

**T.C
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**KÜMELEME ALGORİTMALARI KULLANILARAK TIBBİ İMGELERİN
SINIFLANDIRILMASI**

YÜKSEK LİSANS TEZİ

Öznur ERKUŞ

Anabilim Dalı: Elektronik ve Bilgisayar Eğitimi

Programı: Telekomünikasyon

TEMMUZ-2010

**T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**KÜMELEME ALGORİTMALARI KULLANILARAK TIBBİ İMGELERİN
SINIFLANDIRILMASI**

YÜKSEK LİSANS TEZİ

Öznur ERKUŞ

(07231102)

Anabilim Dalı: Elektronik ve Bilgisayar

Programı: Telekomünikasyon

Tez Danışmanı: Doç. Dr. Engin AVCI

Tezin Enstitüye Verildiği Tarih: 15 Temmuz 2010

TEMMUZ-2010

**T.C
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**KÜMELEME ALGORİTMALARINI KULLANILARAK TIBBİ İMGELERİN
SINIFLANDIRILMASI**

YÜKSEK LİSANS TEZİ

Öznur ERKUŞ

(07231102)

**Tezin Enstitüye Verildiği Tarih: 25 Haziran 2010
Tezin Savunulduğu Tarih: 12 Temmuz 2010**

Tez Danışmanı: Doç. Dr. Engin AVCI (F.Ü)

Diğer Jüri Üyeleri: Yrd. Doç. Dr. Davut HANBAY (F.Ü)

Yrd. Doç. Dr. Tuncay SEVİNDİK (F.Ü)

TEMMUZ-2010

ÖNSÖZ

Bu tez çalışması, Fırat Üniversitesi Fen Bilimleri Enstitüsü Elektronik - Bilgisayar Eğitimi Anabilim Dalı Yüksek Lisans Programı'nda hazırlanmıştır.

Bu çalışmada değişik sınıflara ait RNA virüs resimlerinin yapay sinir ağları ile uyarlamalı ağ tabanlı bulanık çıkarım sistemleri kullanarak sınıflandırma ve bulanık c-ortalamlar ile kümeleme başarısı tespiti edilmeye çalışılmıştır. Bu uygulamada esas olan görüntü ön işlem aşamasının başarıyla yapılabilmesi ve ön işlem sonucundan sonra resmin ayırt edici özelliklerinin çıkartılması ve uygun bir düzende yapay sinir ağına, uyarlamalı ağ tabanlı bulanık çıkarım sistemine ve bulanık c-ortalamlar verilmesi olmaktadır. Çalışmanın bu kısmı değişik tekniklerle özelliği çıkartılmış RNA virüs resimlerinde sınıflandırma ve kümeleme başarılarını kıyaslamak olmuştur.

Bu tez çalışmasının gerçekleşmesinde kıymetli zamanını ayırarak yardım ve desteğini esirgemeyen ve uyarıları ile yol gösteren saygıdeğer tez danışmanım Sayın Doç. Dr. Engin AVCI'ya her konuda sabırla yardımcı olan aileme ve Selami ARAS'a desteklerinden dolayı teşekkür eder, minnet ve şükranlarımı arz ederim.

Öznur ERKUŞ
ELAZIĞ - 2010

İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ	I
İÇİNDEKİLER	II
ÖZET	IV
SUMMARY	V
ŞEKİLLER LİSTESİ	VI
TABLolar LİSTESİ	VII
SEMBOLLER LİSTESİ	VIII
KISALTMALAR	IX
1. GİRİŞ	1
1.1. RNA Virüsleri	1
2. GÖRÜNTÜ İŞLEME TEKNİKLERİ	7
2.1. Görüntü Bölütleme Kavramı	7
2.2. Görüntü İşlemenin Temel Adımları	9
2.2.1. Gri Seviye Gösterimi	9
2.2.2. Histogram Gösterimi	9
2.2.3. Resim Eşikleme	10
2.2.4. Canny Kenar Çıkarma Yöntemi	11
2.2.5. Merkez Kenar Değişim Yöntemi	13
2.2.6. Entropi Kavramı	14
2.3. Kullanılan Yöntem ve Aşamaları	15
2.3.1. Ön İşlem Aşaması	15
2.3.1.1. Görüntünün Renkliden Griye Çevrilmesi	16
2.3.1.2. Görüntünün Eşikleme Yöntemi Kullanılarak Bölütlenmesi	16
2.3.1.3. Morfolojik ve Mantıksal İşlemler	16
2.3.2. Özellik Çıkarım Aşaması	17
3. AKILLI SINIFLANDIRMA VE KÜMELEME TEKNİKLERİ	21
3.2. Yapay Sinir Ağları	21

3.2.1. Biyolojik Sinir Ağları ile Yapay Sinir Ağlarının Karşılaştırılması	22
3.2.2. Yapay Sinir Ağlarının Yapıları ve İşlem Elemanları	26
3.2.3. Yapay Sinir Ağlarında Öğrenme ve Öğrenme Algoritmaları	26
3.2. Uyarlamalı Ağ Tabanlı Bulanık Çıkarım Sistemi (UATBÇS)	30
3.2.1. UATBÇS Mimarisi	31
3.2.2. UATBÇS İçin Geri Yayılımlı Öğrenme Algoritması	33
3.3. Bulanık C-Ortalamalar Sınıflandırıcısı (BCO)	36
3.3.1. Matematiksel Temel	36
3.3.2. Eğitim	37
3.3.3. Test Etme	38
4. RNA VİRÜS RESİMLERİNİN AKILLI SINIFLANDIRMA VE KÜMELEME UYGULAMALARI	40
4.1. Doğru Sınıflandırma Başarımının Testi Aşaması	41
4.1.1. YSA İle Adaptif Özellik Çıkarma ve Sınıflama	41
4.1.2. UATBÇS İle Adaptif Özellik Çıkarma ve Sınıflama	43
4.1.3. BCO İle Adaptif Özellik Çıkarma ve Kümeleme	46
5. SONUÇLAR VE TARTIŞMA	49
5.1. Sonuçların Değerlendirilmesi	49
5.2. Öneriler	50
KAYNAKLAR	51
ÖZGEÇMİŞ	

ÖZET

Günümüzde tıp alanında gerçekleştirilen yenilikler ışığında tıbbi görüntüler ve tıbbi görüntülerin işlenmesi önem kazanmıştır. Şimdiye kadar RNA virüs tespiti, bu virüslerin hücre kültüründe üretilip, elektron mikroskobu kullanılarak, göz kararı ile yapılmaktadır. Bu canlıların belirlenmesi laboratuvar ortamında sadece uzman kişi bilgi ve tecrübesine dayalı olarak yapıldığı için çok fazla zaman alır ve ayrıntılı bir bilgi gerektirir. Ayrıca bu şekilde yapılan bütün mikrobiyolojik canlı türlerini tanıma işlemlerinde uzman kişiden ya da laboratuvar ortamından kaynaklanan bazı yanlışlar meydana gelmektedir. Bu yanlış payını en aza indirmek amacıyla bu çalışmada, elde edilen RNA virüs resimlerinin Çoklu Entropi-Yapay Sinir Ağları, Çoklu Entropi-Uyarlamalı Ağ Tabanlı Bulanık Çıkarım Sistemleri ve Çoklu Entropi-Bulanık C-Ortalamalar temelli olarak ayrı ayrı otomatik tespiti gerçekleştirilmeye çalışılmıştır. Bu amaçla, her bir resim için döndürme ve ölçeklendirme yapıp ön işlem aşamasında merkez - kenar değişimi yöntemi kullanılarak görüntülerin uzaklık vektörü elde edilmiştir. Özellik çıkarım ve sınıflandırma aşamalarında ise, ön işlem aşamasında elde edilen görüntülerin her biri için sırasıyla norm, logaritmik enerji ve eşik entropileri olmak üzere 3 adet entropi değeri hesaplanmıştır. Böylece özellik vektörü elde edilmiş olup bu özellik vektörü, sınıflandırma aşamasında YSA ve UATBÇS sınıflandırıcıları ile BCO kümeleyicisinin girişlerine ayrı ayrı verilmiştir. Son olarak test aşamasında ise YSA ve UATBÇS sınıflandırıcıları ile BCO kümeleyicisi algoritmaları kullanılarak gerçekleştirilen sınıflandırma ve kümeleme sonuçlarının doğruluğunu test edilmiştir.

Anahtar Kelimeler: RNA Virüs Resimleri, Merkez - Kenar Değişimi Yöntemi, Entropi, YSA, UATBÇS, BCO, Sınıflandırma, Kümeleme.

SUMMARY

The Classification of Medicine Image by using Clustering Algorithm

Today, in light of innovations carried out in the field of medicine and medical imaging and processing of medical images has become more important. RNA virus identified so far, these viruses produced in cell culture, using electron microscopy, the decision is made by eye. Identification of these creatures in a lab environment only on the basis of expert knowledge and experience takes time and done so much for the detailed information is required. Also recognized in this way the whole process of microbial species from experts or from the lab environment consists of some wrong. To minimize of this error magrin, the it is seperately used Multi-entropy-Artificial Neural Networks, Multi-entropy-Adaptive Network Based Fuzzy Inference System and Multi-entropy-Fuzzy C-Means for classification of obtained RNA virus pictures. Then, rotating and scaling for each image be made during pre-treatment centers - the edges using the method of variation of distance vector images were obtained. In feature extraction and classification phases, the norm, respectively, of the logarithmic energy and entropy threshold entropy value was calculated to be 3each of the images. Thus, feature vector is obtained and this feature vector are given to ANN, ANFIS classifiers and FCM clustering inputs in classification stage. Finally, in the testing phase, the ANN, ANFIS classifiers and FCM clustering of the correct classification performance and the success rate is calculated.

Key Words: RNA Virus Images, Center - Edge Exchange Method, Entropy, ANN, ANFIS, FCM, Classification, Clustering.

