{1}{g(s, t)}}$

Tablo 5.2: Sıra istatistiği filtreleri

Oranca filtresi	İmpuls gürültünün gidermesi için uygulanır.	$\hat{f}(x, y) = \text{med}_{(s,t) \in S_{xy}} \{g(s, t)\}$
Maksimum filtresi	Tek kutuplu beyaz gürültün gidermesi için uygulanırlar.	$\hat{f}(x, y) = \max_{(s,t) \in S_{xy}} \{g(s, t)\}$
Minimum filtresi		$\hat{f}(x, y) = \min_{(s,t) \in S_{xy}} \{g(s, t)\}$
Orta nokta filtresi	Düzeltilen noktanın değeri, S_{xy} komşuluğundaki maksimum ve minimum değerlerin ortalamasına eşittir.	$\hat{f}(x, y) = 1/2(\min_{(s,t) \in S_{xy}} \{g(s, t)\} + \max_{(s,t) \in S_{xy}} \{g(s, t)\})$

Basit olmasına rağmen bu tür algoritmalar çok gürültülü görüntülerin onarımında başarılı sonuçlar vermemektedir.

Frekansuzayında onarım yapabilmek için görüntüler ilk önce uygun bir dönüşüm ile frekans uzayına dönüştürülür. Daha sonra, frekans uzayındaki belirli katsayıların değiştirilmesiyle gürültü giderme işlemi gerçekleştirilmiş olur.

5.2. Dalgacık Eşikleme

Görüntü işlemede dalgacıkların kullanması 80'li yıllarda başlamıştır. Dalgacık dönüşümün işaret düşük ve yüksek frekanslı bileşenlere ayırma yeteneği vardır. Bu özelliği nedeniyle dalgacık dönüşümü gürültü giderme problemini çözmek için birçok algorithmada kullanılmıştır.

Dalgacık eşikleme, dalgacıkların özelliklerini kullanan bir gürültü giderme tekniğidir. Gürültü giderme işlemi, belli bir eşik değerinin altındaki katsayıları azaltılarak gerçekleştirilir. Gürültü giderme, tamamen eşikleme tekniğine ve eşik parametresine bağlıdır (Yang).

Donoho tarafından iki tür eşikleme algoritması sunulmuştur: yumuşak (Donoho, 1995) ve katı eşikleme.

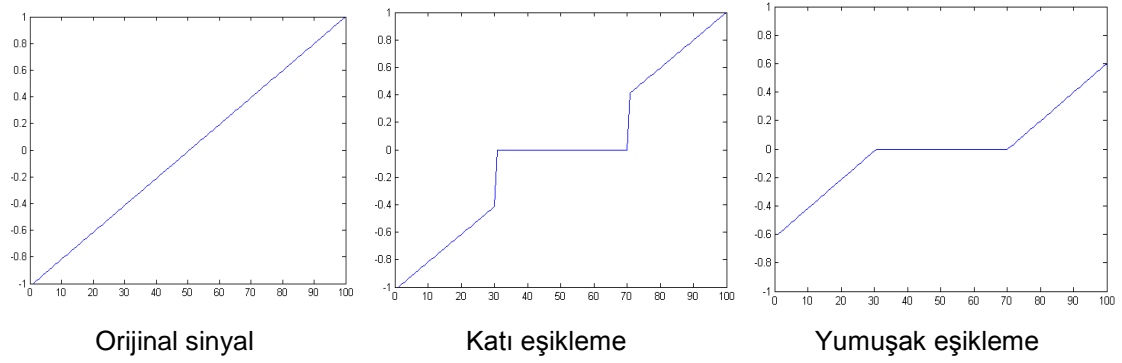
Katı eşiklemede, eşik parametresi T 'den daha küçük değerlere sahip çıkan katsayılar sıfıra eşitlenirler. Diğer katsayıların değerleri değiştirilmez. Katı eşikleme operatörü şöyle gösterilir:

$$D(x, T) = \begin{cases} x, & |x| > T \\ 0, & |x| \leq T \end{cases} \quad (5.1)$$

Yumuşak eşikleme, katı eşiklemenin genişletilmiş halidir. Önce katı eşiklemedeki gibi eşik değerin altındaki katsayı değerleri sıfırlanırlar. Eşik değerden büyük katsayılar sıfırdoğru sıkıştırılırlar. Yumuşak eşikleme operatörü şöyle gösterilir:

$$D(x, T) = \begin{cases} \text{sign}(x)(|x| - T), & |x| > T \\ 0, & |x| \leq T \end{cases} \quad (5.2)$$

Şekil 5.1’de bir doğrusal bir fonksiyon için katı ve yumuşak eşiklemenin grafikleri gösterilmiştir. Eşik değeri 0.4 seçilmiştir.



Şekil 5.1: Katı ve yumuşak eşikleme

Yukarıda söylendiği gibi, gürültü giderme işlemi eşikleme parametresi T 'ye bağlıdır. Bu parametreyi bulmak için farklı yöntemler geliştirilmiştir. Bazı yöntemlerde T globaldir, yani tüm görüntü aynı eşikleme değerini kullanır. Bazıları yöntemlerde ise T yereldir, yani her bir alt band için eşikleme parametresi değişebilir. Bu amaçlar için Sureshrink(Johnstone, 1995), VisuShrink(Donoho, 1994) ve BayesShrink (Vetterli, 2000) gibi yöntemler vardır.

“New Method for Image Denoising while Keeping Edge Information”(Wei, 2009) – ‘Kenarları koruyan yeni bir Gürültü giderme yöntemi’ Donoho’nun çalışmasını iyileştiren çalışmalardan birisidir. Bu çalışmanın arasındaki genel temel kavramı şöyle açıklanabilir: eşikleme parametresi büyükse, gürültünün gidermekle birlikte yumuşak eşikleme yöntemi kenarları da bulanıklaştırılır. Bu problemi çözebilmek için eşikleme aşamasından önce kenar bulma tekniği uygulanır ve kenarlar olduğuna karar verilen noktalar için daha düşük seviyedeki eşikleme uygulanır.