ŞEKİLLER LİSTESİ

Sayfa No

Şekil 2.1. Görüntü İşlemedeki Temel Adımlar	8
Şekil 2.2. Gri seviye histogram biçimi	10
Şekil 2.3. Bölmelenen gri seviye histogram biçimleri	11
a) Tek bir eşik değeri	
b) Birden çok eşik değeri	
Şekil 2.4. Kare şekli için merkez kenar değişimi yönteminin uygulanması	13
Şekil 2.5. RNA virüs resmi	16
a) Astrovirus türü RNA virüsü	
b) Virüs görüntüsünün griye çevrilmiş hali	
Şekil 2.6. RNA virüs resmi	16
a) Astrovirus türü RNA virüsü	
b) Bölütlenmiş görüntü	
Şekil 2.7. RNA virüs resmi	17
a) Astrovirus türü RNA virüsü	
b) Canny kenar bulma işleminden sonraki görüntü	
Şekil 3.1. Yapay sinir ağlarının yapısı	21
Şekil 3.2. Biyolojik nöronun şematik yapısı	22
Şekil 3.3. Bir yapay sinir ağı hücresi modeli	23
Şekil 3.4. Yapay sinir ağları için kullanılan eşik fonksiyonları	24
Şekil 3.5. 2 girişli 9 kurallı bir UATBÇS sınıflandırıcı yapısı	31
Şekil 3.6. Bulanık C-Ortalama kümeleyicisi	38
Şekil 4.1. Bu çalışmada kullanılan ÇE-YSA, ÇE-UATBÇS ve ÇE-BCO algoritmalarının yapısı	40

TABLolar LİSTESİ

Sayfa No

Tablo 3.1. Görüntü piksel matrisi	17
Tablo 3.2. $g_m(m,n)$ katlama maskesi	17
Tablo 3.3. $g_n(m,n)$ katlama maskesi	18
Tablo 2.4. RNA virüs görüntülerinin hesaplanan entropi değerleri	18
Tablo 4.1. Çok katmanlı yapay sinir ağının yapısı ve eğitim parametreleri	41
Tablo 4.2. YSA için doğru sınıflandırılmış oranlar	42
Tablo 4.3. YSA' nın RNA virüs görüntülerini tanıma başarımı	43
Tablo 4.4. UATBÇS modelinin yapısı ve eğitim parametreleri	44
Tablo 4.5. UATBÇS için doğru sınıflandırılmış oranlar	45
Tablo 4.6. UATBÇS'nin RNA virüs görüntülerini tanıma başarımı	46
Tablo 4.7. BCO için doğru kümelendirilmiş oranlar	47
Tablo 4.8. BCO'nun RNA virüs görüntülerini tanıma başarımı	48

SEMBOLLER LİSTESİ

a	: İşlem elemanının x aktivitesini ayarlayan reel değerli bir sabit
Ç	: Çıkış
F	: Eşik Fonksiyonu
m	: Görüntüyü oluşturan gri seviye piksellerinin yataydaki boyutları
M_T	: Eşik değeri
n	: Görüntüyü oluşturan gri seviye piksellerinin düşeydeki boyutları
N_C	: Kategorilerin sayısı
p, q ve r	: Lineer çıkış parametreleri
r	: Öklit uzaklığı
s	: İşaret
s_i	: İşaretin i. katsayısı
T	: Eşik değeri
x	: Sınır üzerindeki noktanın yatay bileşeni
X_i	: Giriş
x_m	: Ağırlık merkezi noktasının yatay bileşeni
y	: Sınır üzerindeki noktanın düşey bileşeni
y_m	: Ağırlık merkezi noktasının düşey bileşeni
W_i	: Ağırlık
Q(m,n)	: Görüntünün eğimi
Ø	: Referans açısı
€	: Pozitif bir eşik değeri
δ	: Eşik basamak fonksiyonu
μ_{Aj}(x)	: üyelik derecesi
μ_{Bj}(y)	: üyelik derecesi

KISALTMALAR

BCO	: Bulanık C-Ortalamalar
UATBÇS	: Uyarlamalı Ağ Tabanlı Bulanık Çıkarım Sistemleri
YSA	: Yapay Sinir Ağları
PV	: Picornavirüsler
TV	: Toguvirüsler
FV	: Flavivirüsler
OV	: Orthamyxovirüsler
BV	: Bunyavirüsler
AV	: Astrovirüsler
RV	: Retrovirüsler
CV	: Coronavirüsler
REV	: Reovirüsler
RHV	: Rhabdovirüsler
FLV	: Filovirüsler
PRV	: Paramyxovirüsler
ÇE	: Çoklu Entropi
ÇE-YSA	: Çoklu Entropi-Yapay Sinir Ağları
ÇE-UATBÇS	: Çoklu Entropi-Uyarlamalı Ağ Tabanlı Bulanık Çıkarım Sistemleri
ÇE-BCO	: Çoklu Entropi-Bulanık C-Ortalamalar

1.GİRİŞ

Son yıllarda teknolojik gelişmeler hızla ilerlemektedir. Bu ilerlemelerle birlikte tıbbi alandaki resim işleme tekniklerinin kullanılması yaygınlaşmaktadır. RNA virüslerinden resim işleme tekniklerini kullanarak özellik çıkarma çalışmaları aşağıda maddeler halinde sıralanan nedenlerden dolayı yapılmaktadır.

- RNA virüslerinden kaynaklanan hastalıkların teşhisini kolaylaştırmak için,
- Hastalık teşhisindeki zaman kaybını en aza indirmek için,
- Hastalıklara neden olan RNA virüs çeşidini belirlemek için,
- Teşhis sırasındaki maddi giderleri en aza indirmek için,
- Yanılgı payını en aza indirmek için kullanılır.

Yaptığımız araştırmalar sonucunda daha önce tıp alanında birçok uygulama yapıldığı görülmüştür. Sammouda ve diğ. [1], renkli patolojik akciğer görüntülerinden kanserli hücre çekirdeklerinin belirlenmesi üzerine bir çalışma gerçekleştirmişlerdir. Esgiar ve diğ. [2], kalınbağırsak kanserinin analizi ve teşhisi için yine resim işleme tekniklerinden faydalanmışlardır. Fakat RNA virüslerinin resim işleme tekniklerini kullanılarak özellik çıkarılmasıyla ilgili pek fazla çalışma yapılmamıştır. Bu çalışmalardaki temel amaç resim işleme teknikleri sayesinde RNA virüslerini en iyi, en doğru ve en hızlı bir şekilde tanınmasını sağlayan yöntemleri bulmak ve uygulamaktır.

1.1. RNA Virüsleri

Virüs, canlı hücreleri hasta edebilen mikroskobik taneciktir. Virüsler ancak bir konak hücreyi hasta ederek çoğalabilirler. En temel haliyle bir virüs, kapsit adlı bir protein örtü içinde bulunan genetik malzemeden oluşur [3]. Canlıların üç temel öbeğinden birini oluşturan virüsler, aslında gerçek canlı hücreler ile organik moleküller arasında sınıflandırılır. Yalnızca bitki ve hayvan hücreleri ile bakteriler gibi organizmaların yardımıyla çoğalabilirler [4]. Virüslerdeki kalıtım maddesi bazılarında DNA, bazılarında ise RNA'dır. Bu genetik maddeye kısaca "genom" denir.

Yapısında yönetici molekül olarak RNA taşıyan virüsler, DNA taşıyan virüslerde olduğu gibi sadece canlı organizmaların hücrelerini kullanarak çoğalabilirler. RNA virüslerinin çoğalma işlemi hücrenin sitoplazmasında gerçekleşir.

Virüsleri üzerinde yaşadıkları ve hastalık yaptıkları canlı grubuna göre; taşıdıkları nükleik asitlere göre ayırıp inceleyebiliriz. Üzerinde yaşadıkları ve hastalık yaptıkları canlı grubuna göre 3'e ayırırız:

- 1. Bitkisel Virüsler:** Bitkilerde hastalık yaparlar. Kalıtım maddesi olarak genellikle RNA bulundurlar. Tütün, patates, marul, salatalık “mozaik virüsleri” örnek verilebilir.
- 2. Hayvansal Virüsler:** Sadece insanlar ve hayvanlarda hastalık yaparlar. Bazıları DNA; bazıları RNA bulundurur. Grip, kızamık, kabakulak, suçiçeği, sarıhumma, çocuk felci, uçuklar, siğiller ve AIDS hayvansal virüslerin sebep olduğu hastalıklardır. Bugün kanserin dahi sebepleri arasında virüsler sayılmaktadır.
- 3. Bakteriyofajlar:** Bakterilerin içinde yaşarlar ve onları öldürürler. Genellikle DNA bulundurlar. Virüslerin hastalık yaptığı canlı çeşitleri farklı olduğu gibi bir canlının değişik dokularında yaşayan virüsler de farklı olabilmektedir. Özetle virüsler, canlı hücreler içerisinde canlılık faaliyeti gösterebilirler.

Virüsleri taşıdıkları nükleik asitlere göre gruplandırılırsa:

a) DNA Virüsleri: Yönetici molekülleri DNA olan virüslerdir. Hayvanlarda yaşayan virüslerin çoğunluğu DNA virüsleridir. Örneğin, çiçek virüsü, suçiçeği virüsleri DNA virüsleridir. Bakteriyofajlar da DNA virüsüdür.

b) RNA Virüsleri: Yönetici molekülü RNA olan virüslerdir. Bazı hayvan hücreleri ile bitki hücrelerinde yaşayan virüsler RNA virüsleridir. Örneğin, tütün mozaik virüsü, grip, çocuk felci, kızamık, kuduz, kabakulak, sarıhummaya yol açan virüsler RNA virüsleridir.

RNA virüslerinin replikasyon evresi, virüslerin bazı özelliklerine göre farklılık göstermekte ve dört gruba ayrılmaktadır.

- **Çift İplikli RNA Virüsleri:** Virüs konak hücrenin dışında özgür bir tanecik iken virion adını alır. Virion içerisinde RNA polimeraz, nükleotitfosforilaz ve guanidil ile metil transferaz aktiviteleri mevcuttur. Her çift iplikli RNA segmentinin negatif iplikçiği kalıp olarak kullanılarak tek iplikli pozitif RNA molekülleri sentez edilir. Bu RNA'lar hem mRNA görevi yaparlar, hem de yeni üretilecek RNA için kalıp fonksiyonu görürler. Böylece yeni çift iplikli yeni üretilecek RNA'lar sentezlenmiş olur.

- **Pozitif Polariteli Tek İplikli RNA Genomu İçeren Virüsler:** Virion içinde RNA

polimeraz enzimi yoktur. Parental RNA, mRNA gibi görev yapar. Virüs hücre içine girer girmez viral RNA poliribozomlara giderek ilk protein sentezini yapar. Bu protein oldukça büyük bir moleküldür ve fonksiyonel proteinlerin oluşması için proteazlar ile ufak parçalara bölünmesi gerekir. Replikasyon sırasında pozitif iplikçik üzerinde yeni negatif iplikçikler sentez edilir. Ayrılan her negatif iplikçik pozitif RNA'ların sentezi için kalıp görevi görür. Negatif kalıptan ayrılan pozitif iplik, hem mRNA görevi yapar hemde virion RNA'sını oluşturur [4-6].

- **Negatif Polariteli Tek İplikli RNA Genomu İçeren Virüsler:** Her bir negatif polariteli segment, virion içerisinde bulunan RNA polimeraz enzimi tarafından pozitif RNA iplikçığı şeklinde kopyalanır. Aynı negatif kalıp birden fazla pozitif iplikçik sentez edebilir. Bu pozitif RNA'lar hem mRNA görevi yapar, hem de yeni negatif RNA iplikçiklerinin sentezi için kalıp görevi yapar [4-6].

- **Reverse Transkriptaz İçeren Virüsler:** Bu virüsler tek iplikli pozitif RNA içerirler. Ayrıca virion içinde reverse transkriptaz, Rnaz-H ve endonükleaz enzimleri bulunur. Reverse transkriptaz enzimi tersine transkripsiyon yapan anlamındadır. Reverse transkriptaz enzimi RNA'yı kalıp olarak kullanarak DNA sentezler. Böylece RNA: DNA hibridi oluşur. Rnaz-H bu hibridi parçalayarak negatif DNA'ya komplementer bir pozitif DNA sentez edilerek çift iplikçikli DNA oluşumuna yardımcı olur. Bu çift iplikçikli DNA molekülü hücre çekirdeği kromozomuna entegre olur ve pozitif RNA ile mRNA'yı oluşturur. mRNA'lar virion proteinlerini oluştururken, pozitif RNA da yeni virüsün virion RNA'sını oluşturur. [4-6]

Literatürde bulunan bazı RNA virüs türleri aşağıdaki gibi özetlenebilir:

Picornavirüsler (PV): Picornaviridia ailesinden Picornavirüsler 25-30 nm. boyutlarında, tek iplikli, artı kutuplu RNA, zarflı, ikozahedral protein kılıflı virüslerdir. Bu ailede Enterovirus, Hepatovirus, Rhinovirus alt aileleri bulunur. Bu virüs ailesinin sebep olduğu hastalıklar Aseptik menenjit, Paralitik poliomiyelit, Aseptik menenjit, miyoperikardit, Aseptik menenjit, raş, Akut hepatit, Soğuk algınlığıdır.

Toguvirüsler (TV): Togaviridae ailesinden olan Toguvirüsler 60-70 nm. boyutlarında, tek iplikli, artı kutuplu RNA, zarflı, ikozahedral protein kılıflı virüslerdir. Bu ailede Alphavirus ve Rubivirus alt aileleri bulunur. Bu virüs ailesinin sebep olduğu hastalıklar Sivrisinek kaynaklı ensefalitler, Kızamıkçık, hafif rash, konjenital fetal defektlerdir.

Flavivirüsler (FV): Flaviviridae ailesinden olan Flavivirüsler 45-55 nm. boyutlarında,

tek iplikli, artı kutuplu RNA, zarflı, ikozahedral protein kılıf, sitoplazmada replike olan virüslerdir. Bu ailede Flavivirus ve Hepacivirus alt aileleri bulunur. Bu virüs ailesinin sebep olduğu hastalıklar Sivrisinek kaynaklı ateş, hepatit, sarıhumma, Sivrisinek kaynaklı hemorajik ateş, Dang ateşi, Ensefalit ve Hepatit C (akut/kronik), karaciğer kanseridir.

Orthomyxovirüsler (OV): Orthomyxoviridae ailesinden olan Orthomyxovirüsler 80-120 nm. boyutlarında, 8 parçalı, tek iplikli, eksi kutuplu RNA, zarflı, sarmal protein kılıflıdır. Bu ailenin Influenzavirus isimli alt ailesi vardır. Bu virüs ailesinin sebep olduğu hastalıklar grip; ateş, myalji, halsizlik, öksürük, pnömonidir.

Bunyavirüsler (BV): Bunyaviridae ailesinden olan Bunyavirüs'ler 100 nm. boyutlarında, zarflı, sarmal protein kılıflıdır. Bu ailenin Hantavirus, Bunyavirus ve Phlebovirus alt aileleri vardır. Bu virüs ailesinin sebep olduğu hastalıklar kemirici kaynaklı pulmoner hastalıklar, renal sendromlu hemorajik ateş (HUS), sivrisinek kaynaklı ensefalit, hemorajik ateşlerdir.

Astrovirüsler (AV): Astroviridae ailesinden olan Astrovirüsler 110-130 nm. boyutlarında, zarflı, sarmal protein kılıf, viral zarf içinde bergel ribozomlar bulunur. Bu ailenin insan için en önemli virüsleri Human astrovirus'dür. Bu virüs ailesinin sebep olduğu hastalıklar Gastroenteritdir.

Retrovirüsler (RV): Retroviridae Ailesi Retrovirus'lar 90 - 120 nm çapında, tek iplikçikli RNA' ya sahip, karmaşık yapılı, zarflı virüslerdir. Bu ailede Oncovirinae, Lentivirinae ve Spumavirinae alt aileleri bulunur. Bu ailenin insan için en önemli virüsleri Human T Lenfotropik Virüs (HTLV) ve AIDS etkeni olan Human Immunodeficiency Virüs (HIV) bulunur. Bu virüslerin en önemli özelliği RNA'ya bağımlı DNA polimeraz, yani Revers Transkriptaz enzimlerinin bulunmasıdır.

Coronavirüsler (CV): Coronavirusler, yaklaşık 120nm büyüklüğünde, tek iplikli, pozitif polariteli RNA (mRNA) içeren, zarflı, heliksel nükleokapsid içeren virüslerdir. Zarf üzerinde boyları 20 nm uzunluğuna ulaşan peplomerler bulunur. Bu peplomerler virüsün taç görünümü kazanmasını sağlar. Zarf üzerindeki yüzey proteinlerinden biri, yüksek glikozilleşme gösteren, yaklaşık 20nm uzunluğundaki Spike (S) proteinidir. Uzunluklarından dolayı bu S Glikoproteini virionun en dıştaki parçasıdır. S Glikoproteini, virüsün özgül olduğu hücreye tutunmasını ve hücre membranının virüs zarfıyla füzyona uğramasını sağlar. Ayrıca konakçının immun sisteminin virüsü tanımasında birincil unsurdur. Bu virüs ailesinin sebep olduğu hastalıklar soğuk algınlığına benzeyen hafif hastalıklardır [6].

Reovirüsler (REV): Bu virüs 60-80 nm çapındadır ve ikozahedral, merkezleri ortak olan iki tabakalı kapsite sahiptir. (Rotaviruslar üç tabakalı bir yapıya sahiptir.) Rotavirusların 132 kapsomeri vardır, zarfsızdırlar. Dış kapsiti olmayan tek tabakalı virus parçacıkları 50-60 nm çapındadır. Parçacıkların iç kısmındaki kor bölgesi (çekirdek) 33-40 nm çapındadır. Çift tabakalı parçacık virüsün tam infeksiyöz şeklidir. Reoviruslar ısıya, pH 3.0-9.0'a ve lipid çözücülere karşı kararlıdırlar, fakat %95'lik etanol, fenol ve klor ile inaktive olurlar. Proteolitik enzimlerle sınırlı muamele infektiviteyi artırır. Reoviruslar; çift sarmallı, parçalı RNA içeren orta büyüklükte viruslardır. Bu virus ailesinde, dünyanın her tarafında yeni doğan gastroenteritlerinin en önemli etkeni olan insan rotavirusları vardır. Akut gastroenterit, toplum sağlığı açısından önemli olan çok yaygın bir hastalıktır. Gelişmekte olan ülkelerde yılda 3.5 milyon kadar çok okul öncesi çocuğun ölümüne sebep olduğu tahmin edilmektedir. ABD'de akut gastroenterit, aile içi hastalık sebebi olarak akut solunum yolu hastalıklarından sonra ikinci sıradadır [6-7].

Rhabdovirüsler (RHV): Rhabdoviridae ailesinden olan Rhabdovirüsler 180x75 nm. boyutlarında, mermi şeklinde, tek iplikli, eksi kutuplu RNA, sarmal protein kılıf, zarflı yapıdadırlar. Bu ailede Vesiculovirus ve Lyssavirus alt aileleri bulunmaktadır. Bu ailenin insan için en önemli virüsleri Rabiesvirus, Marburg virus ve Ebolavirusleri bulunmaktadır. Bu virüs ailesinin sebep olduğu hastalıklar kuduz ve hemorajik ateştir [8].

Filovirüsler (FLV): Filoviridae ailesinden olan Filovirüsler 80 nm. boyutlarında, filament biçimli, zarflı, sarmal protein kılıflıdırlar. Bu ailenin insan için en önemli virüsleri Marburg virus ve Ebolaviru bulunmaktadır. Bu virüs ailesinin sebep olduğu hastalık hemorajik ateştir.

Paramyxovirüsler (PRV): Paramyxoviridae ailesinden olan Paramyxovirüsler 150 nm. boyutlarında, tek iplikli, eksi kutuplu RNA yapısındadırlar. Bu ailede Paramyxovirus, Morbillivirus ve Pneumovirus alt aileleri bulunmaktadır. Bu ailenin insan için en önemli virüsleri Parainfluenza virus tip 1, 2, 3; 4, 5, Mumps (Kabakulak) virus, Measles (Kızamık) virus ve Respiratory syncytial virusleri bulunmaktadır. Bu virüs ailesinin sebep olduğu hastalıklar Soğuk algınlığı, bronşiyolit, pnömoni (4 tip), Kabakulak, aseptik menenjit (nadirenorşit ve ensefalit), Kızamık, ateş, rash, pnömoni (nadiren ensefalit, SSPE) ve Soğuk algınlığı (adult), bronşiyolit, pnömoni (YD) dir.