Kenar bulma aşamasında dalgacık dönüşümünü kullanan kenar bulma tekniği kullanılır. Algoritmanın detayları aşağıda vermiştir:

1. Görüntüde kenarlara uygun dalgacık katsayıları bulunur.
2. Kenarlara karşılık gelen katsayıları korunur.
3. Orijinal gürültülü görüntü bir seviyeli dalgacık dönüşümü uygulanır
4. Alt bantlara yumuşak eşikleme uygulanır. Eşik değeri VisuShrink yöntemi baz alınarak $T = \sigma\sqrt{2\ln N}$ formülünden hesaplanır. (Burada σ gürültü varyansını, N ise alt bantlardaki nöron sayısını belirtmektedir).
5. Kenarlara karşılık gelen dalgacık katsayılar korunmuş katsayılarla değiştirilir. Korunmuş katsayılar gürültülü olduğunda eşiklemeye tabi tutulurlar. Ancak, eşikleme parametresi T bu kez daha küçük seçilir: $T = \beta\sigma\sqrt{2\ln N}$ ($0.2 < \beta < 0.3$)
6. Alt bantlara ters dalgacık dönüşümü uygulanıp, onarılmış görüntü elde edilir.

6. KENAR BULMA

Görüntüyü iki boyutlu bir fonksiyon olarak de alalım ve $f(x,y)$ şeklinde temsil edelim. Bu fonksiyonun gradyanı $\nabla F(x,y)=f(x,y)/dx dy$ olup, gradyanın yerel maksimum değerleri görüntünün kenarlarına karşılık gelir. Ancak, görüntü beyaz Gauss'un gürültüsü ile bozulmuşsa o zaman gradyanın yerel maksimum değerleri gürültüye de karşılık gelecektir. Amacımız görüntünün kenarlara karşılık gelen katsayıları tespit edebilmektir. Aynı zamanda, kenarlar matematiksel olarak Lipschitz düzenliliği ile betimlenebilir. Lipschitz üssünün (exponent) tanımını verelim. Eğer (a, b) aralığında her x ve x_0 için $|f(x)-f(x_0)| \leq K|x-y|^\alpha$ eşitliğine uygun K sabiti varsa, bu aralıkta $f(x)$ fonksiyonuna α üssü ile düzenli Lipschitz denir. Yani, eğer (a,b) aralığında $f(x)$ fonksiyonu düzenli Lipschitz ise bu fonksiyonun eğimi K sabitinden daha büyük olamaz.

Şimdi, çok ölçekli dalgacık kenar bulma tekniği tanıtılacaktır. Yukarıda bahsedildiği gibi görüntünün kenarları, bu görüntünün gradyanının maksimum değerlerine denk gelir. Bu nedenle, kenarları bulmak için gradyan operatörünü kullanabiliriz. Ancak, görüntü gürültülüyse, görüntünün gradyanının hesaplanma doğruluğu azalır. Bu nedenle, kenar bulma yöntemleri genellikle hem yumuşatma filtresini hem de gradyan operatörünü kullanır. Yani önce görüntü alçak geçiren bir filtreden geçirilir, daha sonra gradyan operatörü uygulanır.

İki kez türetebilen bir düzleştirme fonksiyonu $\theta(x)$ olsun. $\psi^a(x)$ ve $\psi^b(x)$ bu fonksiyonun birinci ve ikinci türevlerini gösterebiliriz (Mallat, 1992). Yani,

$$\psi^a(x) = \frac{d\theta(x)}{dx} \quad (6.1)$$

ve

$$\psi^b(x) = \frac{d^2\theta(x)}{dx^2} \quad (6.2)$$

Bu fonksiyonların integralı sıfıra eşit olduğundan ve dalgacığın tanımına göre $\psi^a(x)$ ve $\psi^b(x)$ dalgacıklar olarak düşünülebilirler:

$$\int_{-\infty}^{+\infty} \psi^a(x) dx = 0 \quad (6.3)$$

ve

$$\int_{-\infty}^{+\infty} \psi^b(x) dx = 0 \quad (6.4)$$

Dalgacık dönüşümünü, görüntünün belli bir dalgacık ile konvolüsyonu hesaplayarak gerçekleştirebiliriz. S . ölçekte ve x konumunda, $\psi^a(x)$ ve $\psi^b(x)$ dalgacıklarına göre $f(x)$ fonksiyonunun dalgacık dönüşümü

$$W_s^a f(x) = f * \psi_s^a(x) = f * \left(S \frac{d\theta_s}{dx} \right)(x) = S \frac{d}{dx} (f * \theta_s)(x) \quad (6.5)$$

$$W_s^b f(x) = f * \psi_s^b(x) = f * \left(S^2 \frac{d^2\theta_s}{dx^2} \right)(x) = S^2 \frac{d^2}{dx^2} (f * \theta_s)(x) \quad (6.6)$$

şeklinde ifade edilebilir. Bu nedenle, $W_s^a f(x)$ ve $W_s^b f(x)$, S . ölçekte alçak geçiren filtreden geçirilmiş işaretin birinci ve ikinci türevleridir. $W_s^a f(x)$ fonksiyonun yerel uç noktalarına, $W_s^b f(x)$ fonksiyonun sıfır noktasına ve $f * \theta_s(x)$ ise fonksiyonun kenarlarına karşılık gelir.