Şimdiye kadar RNA virüs tespiti, bu virüslerin hücre kültüründe üretilip, elektron mikroskobu kullanılarak, göz kararı ile yapılmaktadır. Bu durumda uzman kişinin yanılğı payı artmaktadır. Bu yanılğı payını en aza indirmek amacıyla bu çalışmada, elde edilen

RNA virüs resimlerinin Çoklu Entropi (ÇE), Yapay Sinir Ağları (YSA), Uyarlamalı Ağ Tabanlı Bulanık Çıkarım Sistemleri (UATBÇS) ve Bulanık C-Ortalamlar (BCO) temelli olarak otomatik tespiti gerçekleştirilmeye çalışılmıştır. Bu otomatik RNA virüs tespit sistemi, yapısal olarak dört bölümden oluşmaktadır. Bunlar, sırasıyla ön işlem, özellik çıkarımı, sınıflandırma ve test aşamalarıdır. Sırasıyla ön işlem aşamasında, merkez - kenar değişimi yöntemi kullanılarak bu görüntülerin merkez piksellerinden, görüntüdeki nesnelerin kenarları üzerindeki her bir noktaya olan öklit uzaklıkları hesaplanarak, bu uzaklık değerlerinden bir uzaklık vektörü elde edilmiştir. Özellik çıkarım ve sınıflandırma aşamalarında ise, ön işlem aşamasında elde edilen görüntülerin her biri için sırasıyla norm, logaritmik enerji ve eşik entropileri olmak üzere 3 adet entropi değeri hesaplanmıştır. Böylece özellik vektörü elde edilmiş olup bu özellik vektörü, sınıflandırma aşamasında YSA ve UATBÇS sınıflandırıcıları ile BCO kümeleyicisinin girişlerine ayrı ayrı verilerek başarımlarını karşılaştırılmıştır. Son olarak test aşamasında ise YSA ve UATBÇS sınıflandırıcıları ile BCO kümeleyicisi algoritmaları kullanılarak gerçekleştirilen sınıflandırma sonuçlarının doğruluğunu test edilmektedir.

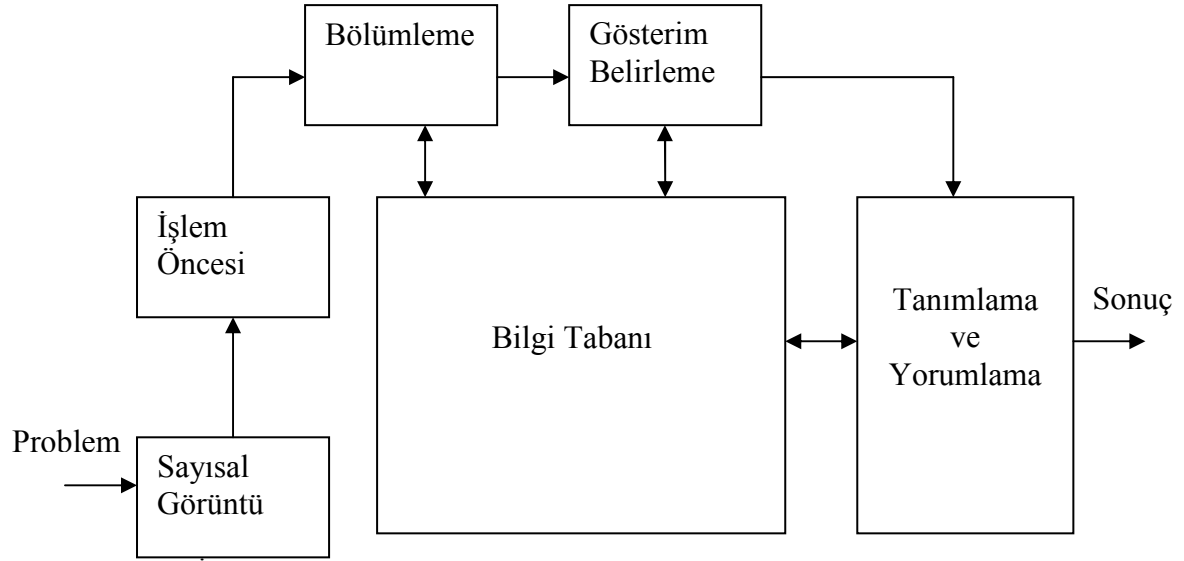
2. GÖRÜNTÜ İŞLEME TEKNİKLERİ

Bu bölümde Bölüm 1’de bahsedilen daha önceki otomatik RNA virüsü tanıma çalışmalarında ön işlem ve özellik çıkarım aşamalarında kullanılan temel morfolojik işlemler tanıtılacaktır.

2.1. Görüntü Bölütleme Kavramı

Görüntüyü iyileştirmeden farklı olarak görüntü bölütleme, görüntünün analizi ile ilgili bir problem olup, görüntü işlemenin gösterim ve tanılama aşamalarına görüntüyü hazırlama işlemidir. Bu anlamda görüntü bölütleme, bir görüntüyü her biri içerisinde farklı özelliklerin tutulduğu anlamlı bölgelere ayırmak olarak ifade edilebilir. Örneğin bu özellikler, görüntü içerisindeki benzer parlaklıklar olabilir ve bu parlaklıklar ilgili görüntünün farklı bölgelerindeki nesneleri temsil edebilir. Görüntü içerisinde aynı parlaklıklara sahip nesne parçacıklarının belirlenmesi, sınıflandırma ve tanılama amacı için kullanılabilir. Unutulmaması gereken bir nokta da görüntülere uygulanabilecek genel bir bölütleme yönteminin bulunmaması ve hiçbir bölütleme yönteminin mükemmel olmamasıdır. Başka bir ifadeyle, görüntü iyileştirme problemlerinde olduğu gibi görüntü bölütleme için tasarlanan yöntemler ve bu yöntemlerin başarımları, görüntüden görüntüye ve uygulamaya bağlı olarak değişiklik gösterir. Görüntü bölütlemedeki temel mantık sınıflar arası benzerliği en az, sınıf içi benzerliği ise en fazla yapmaktır [9].

Genel olarak gri-ton görüntüler için bölütleme algoritmaları, gri seviye değerlerinin iki temel özelliğinden birine dayalı olarak tasarlanırlar. Bu özellikler, görüntü içerisindeki gri seviye değerlerindeki süreksizlik ve benzerlik ile ilgilidir. Gri seviye değerlerindeki süreksizliklere göre görüntü bölütleme problemindeki temel yaklaşım, gri seviyelerdeki ani değişiklere dayalı olarak bir görüntüyü bölmelemektir ki bu aşama bir görüntüdeki kenar ve ayrıntıların belirlenmesine karşı düşer. Gri seviye değerlerindeki benzerliklere göre görüntü bölütleme ise, bölge bölütlemesi olarak adlandırılır ve eşikleme, büyütme ve yarma - kaynaştırma işlemlerine dayalı olarak gerçekleştirilir. Piksellerin gri seviye değerlerindeki benzerlik veya farklılıklara dayalı olarak bir görüntünün bölütlenmesi kavramı hem durağan hem de dinamik (zamanla değişen) görüntülere uygulanabilir [10].



Şekil 2.1. Görüntü İşlemedeki Temel Adımlar

Görüntü işlemedeki ilk adım sayısal görüntü elde etmektir. Bunun için bir sensör ve sensör çıkışındaki işaretinde sayısallaştırılması gerekir. Sensörün özelliği ve oluşturduğu görüntü uygulamaya göre belirlenebilir. Sayısal görüntüyü elde ettikten sonra ikinci adım işlem öncesi aşamasıdır. İşlem öncesi aşamasından sonraki aşamalarda daha iyi sonuç elde edebilmek için resim üzerinde iyileştirme işlemleri gerçekleştirilir. Bu kontrast, genişletme, gürültü yok etme gibi işlemler olabilir. Üçüncü aşamada bölümlendirme işlemi yapılır. Bölümlendirme resmi kendi bünyesinde parçalara ayırmaktır. Otonom bölümlendirme sayısal görüntü işlemede en zor işlemlerden biridir. Diğer bir yandan da yanlış veya düzensiz yapılan bölümlendirme işlemlerin başarılı sonuca gitmesinin gecikmesine veya hiç başarılı olamamasına bile neden olabilir. Bölümlendirme sonrası oluşan çıktı işlenmemiş veridir. Bu verilerin bilgisayarın işleyebileceği bir hale getirilmesi gerekir. Bu noktada verilmesi gereken karar verinin resimdeki nesneye ait bir dış sınırının mı yoksa iç hattının mı alınacağıdır [11].

Dış sınırları veri olarak alma işlemi nesnenin kenar, köşe ve büküm noktalarının önemli olduğu ve resimdeki nesnenin dış hatlarıyla ilgilenildiği zaman kullanılır. Eğer veri olarak resimdeki nesnenin iç hatlarıyla ilgileniliyorsa yani nesnenin kendini tanıtan kısmının kenarlarda değil de resmin iç yapısında saklı olduğu durumlarda, nesnenin iskelet yapısı oluşturularak nesnenin iç kısımları vurgulanır ve iç sınırları veri olarak alınmış olur. Bazı uygulamalarda hem dış sınırlar hem de iç sınırların bir arada kullanılması gerekebilir [11].

Gösterim belirleme aşaması işlenmemiş veriyi kendinden sonraki işlem aşamalarında değerlendirilebilecek hale getirir. Bu aşamada aynı zamanda elde edilen verinin hangi özelliklerinin kullanılacağını belirleyen bir metot belirtilir. Özellik seçimi ile resimdeki birbirinden farklı olan ve belirli özelliklerine göre birbirinden ayrılacak nesnelerin birbirinden ayrılırken hangi özelliğine bakılarak bu ayırma işleminin yapılacağı belirlenir [11].

Son aşama ise tanıma ve yorumlama aşamasıdır. Tanıma işleminde resim içindeki her nesneye tanıma işlemi ile belirlenen özelliklerine göre bir etiket atanır. Yorumlama aşamasında etiket atanan nesneler arasındaki benzerliğe bakılarak nesnenin ne olduğu veya sonuç olarak ne gibi bir karar verilmesi isteniyorsa bu karar veriliyor. Yorumlama etiket atanmış özelliklere bir anlam yükler [11].

2.2. Görüntü İşlemenin Temel Adımları

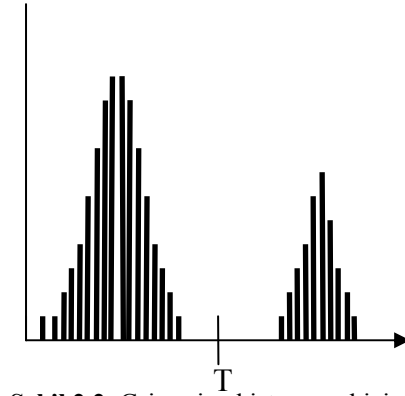
Sayısal görüntü işleme çok geniş donanım, yazılım ve teorik altyapı alanını kapsar. Şekil 2.1. de görüntü işleme ile bir sonuç üretilene kadarki aşamalar görülmektedir.

2.2.1. Gri Seviye Gösterimi

Renkli görüntülerin bilgisayarda daha iyi işlenebilmesi için görüntüye 0 ve 255 arasında sayısal değer atar.

2.2.2. Histogram Gösterimi

Histogram gösterimi, görüntüdeki gri seviye dağılımlarını göstermek için kullanılır. Örneğin, koyu bir arka plan üzerinde açık renkli nesnelerden oluşan $f(i, j)$ görüntüsüne ilişkin gri seviye histogramı Şekil 2.2'deki biçime sahip olacaktır [11].

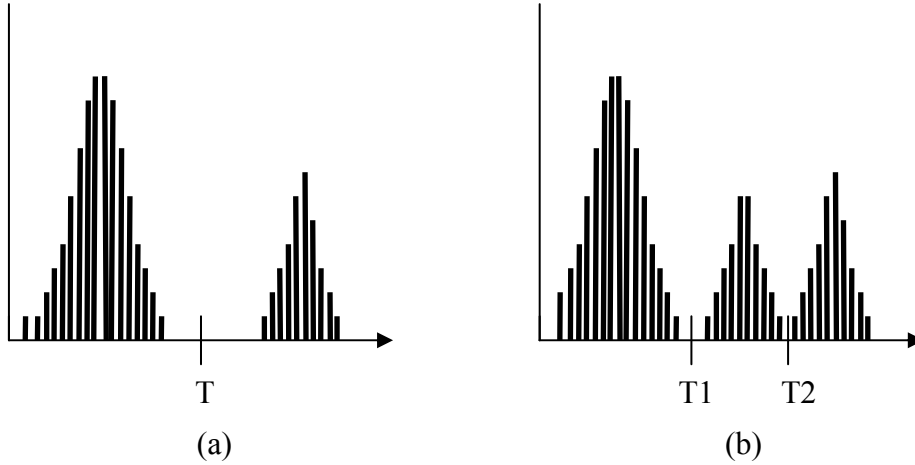


Şekil 2.2. Gri seviye histogram biçimi

2.2.3. Resim Eşikleme

Eşikleme, görüntü bölütleme amacı için kullanılan en önemli yaklaşımlardan birisidir [13]. Eşikleme işleminden amaç, görüntü içerisindeki nesneleri görüntü arka planından ayırmaktır.

Eşikleme için, görüntüdeki gri seviye dağılımlarını gösteren görüntü histogramından faydalanılır. Örneğin, koyu bir arka plan üzerinde açık renkli nesnelerden oluşan $f(i, j)$ görüntüsüne ilişkin gri seviye histogramı Şekil 2.3(a)'daki biçime sahip olacaktır. Bu histograma göre, nesnelere ve arka plana ait pikseller olmak üzere, görüntüyü iki ana grupta değerlendirmek mümkündür. Bu durumda nesneleri arka plandan ayırmak için en kolay yol, histogramdan göreceli olarak belirlenen bir T eşik değeri ile görüntüdeki piksel değerlerini karşılaştırmak olacaktır. Buna göre, görüntüdeki herhangi bir (i, j) pikseli için; $f(i, j) > T$ ise (i, j) pikseli nesneye ait bir nokta, $f(i, j) \leq T$ ise (i, j) pikseli arka plana ait bir nokta olacaktır. Diğer taraftan, görüntüye ilişkin histogram Şekil 2.3(b)'deki gibi ikisi nesneye biri de arka plana ait olmak üzere üç gri seviye grubundan oluşabilir. Buna göre görüntüdeki herhangi bir (i, j) pikseli için; $T_1 < f(i, j) \leq T_2$ aralığındaki pikseller bir nesneye, $f(i, j) > T_2$ aralığındaki pikseller diğer bir nesneye ve $f(i, j) \leq T_1$ aralığındaki pikseller de görüntü arka planına karşı düşecektir [12].



Şekil 2.3. Bölmelenen gri seviye histogram biçimleri
a) Tek bir eşik değeri b) Birden çok eşik değeri

2.2.4. Canny Kenar Çıkarma Yöntemi

Kenar çıkarma, görüntü işlemede önemli konulardan birisidir. Bir görüntüdeki kenar, aydınlatma veya yüzey yansımaları gibi bir görüntünün fiziksel görünüşünde oluşan önemli bir değişime karşı düşer ki bu değişim kendisini parlaklık, renk ve doku olarak gösterir.

Canny kenar çıkarma yöntemi, çok adımlı bir kenar çıkarma tekniğidir. Bu yöntemin temel amacı görüntüdeki nesnelerin kenarlarını iyi bir şekilde ortaya çıkarırken aynı zamanda görüntüdeki gürültüleride yok etmektir. Canny algoritmasının içeriği aşağıda maddeler halinde sıralanmıştır:

Boyutu (m,n) olan bir görüntü, $f(m,n)$ bir Gauss filtresinden geçirilerek yapısındaki gürültüler ve istenmeyen detaylar azaltılır. Bu işlem Denklem 2.1. de ifade edilmiştir.

$$g(m,n) = G_{\sigma}(m,n).f(m,n) \quad (2.1)$$

burada $G_{\sigma} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{m^2 + n^2}{2\sigma^2}\right)$ şeklinde bir Gauss filtre fonksiyonudur.

Canny kenar çıkarma algoritmasını gerçekleştirmek için Sobel kenar belirleyici yöntemi kullanılmıştır. Buna göre ele alınan filtrelenmiş $g(m,n)$ görüntüyü oluşturan gri seviye pikselleri boyutları m ve n olmak üzere iki boyutlu bir matris olarak düşünülür. Bu görüntü piksel matrisi küçük bir resim için Tablo 2.1. de verilmiştir. Burada m , bu matrisin yataydaki boyutu, n ise düşeydeki boyutudur.

Tablo 2.1. Görüntü piksel matrisi

2	4	5	4	2
4	9	12	9	4
5	12	15	12	5
4	9	12	9	4
2	4	5	4	2

Sobel kenar belirleme yönteminde bu görüntünün düşeydeki eğimi ($g_m(m,n)$) ve yataydaki eğimi ($g_n(m,n)$)'i bulabilmek için biri yatay biride düşey olmak üzere her biri 3 x 3 boyutunda iki adet katlama maskesi kullanılmaktadır. Bu katlama maskeleri Tablo 2.2. ve Tablo 2.3. de verilmiştir.

Tablo 2.2. $g_m(m,n)$ katlama maskesi

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

Tablo 2.3. $g_n(m,n)$ katlama maskesi

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

Tablo 2.1. deki görüntü piksel matrisi ayrı ayrı Tablo 2.2. ve Tablo 2.3. deki yatay ve düşey katlama maskeleri ile çarpım işlemine tabi tutulur. Elde edilen değerler yatay ve düşey için ayrı ayrı toplanarak bu eksenlerdeki eğimler olan $g_m(m,n)$ ve $g_n(m,n)$ bulunur.

Eğim operatörlerinden Sobel kullanılarak filtrelenmiş görüntü $g(m,n)$ 'in eğimi aşağıdaki gibi hesaplanır.

$$M(m,n) = \sqrt{g_m^2(m,n) + g_n^2(m,n)} \quad (2.2)$$

ve

$$\theta(m, n) = \tan^{-1}(g_n(m, n) / g_m(m, n)) \quad (2.3)$$

Burada $\theta(m, n)$, görüntünün eğimidir.

Eşik değeri M_T aşağıdaki gibi bulunur.

$$M_T(m, n) = \begin{cases} M(m, n) & M(m, n) > T \quad \text{ise} \\ 0 & M(m, n) \leq T \quad \text{ise} \end{cases} \quad (2.4)$$

Burada T değeri, görüntü gürültüden ve istenmeyen ayrıntılardan temizlendikten sonra görüntüdeki nesnelerin kenarlarını en iyi şekilde görebilmemizi sağlayacak şekilde seçilen piksel değeridir. Bu şekilde Canny kenar çıkarma algoritması kullanılarak bir görüntüdeki nesnelerin kenarları elde edilir [14].

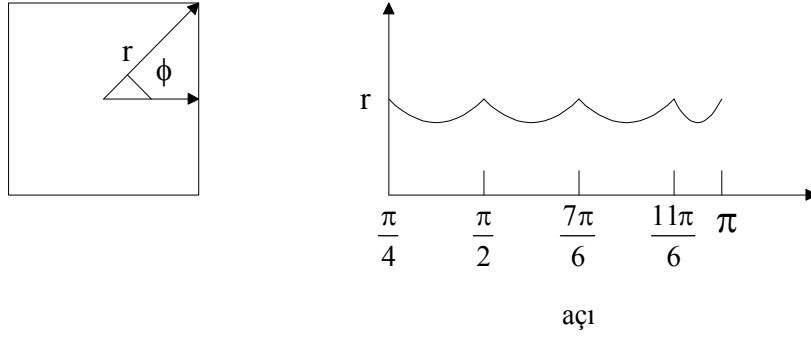
2.2.5. Merkez Kenar Değişim Yöntemi

Merkez kenar değişim yöntemi, iki boyutlu bir sınırın bir boyutlu temsidir. Merkez kenar değişimi, $r(\phi)$ fonksiyonunun bir çizimi olarak tanımlanabilir. $r(\phi)$ fonksiyonundaki r , şeklin ağırlık merkezi ile bu şeklin sınırları üzerinde bulunan herhangi bir nokta arasındaki öklit uzaklığını temsil eder. Denklem 2.5. de öklit uzaklığının nasıl hesaplandığı gösterilmiştir.

$$r = \sqrt{(x - x_m)^2 + (y - y_m)^2} \quad (2.5)$$

Burada x sınır üzerindeki noktanın yatay bileşeni, x_m ağırlık merkezi noktasının yatay bileşeni, y sınır üzerindeki noktanın düşey bileşeni ve y_m ise ağırlık merkezi noktasının düşey bileşenidir.

ϕ ise Şekil 2.4. de gösterildiği gibi sınır üzerindeki herhangi bir noktanın yatay konumla yaptığı referans açığı temsil etmektedir. Bu açığa göre şeklin ağırlık merkez noktasıyla bu şeklin sınırları üzerindeki her bir nokta arasındaki öklit uzaklığının çizimi merkez kenar değişimi işaretini verir [13]. Şekil 2.4. den de görüldüğü gibi bu işaret bir kare şekli için belirli peryotlarda tekrarlanan bir dairesel çizimdir.