Şimdi bu dönüşümünün iki boyutlu halini tanıttacağız. $\psi_s^1(x, y)$ ve $\psi_s^2(x, y)$ dalgacıkları

$$\psi_s^1(x, y) = \frac{1}{s^2} \psi_1\left(\frac{x}{s}, \frac{y}{s}\right) \quad (6.7)$$

ve

$$\psi_s^2(x, y) = \frac{1}{s^2} \psi_2\left(\frac{x}{s}, \frac{y}{s}\right) \quad (6.8)$$

olarak tanımlansın. $f(x, y)$ görüntüsünün S . ölçütteki dalgacık dönüşümünden iki farklı bileşen elde edilir:

$$W_s^1 f(x, y) = f * \psi_s^1(x, y) \quad (6.9)$$

ve

$$W_s^2 f(x, y) = f * \psi_s^2(x, y) \quad (6.10)$$

O halde,

$$\begin{pmatrix} W_s^1 f(x, y) \\ W_s^2 f(x, y) \end{pmatrix} = S \begin{pmatrix} \frac{\partial}{\partial x} (f * \theta_s)(x, y) \\ \frac{\partial}{\partial y} (f * \theta_s)(x, y) \end{pmatrix} = S \nabla (f * \theta_s)(x, y) \quad (6.11)$$

Özetle, görüntünün kenarları dalgacık dönüşümünün $W_s^1 f(x, y)$ ve $W_s^2 f(x, y)$ bileşenlerinden belirenebilir

Alçak geçiren fonksiyon olarak aşağıda verilen Gaussian fonksiyonunu kullanıldığını varsayalım (Li, 2003):

$$\theta(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (6.12)$$

Bu durumda, ilgili dalgacıklar

$$\psi^1(x, y) = \frac{\partial \theta}{\partial x} = -\frac{x}{2\pi\sigma^4} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (6.13)$$

$$\psi^2(x, y) = \frac{\partial \theta}{\partial y} = -\frac{y}{2\pi\sigma^4} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (6.14)$$

eşitliklerinden hesaplanır.

Dalgacığın ölçeği artınca görüntünün gerçek kenarlarını bulma olasılığı da artar. Ancak bu yöntemin de bir eksik tarafı vardır. Dalgacığın ölçeği artınca tespit edilen kenarın bulunduğu yerin hatası da artmaktadır. Küçük ölçekteki dalgacıklar kenarın bulunduğu yeri saklarlar, fakat kenarlar ve gürültü arasındaki farkı ayırt edemezler.

Dalgacık katsayılarını kullanarak Lipschitz düzenliliğini bulabiliriz. Ölçeğin artmasıyla birlikte katsayının değeri de artıyorsa, Lipschitz düzenliliği pozitifdir, aksi halde negatiftir. Daha düşük düzenliliğe sahip yerlerde genellikle ya detaylar ya da gürültü bulunur. Çok ölçekli dalgacık kenar bulma tekniğinin arkasındaki fikir şöyle açıklanabilir: Gürültüleri gidermek amacıyla dalgacık ölçeği artınca hızlı düşen katsayılar için büyük ölçekte, yavaş değişen katsayılar içinse küçük ölçekte dalgacık kullanırız.

7. GÜRÜLTÜ GİDERME ALGORİTMALARI

7.1. Görüntü Onarımı Amaçlı Yapay Sinir Ağ

Bu çalışmada, 2005'te yayınlanmış Zhang'ın 'Image de-noising using a neural network based non-linear filter in wavelet domain' başlıklı çalışması referans alınmıştır (Zhang, 2005). Bu çalışmada da görüntü onarımı yine dalgacık bölgesinde yapılmaktadır. Dalgacık bölgesindeki görüntü onarımının arkasındaki fikir, belirli bir algoritmayla dalgacık dönüşümünü gerçekleştirerek, görüntüyü dalgacık bölgesine çevirmek ve daha sonra bazı katsayıları sıfıra eşitlemek ve daha sonra bazı katsayıları sıfıra eşitlemektir.

Daha önce ifade edildiği gibi Donoho'nun algoritmasında bir eşik değeri belirlenip, bu eşik değeri altında kalan katsayılar sıfırlanarak gürültü giderme işlemi gerçekleştirilir (Yang). Ancak, Zhang eşikleme yerine görüntünün dalgacık bölgesinde gürültü giderme amaçlı bir yapay sinir ağı tabanlı yöntem geliştirilmiştir. Yöntem üç adımdan oluşur:

1. Görüntünün bir seviyeli dalgacık dönüşümünün oluşturulması.
2. 4 bantın yapay sinir ağlarla onarımı.
3. Onarılmış 4 bantın ters dalgacık dönüşümü alınarak onarılmış görüntü elde edilmesi.

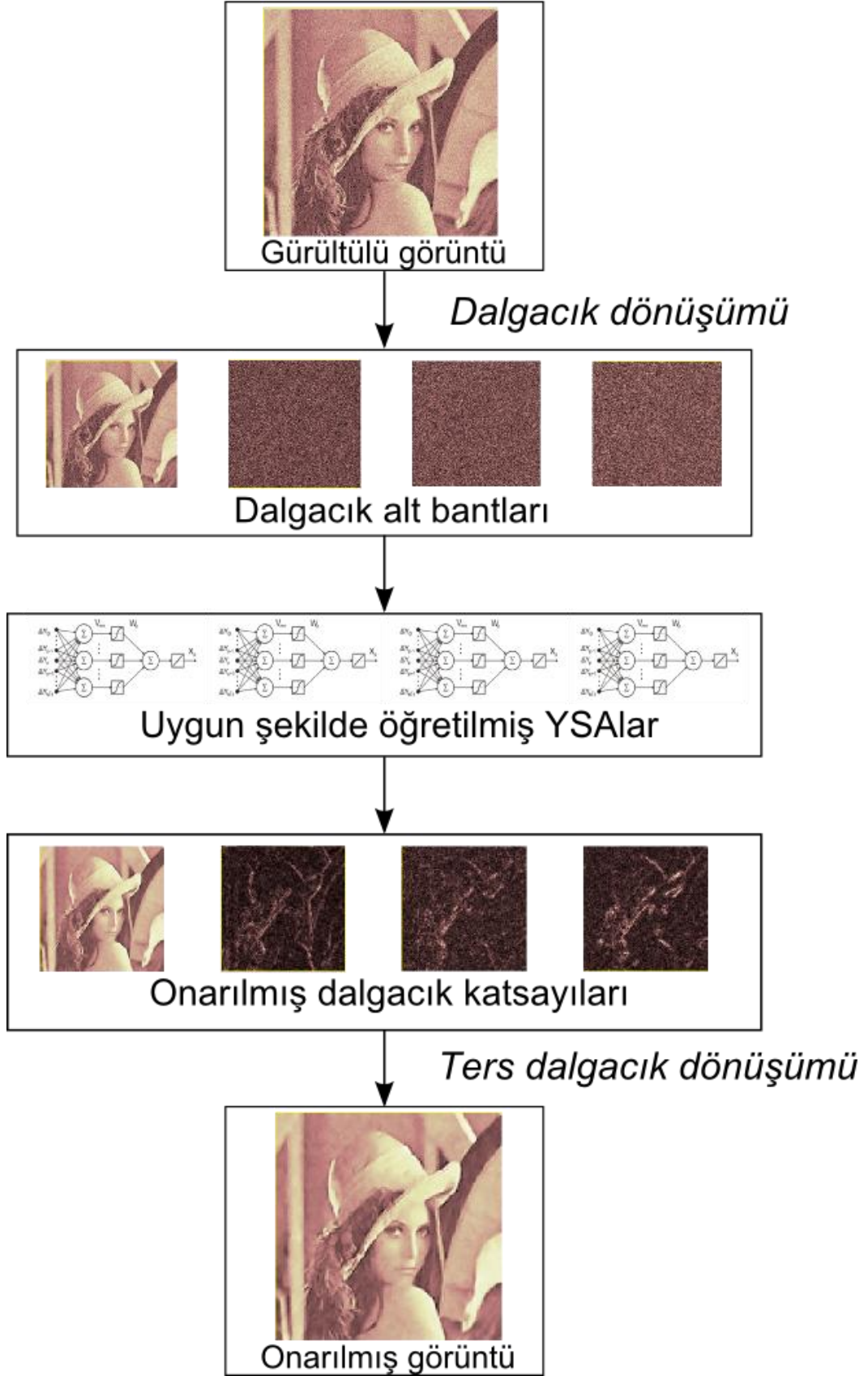
Birinci ve üçüncü adımlar Donoho'nun çalışmasıyla aynıdır, ancak ikinci adım farklıdır.