Şekil 2.4 Kare şekli için merkez kenar değişimi yönteminin uygulanması

2.2.6. Entropi Kavramı

İşaret işlemede ve özellikle haberleşme sahasında ilk olarak Shannon tarafından kullanılan entropi kavramı, entropi tabanlı özellikler verilen bir işaret gösteriminin özellikleri ile ilgili bilgileri tanımlar [16]. Entropi kavramı, bir sistemin düzenliliğini ölçmek amacı ile termodinamik fiziğinde iyi bilinen bir kavramdır. Entropi ölçüm yöntemi herhangi bir işaretin düzensizlik derecesini ölçmek için ideal bir araç olarak görünmektedir [16]. Ayrıca, entropi kavramı bir ortamdaki bilginin ortalama miktarını ölçmek amacı ile de kullanılmaktadır [17].

Son yıllarda, entropi işaret işleme alanında da yaygın olarak kullanılan bir kavram haline gelmiştir. Bunlardan bazıları; JPEG görüntü kodlamada oluşabilecek gürültüyü tespit etmek [19], entegre devre teknolojisinde karbon üzerindeki elektron enerji kaybını bulmak [18], biyomedikal NMR işaretlerinin dalgacık dönüşüm uzayından istatistiksel bilgilerini elde ederek işaretin gürültüsüz şeklini kestirmek, EEG işaretlerinin spektral entropilerini kullanarak anestezi derinliği tahmin etmek ve MEG biyomedikal görüntülerin yeniden yapılandırılmasında Norm entropi metodu kullanarak yüksek kalitede MEG görüntüleri elde etmek gibi.

İşaret işlemede yaygın olarak kullanılan entropi hesaplama türleri : Shannon, Norm, Eşik (Threshold), Logaritmik Enerji ve Sure yöntemleridir [20]. Bunun dışında diğer entropi hesaplama teknikleri de bulunmaktadır. Aşağıdaki entropi yöntemlerinde, s işareti ve s_i de işaretin i . katsayısını göstermektedir.

İşaret işleme alanında yaygın kullanılan bu entropi türleri ve bu entropi türlerine ait denklemler aşağıda verilmiştir.

Shannon Entropisi

$$E(s) = -\sum_i s_i^2 \cdot \log_2(s_i^2) \quad \text{ve} \quad \log(0) = 0 \text{ kabul edilerek} \quad (2.6)$$

Norm Entropisi

$$E(s) = \sum_i |s_i|^p \quad \text{ve} \quad 1 \leq p < 2 \quad (2.7)$$

Logaritmik Enerji Entropisi

$$E(s) = \sum_i \log_2(s_i^2) \quad \text{ve} \quad \log(0) = 0 \text{ kabul edilerek} \quad (2.8)$$

Eşik Entropisi

$$E(s) = \sum_i E(s_i) \quad \varepsilon \text{ pozitif bir eşik değeri olup} \quad (2.9)$$

$$|s_i| > \varepsilon \Rightarrow E(s_i) = 1 \quad \text{ve} \quad |s_i| \leq \varepsilon \Rightarrow E(s_i) = 0$$

Sure Entropisi

$$|s_i| \leq \varepsilon \Rightarrow E(s) = \sum_i \min(s_i^2, \varepsilon^2) \text{ burada } \varepsilon \text{ pozitif bir eşik değeri olup} \quad (2.10)$$

2.3. Kullanılan Yöntem ve Aşamaları

Bu uygulama, temel olarak ön işlem, özellik çıkarım, sınıflama ve test aşamaları olmak üzere dört kısımdan meydana gelmiştir. Aşağıda bu aşamalar sırası ile verilmiştir:

2.3.1. Ön İşlem Aşaması

Bu çalışmada Entropi yapısı kullanılarak, tanınması istenilen RNA virüs çeşitleri: Picornavirüsler, Toguvirüsler, Flavivirüsler, Orthomyxovirüsler, Bunyavirüsler, Astrovirüsler, Retrovirüsler, Coronavirüsler, Reovirüsler, Rhabdovirüsler, Filovirüsler, Paramyxovirüsler' dir.

Ön işlem aşamasında yukarıda anlatılan görüntü işleme teknikleri kullanılmıştır. Bu aşamanın birinci kısmında yukarıda bahsedilen RNA virüs görüntüleri renkliden griye çevrilmiştir. Griye çevrilen görüntülerde eşikleme yapabilmek için, gri seviye dağılımlarını gösteren görüntü histogramı çıkarılmıştır. Görüntü histogramından yararlanılarak eşik değeri belirlenmiştir. Belirlenen eşik değerine göre bu değer üstündeki değerler için

çıktı resimdeki ilgili piksele 1 altındaki değerler içinde 0 değeri atanmıştır. Böylece resim arka plandan ayrılmıştır. Arka plandan ayırdığımız görüntüye Canny kenar çıkarma algoritması uygulanarak, bu görüntüdeki nesnelerin kenarları belirlenmiştir. Merkez-Kenar Değişimi yöntemi kullanılarak bu görüntülerin merkez piksellerinden, görüntüdeki nesnelerin kenarları üzerindeki her bir noktaya olan öklit uzaklıkları hesaplanarak, bu uzaklık değerlerinden bir uzaklık vektörü elde edilmiştir.

Aşağıda bahsedilen bu işlemler kısaca özetlenmiştir:

2.3.1.1. Görüntünün Renkliden Griye Çevrilmesi

Bu katmanda görüntüde eşikleme yapabilmek için görüntü renkliden griye çevrilmiştir. Aşağıda Şekil 2.5. (a)'da bir RNA virüsünün renkli görüntüsü, Şekil 2.5. (b)'de ise bu görüntünün griye çevrilmiş sonucu görülmektedir.



Şekil 2.5. RNA virüs resmi a) Astrovirus türü RNA virüsü b) Virüs görüntüsünün griye çevrilmiş hali

2.3.1.2. Görüntünün Eşikleme Yöntemi Kullanılarak Bölütlenmesi

Bu katmanda, görüntünün eşiklenmesi yöntemi kullanılarak bölütleme işlemi yapılmıştır. Bu amaç için görüntünün gri seviye histogramının orta noktası eşik değeri olarak atanır. Bu yöntem ikili görüntülerde iyi sonuç vermektedir. Aşağıda gri düzeydeki RNA virüsü görüntülerinin, bu işlem gerçekleştirilmeden ve bu işlem sonundaki durumları sırasıyla verilmiştir. Şekil 2.6. (a) ve (b)'de verilmiştir.



Şekil 2.6. RNA virüs resmi a) Astrovirus türü RNA virüsü b) Bölütlenmiş görüntü

2.3.1.3. Morfolojik ve Mantıksal İşlemler

Yapısal işlemlerdir. Bazen bir görüntüdeki nesne ile arka planı birbirinden ayırt etmek için uygun eşik değerini belirlemek yeterli olmayabilir. Bu durumda ek işlemler yapmamız gerekir. Görüntü işlemede nesnenin sınırlarını belirleyerek nesneyi arka plandan ayırabiliriz. İşte bu teknikler, bazı morfolojik ve mantıksal operatörler kullanılarak gerçekleştirilir. Bu çalışmada, morfolojik ve mantıksal operatörler olarak merkez kenar değişimi kenar bulma algoritması kullanılarak özellik vektörü elde edilmiş ve canny kenar bulma yöntemi ile nesnelerin sınırları belirlenmiştir. Bu morfolojik ve mantıksal işlemler yapılmadan ve yapıldıktan sonraki örnek birer RNA virüs resmi, aşağıda sırasıyla Şekil 2.7. (a) ve (b)'de verilmiştir.

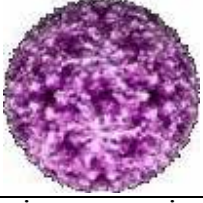
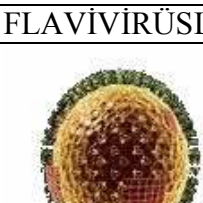


Şekil 2.7. RNA virüs resmi a) Astrovirus türü RNA virüsü
b) Canny kenar bulma işleminden sonraki görüntü

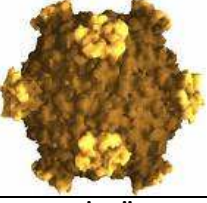
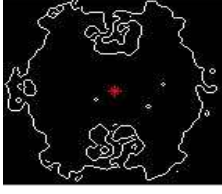
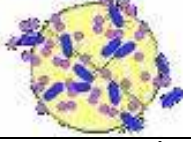
2.3.2. Özellik Çıkarım Aşaması

Bu aşamada ise yukarıda bahsedilen ön işlemten geçirilmiş RNA virüs görüntülerinin her biri 0° ile 165° arasında 15° derecelik açılarla döndürülerek, aynı RNA virüsünün birbirinden farklı 12 adet görüntüsü elde edilmiştir.

Tablo 2.4. RNA virüs görüntülerinin hesaplanan entropi değerleri

ÖNİŞLEMDEN ÖNCE	ÖNİŞLEMDEN SONRA	E1	E2	E3
		562.0537	1.5994e+004	1595
PİCORNAVİRÜSLER				
		716.2803	1.8434e+004	1724
TOGUVİRÜSLER				
		456.9634	1.2865e+004	1275
FLAVİVİRÜSLER				
		478.1944	1.3618e+004	1340
ORTHAMYXOVİRÜSLER				
		167.6647	4.3627e+003	428
BUNYAVİRÜSLER				

Tablo 2.4. RNA virüs görüntülerinin hesaplanan entropi değerlerinin devamı

ÖNİŞLEMDEN ÖNCE	ÖNİŞLEMDEN SONRA	E1	E2	E3
		50.7292	1.8884e+003	218
ASTROVİRÜSLER				
		127.1062	3.2710e+003	304
RETROVİRÜSLER				
		119.3178	4.0817e+003	439
CORONAVİRÜSLER				
		509.4558	1.0894e+004	960
REOVİRÜSLER				
		183.3079	6.4755e+003	723
RHABDOVİRÜSLER				
		197.6062	6.0050e+003	623
FİLOVİRÜSLER				

Tablo 2.4. RNA virüs görüntülerinin hesaplanan entropi değerlerinin devamı

ÖNİŞLEMDEN ÖNCE	ÖNİŞLEMDEN SONRA	E1	E2	E3
		356.0057	1.0655e+004	1101
PARAMYXOVİRÜSLER				

Ayrıca bu elde edilen her bir görüntü 10 farklı büyüklükte ölçeklendirilmiştir. Buna göre her bir farklı RNA virüs türü görüntüsü için farklı açı ve büyüklükte 12x10=120 adet resim elde edilmiştir. Daha sonra bu elde edilen görüntülerin her biri için sırasıyla norm, logaritmik enerji ve eşik entropileri olmak üzere 3 adet entropi değeri hesaplanmıştır. Böylece özellik vektörü elde edilmiş olup bu özellik vektörü, sınıflandırma aşamasında YSA ve UATBÇS sınıflandırıcıları ile BCO kümeleyicisinin girişlerine verilmiştir. Tablo 2.4 'de örnek olarak 120°'lik döndürmeyle elde edilmiş bazı RNA virüs görüntülerinin hesaplanan entropi değerleri gösterilmiştir.

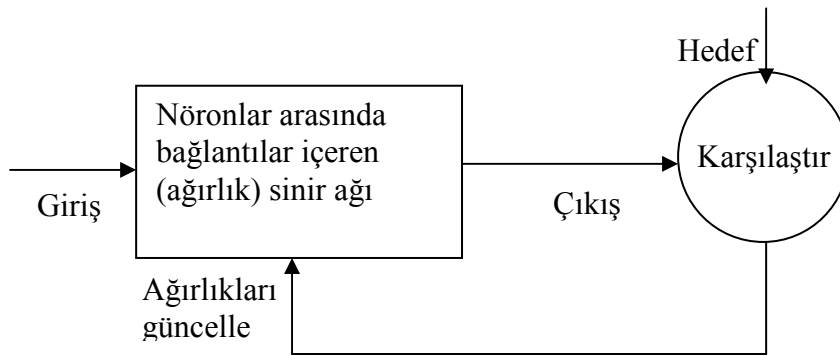
3. AKILLI SINIFLANDIRMA VE KÜMELEME TEKNİKLERİ

Tezin bu bölümünde Bölüm 2’ de anlatılan, görüntü işleme teknikleriyle yapılan önışlem ve özellik çıkarım aşamalarından sonra resimlere uygulanan akıllı sınıflandırma ve kümeleme teknikleri tanıtılacaktır.

3.1. Yapay Sinir Ağları

Yapay Sinir Ağları (YSA) insan beyninin özelliklerinden olan öğrenme yoluyla yeni bilgiler türetebilme, oluşturabilme ve keşfedebilme gibi yetenekleri herhangi bir yardım almadan otomatik olarak gerçekleştirebilen bilgisayar sistemleridir. YSA insan beyninden esinlenerek öğrenme sürecinin matematiksel olarak modellenmesi uğraşı sonucunda ortaya çıkmıştır. Bu sistemlerin başlıca belirgin özellikleri algoritmasız, tamamıyla paralel, uyarlanabilen, öğrenebilen ve paralel dağıtılmış bir hafızaya sahip olmalarıdır [21].

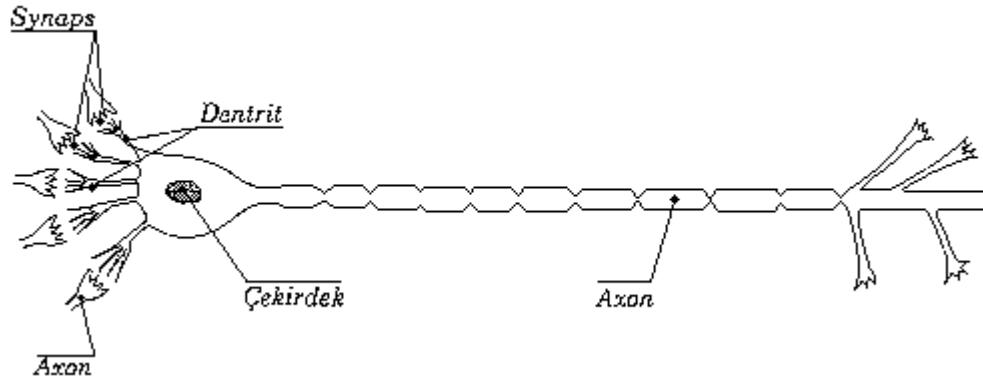
Sinir ağları genellikle eğitilerek belirli bir girişe karşılık verilen, istenilen çıkışı elde ederler. Bu durum aşağıdaki Şekil 3.1. de verilmiştir. Ağın çıkışı arzu edilen hedefe ulaşınca kadar çıkış ile hedef karşılaştırılarak ağın eğitimi (ağırlık değerlerinin ayarlanması) gerçekleştirilir. En basit tanımıyla yapay sinir ağları, bilinen en iyi eğri uydurma tekniklerinden birisidir [22].



Şekil 3.1. Yapay sinir ağlarının yapısı

3.1.1. Biyolojik Sinir Ağları ile Yapay Sinir Ağlarının Karşılaştırılması

Biyolojik sinir ağları başlangıçta sadece nörofizyolojistler tarafından araştırılmıştır. Araştırmaların konusu, genel olarak her bir nöronun yapısı ve birbirleri ile nasıl ilişkide bulundukları üzerine olmuştur. Yapay sinir ağlarının oluşturulmasında bu çalışmaların faydası inkar edilemez. Biyolojik sinir ağının en temel elemanı olan sinir hücresi, sinir sistemi içerisindeki fonksiyon ve görevlerine göre değişik şekil ve büyüklükte olabilir (Şekil 3.2). Sinir sisteminin en basit yapısı olan bir nöronun analizi bu nedenle önemlidir. Nöronun bir ucunda “dentrit” adı verilen ve hücreye, diğer hücrelerden veya dış dünyadan gelen bilgileri toplayan bağlantı elemanı, diğer ucunda ise tek bir life benzer “akson” adı verilen ve hücreden diğer hücrelere ve dış dünyaya bilgi taşıyan bağlantı elemanı vardır. Akson diğer hücrelerle birleşme esnasında dağınık dallara ayrılmaktadır. Bu iki uçtaki bağlantı noktalarının, elektrofizyolojik olarak hücrelerdeki bilgileri işlemede önemli yeri vardır. Hücrelerin birbiri ile elektrik işaretleri vasıtasıyla irtibat kurduğu belirlenmiştir. İşaretler, bir hücrenin aksonundan, diğerinin dentritine gönderilir. Bir akson birden fazla dentrit ile bağlantı kurabilir. Bu bağlantıların yapıldığı yere “synaps” denir.



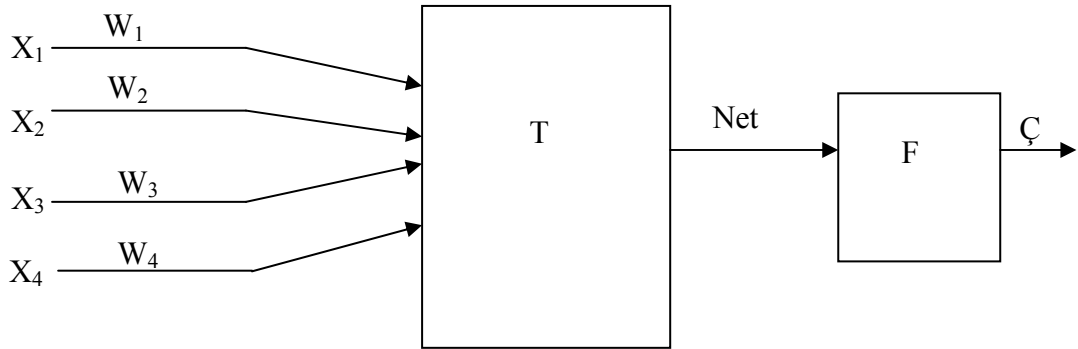
Şekil 3.2. Biyolojik nöronun şematik yapısı

Hücreler elektrik sinyalinin hücre duvarındaki voltajı değiştirerek üretirler. Bu ise hücrenin içinde ve dışında dağılmış bulunan iyonlar vasıtasıyla olur. Bu iyonlar sodyum, potasyum, kalsiyum ve klor gibi iyonlardır. Potasyum yoğunluğu nöronun içinde sodyum yoğunluğu nöronun dışındadır.

Bir hücre, diğer hücreye elektrik enerjisini bu kimyasal iyonlar vasıtasıyla transfer eder. Bazı iyonlar elektrik ve magnetik kutuplaşmaya sebep olurken, bazıları kutuplaşmadan kurtulup hücre zarını açarak iyonların hücreye geçmesine olanak sağlar. Zaten işaretlerin

bir hücreden diğerine akmasını sağlayan da bu kutuplaşmanın azalması olayıdır. İşaretler, hücrenin etkinliğini belirler. Bir hücrenin etkinliği, hücreye gelen synaps sayısı, synapslardaki iyonların konsantrasyonu ve bir de synapsın sahip olduğu güç olmak üzere üç faktöre bağlıdır. Bir hücre sahip olduğu dürtü miktarınca diğer hücreleri etkiler. Bazı hücreler diğerlerinin dürtülerini pozitif yönde, bazı hücreler de negatif yönde etkiler. İnsan sinir ağı sistemi, bu şekilde çalışan milyonlarca hücrenin bir araya gelmesinden oluşur. Beyinde korteks denilen bölgede her nöronun bir karşılığı vardır. Bir nöronun çıkışı ona bağlı olan bütün nöronlara iletilir. Fakat korteks, işin yapılabilmesi için hangi nöron harekete geçecekse sadece ona komut gönderir [23, 25].

Biyolojik beynin en önemli özelliklerinden birisi de öğrenme olayıdır. İnsanlar ve hayvanlar sürekli olarak içlerinde bulunduğu çevre ile ilişkiler sonucunda bir öğrenme süreci içerisindeyler. Öğrenilen her yeni bilgi, hemen beynin fonksiyonlarını etkileyerek davranışlarda da kendini gösterir. YSA'nın geliştirilmesinde bu özellik temel teşkil eder. Bir takım özellikler ise nörofizyolojik yaklaşımlar yerine mühendislik yaklaşımı ile geliştirilmektedir. Bir yapay sinir ağının yapısını belirleyen bazı faktörler vardır. Yapay sinir hücreleri veya mühendislik tabiriyle işlem elemanları, sinir ağının yapısal modelleri, ağın sahip olduğu kural ve stratejisi bunların başında gelir [25].