Algoritmanın merkezi elemanı, hatanın geri yayılmasını kullanan ileri beslemeli bir yapay sinir ağıdır (feed forward NN). Aynı anda görüntünün dalgacık dönüşümü önemli bir rol oynamaktadır; çünkü sinir ağının çalışması dalgacık bölgesinde gerçekleşmektedir. Toplam onarım mekanizması üç adıma ayrıştırılabilir:

1. Görüntünün bir seviyeli ayrık dalgacık dönüşümü hesaplanır. Bu dönüşümün sonucunda 4 alt bant oluşur: 1 adet yaklaşım katsayılarından oluşan alt bant ve 3 adet detay katsayılarından oluşan alt bant.

2. Her bir alt bant bazı dönüşümlerden geçirilerek kendisine uygun öğretilmiş bir yapay sinir ağının girişine verilir.
3. Yapay sinir ağlarının çıkışında alt bantların yeni katsayıları oluşur. Alt bantların ters dalgacık dönüşümü alınarak onarılmış görüntü elde edilir.

Bu üç adım, şematik olarak Şekil 7.1 gösterilmiştir:



Şekil 7.1: Gürültü giderme işleminin şematik gösterilimi

Aşamaların her biri aşağıda detaylı olarak açıklanmıştır.

7.2. Görüntü Hazırlama

Yapay sinir ağına, gürültülü ve orijinal görüntülerindalgacak katsayıları arasında korelasyonu bulabilmesi öğretilmiştir. Bu çalışmada, iki katmanlı ileri beslemeli ağ kullanılmıştır. Ağın girişine gürültülü görüntünün katsayılarıuygulanarak çıkışında bu katsayıların onarılmış halleri elde edilir. Bu amaçla katsayısıyla beraber ağa, katsayının komşu olduğu katsayılar arasındaki farklar da verilir.

Y_p ve ΔY_k ağ için giriştir. Burada Y_p , Şekil 7.2'te gösterildiği gibipencerenin merkezindeki parlaklık seviyesi, ΔY_k ise merkez ve onun komşuları arasındaki farkı temsil etmektedir. Şekil 7.3'te pencerenin görüntünün (2,2) konumundaki yerleşimi görülmektedir.

160	154	157	158	157	159	..	155
156	159	158	155	158	156	..	158
160	154	157	158	157	159	..	155
156	153	155	159	159	155	..	155
155	155	155	157	156	159	..	152
156	153	157	156	153	155	..	155
154	157	157	157	156	155	..	161
152	150	155	154	152	156	..	157
..	157
151	154	157	156	156	158	159	158

Şekil 7.2: Pencerenin görüntünün (2, 2) konumundaki yerleşimi

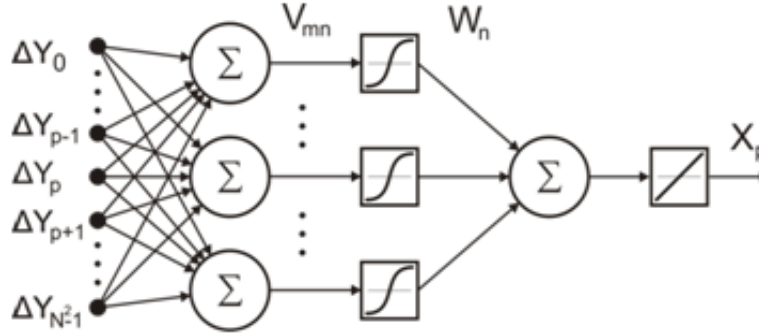
Şekil 7.3'te,3x3 boyutlarında bir pencere için piksellerinçekirdek içinde numaralanması gösterilmektedir.Bu örnekte Y_5 , pencerenin merkezindeki piksel ve $\Delta Y_k = Y_k - Y_5$ merkez piksel ile komşuları arasındaki farktır.

Y_1	Y_2	Y_3
Y_4	Y_5	Y_6
Y_7	Y_8	Y_9

Şekil 7.3: Merkezi Y_5 olan 3*3 boyutlu pencere

7.3. YSA'ların Yapıları

Çalışmada kullanılan yapay sinir ağının yapısında üç katman vardır: giriş katmanı, gizli katman ve çıkış. Ağın yapısı Şekil 7.4'te verilmiştir.



Şekil 7.4: YSA'nın yapısı

Ağın çıkışı şöyle gösterilebilir:

$$X_p = f_o \left(\sum_{n=0}^M W_n f_h \left(\sum_{m=0}^{N^2} Y_m V_{mn} \right) \right) \quad (7.1)$$

Burada $\Delta Y_k (k=0,1,\dots,N^2-1)$ yapay sinir ağının girişleri, X_p ise çıkıştır. m gizli katmandaki toplam nöron sayısı, W_n gizli katmandaki n . nöronü ağın çıkışına birleştiren ağırlık, V_{mn} ise gizli katmandaki n . ile giriş katmandaki m . nöronları birleştiren ağırlıktır.

Zhang'ın çalışmasında etkinleştirme (activation) fonksiyonları olarak aşağıdaki fonksiyonlar kullanılmıştır:

$$f_h(x) = xe^{-\gamma x^2} \quad (7.2)$$

$$f_o(x) = \frac{2}{(1+e^{-\lambda x})} - 1 \quad (7.3)$$

Ancak uygulamalarda tansig ve purelin etkinleştirme fonksiyonlar daha verimli sonuçlar vermişti. Bu nedenle çalışmada bu fonksiyonlar kullanılmıştır. Bu fonksiyonlar

$$f_h(x) = \frac{2}{(1 + e^{-2x})} - 1 \quad (7.4)$$

$$f_o(x) = x \quad (7.5)$$

şeklinde ifade edilebilir. Burada $f_o(x)$ çıkış katmanının ve $f_h(x)$ – gizli katmanının etkinleştirme fonksiyonunun belirtmektedir.

Şekil 7.1’de verilen onarım metodu dörtadet yapay sinir ağı kullanmaktadır. Bütün ağların yapıları aynıdır. Gürültülü görüntünün bir seviyeli dalgacık dönüşümü hesaplandıktan sonra ağlardan her biri her bant için ayrı öğretilir. Daha sonra, her ağ gürültülü görüntünün uygun dalgacık bandına uygulanır. Her bir ağın çıkışında onarılmış dalgacık katsayılarını elde ederiz. Bu katsayılara ters dalgacık dönüşümü uyguladıktan sonra onarılmış görüntü oluşur.

Tablo 7.1’de, bu çalışmada ve Zhang tarafından önerilmiş aktivasyon fonksiyonlarının etkinliği karşılaştırılmaktadır. Etkinlik parametresi olarak öğretim hatası kullanılmıştır. Öğretim hatası beklenen ve gerçek ağ çıkışları arasındaki farktır. Bu hatanın hesaplama fonksiyonu böyle verilir:

$$MSE = \frac{\sum_{t=1}^n |y_t - o_t|^2}{n} \quad (7.6)$$

Burada, y_t beklenen değerleri ve o_t çıkış değerlerini göstermektedir.

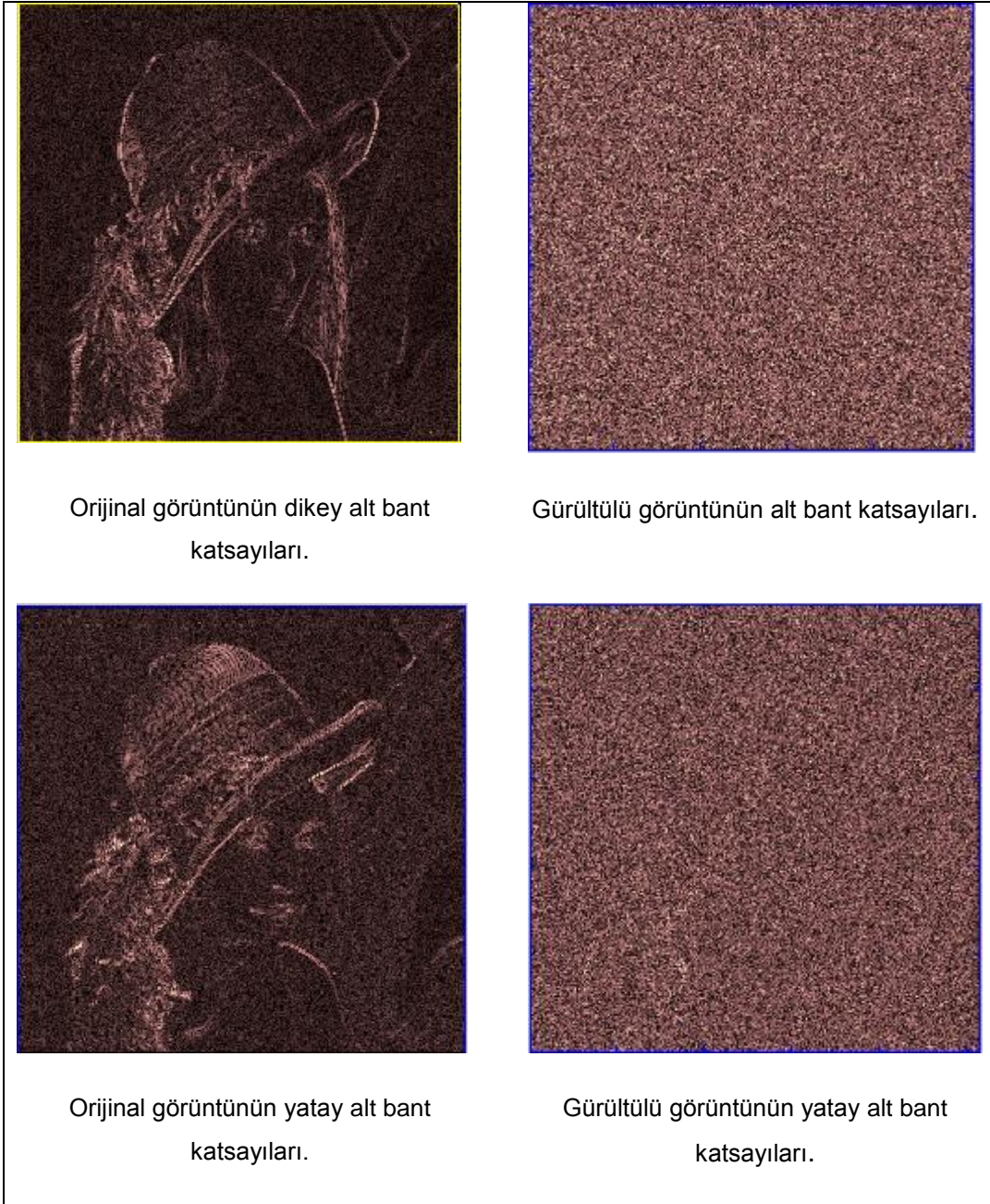
Tablo 7.1: Aktivasyon fonksiyonların karşılaştırılması

	Önerilen	Zhang
Yaklaşım katsayıları.	436	995
Yatay detay katsayıları	88	153
Dikey detay katsayıları	75	117
Diagonal detay katsayıları	13	19

7.4. Kenar Korunan Gürültü Giderme Algoritması

Gürültüsüz bir görüntünün dalgacık dönüşümünden elde edilmiş detay katsayıları kenarlar hakkında bilgi vermektedir. Ancak gürültülü bir görüntünün detay katsayıları kenar bilgisinin yanı sıra gürültü bilgisini de içermektedir.

Şekil 7.2'da orijinal ve gürültülü görüntüye karşılık gelen dikey ve düzey alt bant katsayıları gösterilmiştir:

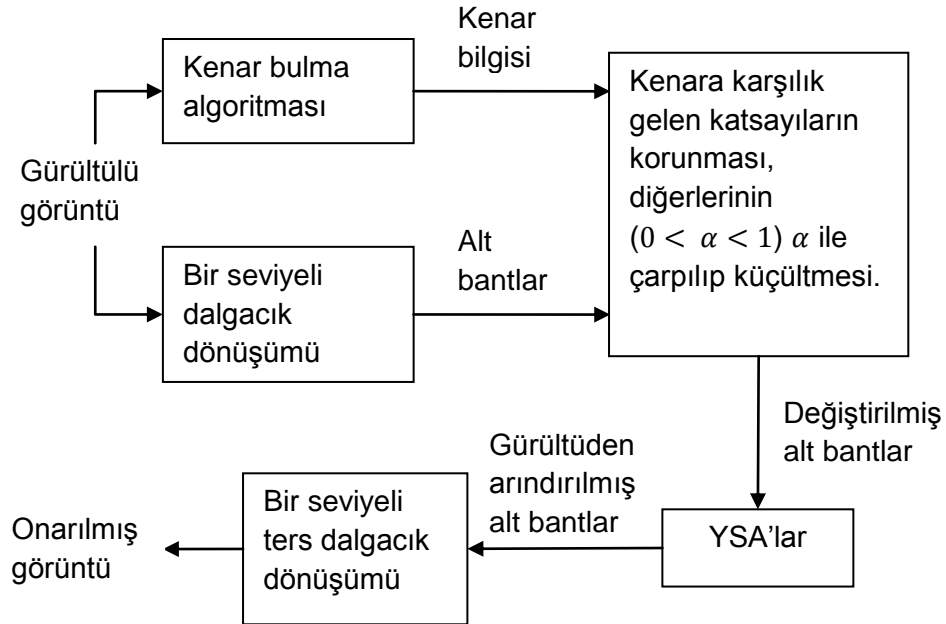


Şekil 7.5: Orijinal ve gürültülü görüntüye karşılık gelen yatay ve dikey alt bant katsayıları.

Şekil 7.5'dan açık olduğu üzere gürültülü görüntünün alt bantlarından kenarları tespit etmek zordur. Bu nedenle, YSA'lar eğitilirken gürültülü görüntülerin kullanılması durumunda YSA'lar yanlış bir şekilde öğretilmiş olabilir. Bu durumun üstesinde gelmek için aşağıda verilen yöntem kullanılabilir:

1. Görüntünün yatay ve dikey kenarları belirlenir.
2. Görüntünün bir seviyeli dalgacık dönüşümü hesaplanıp dört tane alt bant elde edilir.
3. Yatay ve dikey detay katsayılar için aşağıdaki işlem yapılır: Birinci adımda katsayının kenara karşılık geldiği belirlenmişse katsayı değiştirilmez. Aksi halde, $0 < \alpha < 1$ olmak üzere katsayı α ile çarpılır.
4. Her alt bant uygun şekilde öğretilmiş YSA'dan geçirilerek gürültü giderme işlemi gerçekleştirilir.
5. Alt bantlar ters dalgacık dönüşümünden geçirilerek onarılmış görüntü elde edilir.

Bu algoritma şematik olarak Şekil 7.6'de gösterilmiştir.



Şekil 7.6: Kenar korunan gürültü giderme algoritmasının şeması

8. SONUÇLAR

Aşağıda kenar bilgisini dikkate alan ve almayan algoritmaların başarımı tartışılmıştır.

Orijinal görüntüye 0.01 varyanslı beyaz Gauss gürültü eklenmiştir. Kenar bilgisi kullandığında hata yatay detaylar için 42.1367, dikey detaylar için 22.7687 olmuştur.

Kenar koruma yöntemi diyagonal detaylar ve yaklaşım katsayılarına α ile çarpma işlemini gerçekleştirilmemektedir. Bu nedenle iki yöntemdeki diyagonal detaylar ve yaklaşım katsayılarının hataları aynıdır:

TRAINLM-calcjx, Epoch 50/50, MSE 3.78706/0, Gradient 0.492537/1e-010

TRAINLM-calcjx, Epoch 50/50, MSE 428.673/0, Gradient 623.4/1e-010

Kenar bilgisini dikkate almayan yöntemin hatası yatay katsayılar için 45.0896 ve dikey katsayılar için 24.7874'dür. Sonuçlar Tablo 8.1'de özetlenmiştir.

Kenar bulmayı kullanan algoritmaların ağırlıkta daha iyi eğitmesine rağmen PSNR değeri daha küçüktür.

Tablo 8.1: Onarım sonuçları ($\sigma^2 = 0.01$)

	YSA	YSA+Kenar bulma
Yatay (MSE)	45.0896	42.1367
Dikey (MSE)	24.7874	22.7687
PSNR	36.94 dB	36.12 dB

Daha sonra aynı deney 0.05 varyanslı gürültü ile tekrarlanmıştır. Bu sefer kenar bilgisini dikkate alan yöntemin hem öğretim değerleri hem de PSNR değeri daha iyi bulunmuştur.

Tablo 8.2: Onarım sonuçları ($\sigma^2 = 0.05$)

	YSA	YSA+Kenar bulma
Yatay (MSE)	88.0902	64.0341
Dikey (MSE)	28.9024	28.6985
PSNR	31.69 dB	33.61 dB

Görsel sonuçlar Şekil 8.1'de ve 8.2'de verimştir.



(a)



(b)

Şekil 8.1: $\sigma^2 = 0.05$ gürültülü görüntünün iki yöntemle onarılmış halleri. a) YSA+kenar bulma. PSNR=33.61. b) YSA. PSNR=31.69



(a)











(b)

Şekil 8.2: $\sigma^2 = 0.01$ gürültülü görüntünün iki yöntemle onarılmış halleri. a) YSA+kenar bulma. PSNR=36.12. b) YSA. PSNR=36.94

Sunulan yöntemin performansını literatürde var olan yöntemlerin performansı ile karşılaştırmak amacıyla Tablo 8.3'te ve 8.4'te adaptif Wiener filtresi, yumuşak eşikleme ve bu tezde önerilmiş yöntemler ile elde edilen PSNR sonuçları verilmiştir.

Tablo 8.3: Gürültü giderme yöntemlerin görsel görüntü kalitesi cinsinden karşılaştırılması

Gürültü giderme yöntemleri	$\sigma^2 = 0.01$	$\sigma^2 = 0.05$
YSA+kenar bulma		
YSA		
Adaptif Weiner filtresi		
Yumuşak eşikleme		

Tablo 8.4: Gürültü giderme yöntemlerin PSNR cinsinden karşılaştırılması









Gürültü giderme yöntemleri	$\sigma^2 = 0.01$	$\sigma^2 = 0.05$
YSA	36.94 dB	33.61 dB
YSA+Kenar bulma	36.12 dB	31.69 dB
Wiener filtresi	32.72 dB	30.67 dB
Yumuşak eşikleme	32.20 dB	30.54 dB

Sonuçlardan sunulan yöntemlerin hem PSNR hem de görsel olarak daha iyi sonuçlar verdiği görülmektedir. Gürültünün varyansı az olduğunda YSA'ları kullanan yöntem en iyi PSNR değeri vermiştir. Ancak, kenar bilgisini kullanan ve kullanmayan yöntemler arasındaki performans farkı çok büyük değildir.

Gürültü varyansı büyük değerler aldığıında, iki yöntem arasındaki PSNR farkı 3.33 db civarındadır. O halde, az gürültülü giderilmesi için YSA ve yüksek gürültülü YSA+kenar bulma yönteminin kullanılması en iyi sonuçları vermektedir.

YSA ile yapılan eğitim 'Lena' görüntüsüyle gerçekleştirilmiş olmasına rağmen, 'Man' görüntüsünde yapılan test uygulamalarında, Tablo 8.5'te verilen sonuçlar elde edilmektedir.

Tablo 8.5: 'Man' görüntüsünün gürültü giderme yöntemlerin görsel görüntü kalitesi cinsinden karşılaştırılması

Gürültü giderme yöntemleri	$\sigma^2 = 0.01$	$\sigma^2 = 0.05$
YSA+kenar bulma		
YSA		
Adaptif Weiner filtresi		
Yumuşak eşikleme		

Tablo 8.6'da PSNR tabanlı sonuçlar verilmektedir.

Tablo 8.6: 'Man' görüntüsünün gürültü giderme yöntemlerin PSNR cinsinden karşılaştırılması

Gürültü giderme yöntemleri	$\sigma^2 = 0.01$	$\sigma^2 = 0.05$
YSA	36.31 dB	32.86 dB
YSA+Kenar bulma	35.88 dB	34.61 dB
Wiener filtresi	31.48 dB	30.19 dB
Yumuşak eşikleme	33.11 dB	29.76 dB

9. TARTIŞMA VE ÖNERİLER

Gürültü giderme problemi en eski görüntü işleme problemlerinden olmasına rağmen, günümüzde de ilgi çeken, yeni cevapları arayan bir alandır. Gürültü giderme yöntemlerinin amacı, genelde beyaz Gauss gürültüden bir görüntüsü olarak modellenen bir gürültülü görüntüden yok etmektir.

Bu çalışmada, dalgacık uzayında yapay sinir ağları yardımı ile gürültü giderme algoritması önerilmiştir. Çalışmanın temeli olarak YSA'a dayalı doğrusal olmayan bir filtre ile dalgacık bölgesinde gürültülü bir görüntünün giderilmesi' isimli çalışma kullanılmıştır.

Bu algoritmanın, genişletilmiş olarak YSA'ların aktivasyon fonksiyonları değiştirilmiş ve kenar koruma aşaması eklenmiştir. Aktivasyon fonksiyonları değiştirilip YSA'ların öğretim hatası azaltmıştır ve sonuçta gürültü giderme işlemi daha etkili olmuştur.

Kenar koruma yönteminde dalgacık kenar bulma tekniği ile kenarlara uygun olmayan detay katsayıların değerleri azaltılmaktadır. Sonra, alt bandların katsayıları YSA'lardan geçirilmiştir. Kenar korumayı kullanılan yöntemin simülasyonu sonuçları şöyle özetlenebilir. Onarılan görüntünün gürültü varyansı küçük ($\sigma^2 = 0.01$) olduğunda yatay ve dikey detayları işleten YSA'ların öğrenme hatası daha küçüktür. Buna rağmen iki yöntem ile onarılmış görüntülerin PSNR değerleri karşılaştırdığında kenar korumayı kullanmayan tekniğin PSNR değeri biraz daha yüksektir, (+36.94 dB ve +36.12 dB).

Gürültünün varyansı büyükse ($\sigma^2 = 0.05$) hem YSA'ların öğrenme hatası hem de PSNR değeri daha yüksektir (+31.69 dB ve +33.61 dB).

YSAların yardımı ile dalgacık detay katsayıların onarımı tatmin edici sonuçlar vermiştir. Aynı zamanda yaklaşım katsayılarını işleyen YSA'nın öğretim hatası belli bir eşik değerden daha az yapılmamıştır. Bu nedenle yüksek frekanslı verilerin onarımı için YSA'ların kullanılmasını iyi sonuçlar vereceği söylenilebilir.

KAYNAKLAR

Donoho,D., "Ideal spatial adaptation by wavelet shrinkage", *Biometrika*, 3, 425-455,(1994).

Donoho, D.,"De-noising by soft-thresholding", *IEEE transactions on information theory*, 41, 613 – 627, (1995).

Dyakonov, V., "Dalgacıklar. Teori ve pratik", *Solon-R*, 50 – 87, (2004).

Gonzales. R., Woods, R., "Digital Image Processing", 2nd edition, *Pearson Education*, 511– 598, (2002).

Gonzales,R., Woods, R., Eddins, S. "Digital Image Processing in MATLAB", *Pearson Education*, 154 – 205, (2004).

Haykin, S., "Neural Networks.A comprehensive Foundation",2nd edition, *Prentice Hall*, 89 - 110, (1999).

Jacob, N., Martin, A., 2004, "Image Denoising In The Wavelet Domain Using Wiener Filtering" [online],homepages.cae.wisc.edu/~ece533/project/f04/jacob_martin.pdf, (Ziyaret tarihi: 06 Ocak 2011).

Johnstone, I., Donoho, D., "Adapting to smoothness via wavelet shrinkage", *Journal of the Statistical Association*, 90, 1200–1224, (1995).

Kalinkina, D., Varolin, D., 2005. "Görüntü ve video gürültülerini giderme yöntemleri" [online], Görüntü İşleme İnternet dergisi, <http://cgm.computergraphics.ru/content/view/74>, (Ziyaret tarihi: 06 Ocak 2011).

Kharlamov, A. Podlozhnyuk, V. "Image Denoising" [online], http://developer.download.nvidia.com/compute/cuda/1_1/Website/projects/imageDenoising/doc/imageDenoising.pdf, (Ziyaret tarihi: 6 Ocak, 2011).

Kruglov, V., Dli, M., Golunov R., "Bulanık mantık ve yapay sinir ağları",*Fizmatlit*, 53 – 110, (2001).

Kruglov, V., Borisov, V.,"Yapay Sinir Ağları. TeorivePratik", *Goryaçaya Liniya*, 110 – 134, (2002).

Kwon, J., Ro, Y., "Improvement of speckle noise reduction using multi-resolucional coherence measurement in ultrasound image", *Engineering in Medicine and Biology Society*, 4735,(2010).

Li, J., "A Wavelet Approach to Edge Detection", Yüksek Lisans Tezi, *Sam Houston State University*, Texas, 52 – 57, 2003.

- Mallat, S. "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation", **IEEE transactions on pattern analysis and machine intelligence**, 2, 674 – 693, (1989).
- Mallat, S., Hwang, W., "Singularity detection and processing with wavelets", **IEEE transactions on information theory**, 38, 617 – 643, (1992).
- Nunez, J., "Special Issue: Image Reconstruction and Restoration in Astronomy", **Imaging Systems and Technology**, 6, 295 – 296, (1995).
- Osovskiy, S., "YSA'lar ile bilgi işleme", **Finansı i statistika**, 63 – 78, (2002).
- Paik, J., Katsaggelos, A., "Image Restoration Using a Modified Hopfield Network", **Image Processing**, 1, 49 – 63, (1992).
- Rutkovskaya, D., Polinskiy, M., Rutkovsky, L., "YSA'lar, genetik algoritmaları ve bulanık sistemleri", **Goryaçaya Liniya**, 210 – 213, (2006).
- Sahraeian, S., Marvasti, F., Saadati, N., "Wavelet Image Denoising Based on Improved Thresholding Neural Network and Cycle Spinning", **Acoustics, Speech and Signal Processing**, 585 - 588, (2007).
- Slump, C., "Real-Time Image Restoration in Diagnostic X-Ray Imaging, the Effects on Quantum Noise", **Pattern Recognition Methodology and Systems**, 11, 693-696, (1992).
- Smolentsev, N., "Dalgacık Teorisinin Temelleri", **DMK Press**, 34 – 67, (2005).
- Vetterli, M., Chang, S., Yu, B., "Adaptive wavelet thresholding for image denoising and compression", **IEEE Transactions on Image Processing**, 9, 1532–1546, (2000).
- Wei, L., "New Method for Image Denoising while Keeping Edge Information", **Image and Signal Processing**, 1 – 5, (2009).
- White, R., "Restoration of Images and Spectra from the Hubble Space Telescope", **Conf. on Information Sciences and Systems**, Baltimore, 655–660, (1991).
- Yang, Y., "Image Denoising Using Wavelet Thresholding Techniques [online]", <http://homepages.cae.wisc.edu/~ece533/project/f03/yangyang.pdf>, (**Ziyaret tarihi: 6 Ocak 2011**).
- Zhang, S., Salari, E., "Image Denoising Using a Neural Network Based Non-Linear Filter in Wavelet Domain", **Acoustics, Speech, and Signal Processing**, 2, 989 – 992, (2005).

ÖZGEÇMİŞ

1985 yılında Özbekistan'da doğdu. İlk, orta ve lise öğrenimini Kırgızistan'da tamamladı. 2002 yılında girdiği Kırgızistan-Türkiye Manas Üniversitesi, Bilgisayar Mühendisliği Bölümü'nden 2007 yılında mezun oldu.