X_i : Giriş

W_i : Ağırlık

T: Toplam Fonksiyonu

F: Eşik Fonksiyonu

Ç: Çıkış

Şekil 3.3. Bir yapay sinir ağı hücresi modeli

Şekil 3.3. de verilmiş olan bir yapay sinir ağı hücresinin matematiksel ifadesi şu şekildedir;

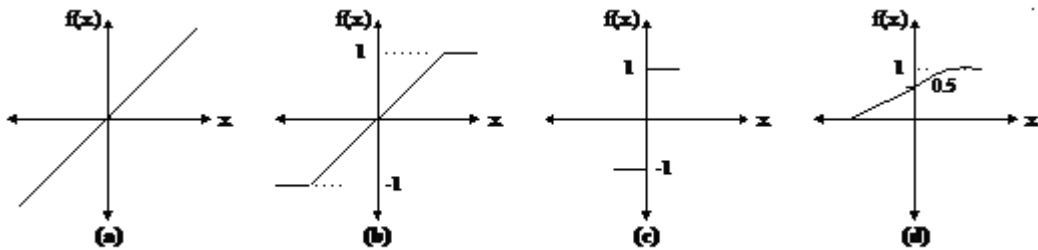
$$\text{Net} = \sum_{i=1}^4 w_i \cdot x_i = w_1 \cdot x_1 + w_2 \cdot x_2 + w_3 \cdot x_3 + w_4 \cdot x_4 \quad (3.1)$$

$$\text{Ç} = F(\text{Net})$$

Tıpkı biyolojik sinir hücresi gibi, işlem elemanına birden fazla giriş gelmekte ve sadece bir çıkış gitmektedir. Girişler, dentritlere benzer şekilde diğer yapay hücrelerden bağlantılar vasıtasıyla işlem elemanına bilgi gelmesini sağlarlar. Bazı durumlarda bir işlem elemanı kendisine de bilgi geri gönderebilir (geri besleme). Bahsedilen bu bilgiler elemanlar arasında bulunan bağlantı hatları üzerinde depolanır. Her bağlantının bir ağırlığı vardır. Bu ağırlık bir işlem elemanının diğer üzerindeki etkisini gösterir. Ağırlık büyüdükçe etki de büyür. Ağırlığın sıfır olması hiç bir etkinin olmaması, negatif olması ise etkinin ters yönde olması demektir. Bu ağırlıklar sabit olabildikleri gibi değişken de olabilirler.

YSA bilgi işleme sistemlerinin yeni neslini temsil eder. Genellikle YSA; model seçimi ve sınıflandırılması, işlev tahmini, en uygun değeri bulma ve veri sınıflandırılması gibi işlerde başarılıdır. Geleneksel bilgisayarlar ise özellikle model seçme işinde verimsizdir ve sadece algoritmaya dayalı hesaplama işlemleri ile kesin aritmetik işlemlerde hızlıdır [26].

Eşik fonksiyonları, işlem elemanlarının sınırsız sayıdaki girişini önceden belirlenmiş sınırdaki çıkış olarak düzenler. En çok kullanılan dört tane eşik (aktivasyon) fonksiyonu vardır. Bunlar, lineer, rampa, basamak ve sigmoid fonksiyonudur. Şekil 3.4 de bu fonksiyonlar gösterilmiştir.



Şekil 3.4 Yapay sinir ağları için kullanılan eşik fonksiyonları

Şekil 3.4. (a)'da gösterilen lineer fonksiyonun denklemi aşağıdaki gibidir.

$$f(x) = a.x \quad (3.2)$$

a: işlem elemanının x aktivitesini ayarlayan reel değerli bir sabittir. Lineer fonksiyon $[-\tau, +\tau]$ sınırları arasında kısıtlandığında (b)'deki rampa eşik fonksiyonu olur ve denklemi;

$$f(x) = \begin{cases} +\tau: \text{eğer } x \geq t \text{ ise} \\ x: \text{eğer } |x| < t \\ -t: \text{eğer } x \leq -t \text{ ise} \end{cases} \text{ şeklini alır.} \quad (3.3)$$

$+\tau$ ($-\tau$) işlem elemanının maksimumu (minimumu) çoğu zaman doyma seviyesi olarak adlandırılan çıkış değeridir. Eğer eşik fonksiyonu bir giriş işaretiyle bağlı ise yaydığı $+\tau$ giriş toplamı pozitif, bağlı değilse eşik basamak fonksiyonu $[-\delta]$ olarak adlandırılır. Şekil 3.4 (c), basamak eşik fonksiyonunu gösterir ve denklemi;

$$f(x) = \begin{cases} +\tau: \text{eğer } x > 0 \text{ ise} \\ -\delta: \text{diğer durumlar} \end{cases} \text{ şeklindedir.} \quad (3.4)$$

Son ve en önemli eşik fonksiyonu (bu çalışmada kullanılan) sigmoid fonksiyonudur. Şekil 3.4. (d) de gösterilen S biçimindeki sigmoid fonksiyonu; seviyeli, lineer olmayan çıkış veren, sınırlı, monoton artan fonksiyondur. Denklemi;

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

Toplama fonksiyonu, bir işlem elemanına gelen net girişi hesaplayan bir fonksiyondur. Net giriş genellikle gelen bilgilerin, ilgili bağlantıların ağırlıkları ile çarpılarak toplanması sonucunda belirlenir. Bu nedenle adı, toplama fonksiyonu olarak verilmiştir. Eşik fonksiyonu da, toplama fonksiyonu tarafından belirlenen net girişi alarak, işlem elemanının çıkışını belirleyen fonksiyondur. Genel olarak türevi alınabilen bir fonksiyon olması tercih edilir (Şekil 3.3.).

Toplama ve çıkış fonksiyonları, ilgili probleme bağlı olarak farklı şekiller alabilirler. İşlem elemanının çıkış ünitesi ise çıkış fonksiyonunun ürettiği dürtüyü diğer işlem elemanlarına veya dış dünyaya aktarma işlevini yapar. İşlem elemanları ağıın topolojik yapısına bağlı olarak tamamen birbirinden bağımsız ve paralel olarak çalışabilirler.

3.1.2. Yapay Sinir Ağlarının Yapıları ve İşlem Elemanları

Yapay sinir ağları temelde basit yapıda ve yönlü bir ağ biçimindedir. Yapay sinir ağlarının görevlerini yapabilmeleri için sahip oldukları fiziksel yapının da önemi vardır. Bugün 50 'ye yakın farklı model görülmekte ve bu sayı her geçen gün artmaktadır [25, 27, 28, 29]. Farklı modellerin oluşmasının sebebi, işlem elemanlarının birbirleri ile olan bağlantılarından ve uygulanan öğrenme kuralından kaynaklanmaktadır. İşlem elemanları ya tamamen birbirleri ile bağlantılı ya da yerel olarak gruplar halinde bağlantılı olabildikleri gibi değişik şekilde de birbirleri ile bağlanabilmektedirler. Bilgi akışı bu bağlantılar üzerinden tek yönlü olduğu gibi, çift yönlü de olabilir. Bir grup işlem elemanı bir araya gelerek bir katman oluştururlar. Genel itibariyle yapay sinir ağlarında 3 çeşit katman bulunur. Giriş katmanı sinir ağının dış dünya ile bağlantısını kurar. Ara katmanlar giriş katmanından gelen bilgileri işleme yeteneğine sahiptir. Çıkış katmanı ise sinir ağının kararlarını dış dünyaya aktarır. İşlem elemanları, katmanların birbirleri ile ilişkileri ve katmanlar arası ilişkiler değişik yapısal modellerin oluşmasına neden olmaktadır.

3.1.3. Yapay Sinir Ağlarında Öğrenme ve Öğrenme Algoritmaları

Bilginin kurallar şeklinde açıklandığı klasik uzman sistemlerin aksine, YSA gösterilen örnekten öğrenerek kendi kurallarını oluşturur. Yapay sinir ağının sahip olduğu bilgi, işlem elemanları arasındaki bağlantı hatları üzerinde saklanır ve ağırlıklar aracılığıyla gösterilir. Ağ, olaylar hakkında girişler ve çıkışlar arasındaki ilişkiyi, eldeki mevcut örneklerden genellemeler yaparak öğrenir ve bu genelleme ile yeni oluşan ve ortaya çıkan daha önce hiç görülmemiş olaylar hakkında karar verir. Yani ağa, bir örnek olay gösterildiğinde, giriş katmanından alınıp ara katmanlarda işlenerek, ağın o olay hakkında ürettiği sonuç, çıkış katmanına sunulur. Bu bilgiyi işleme, ağın sahip olduğu tecrübeye göre, bilginin ara katmanlarda ilişkilendirilmesi ile gerçekleştirilir. Bu ilişkilendirme modelden modele değişmektedir. Örneğin, ara katmanlardaki işlem elemanları sahip oldukları bağlantılar ile kendi kararlarını üretir ve çıkış katmanındaki işlem elemanlarına gönderirler. Çıkış katmanındaki işlem elemanları da, yine ilgili ağırlıkları kullanarak ağın en son kararını oluştururlar. Bu ağırlıklar, ilgili olayın belirli özelliklerini hafızada saklayan elemanlar gibi düşünülebilirler. Bilgi işleme ise (örüntü tanıma gibi), bir olay gösterildiğinde hafızadan

ilgili özellikleri çağırarak ve bunlar ile ilgili girişleri birlikte analiz ederek karar vermek şeklinde yorumlanabilir [24].

Yapay sinir ağlarında bilgi, ağdaki bağlantıların ağırlıklarında depolanır. Bir ağda öğrenme özetle, istenen bir işlevi yerine getirecek şekilde ağırlıkların ayarlanması sürecidir. Yapay sinir ağlarında öğrenme, sinirler arasındaki ağırlıkların değiştirilmesi ile gerçekleşmektedir [30]. Yapay sinir ağının öğrenme işlemini gerçekleştirebilmesi için, sahip olduğu bütün ağırlıkların, ilgili problemde öğrenilmesi istenen özellikleri genelleştirecek şekilde doğru değerlere sahip olması gerekir. Bu doğruluk ne kadar artarsa ağın sınıflandırma kabiliyeti dolayısı ile öğrenme işlemi o kadar iyi olur. Doğru ağırlık değerleri, bir öğrenme kuralına göre tespit edilirler. Çoğunlukla bağlantılara başlangıç değerleri olarak rastgele ağırlıklar atanır ve bu ağırlıklar eldeki örnekler incelendikçe bir kurala göre değiştirilerek doğru ağırlık değerleri bulunmaya çalışılır. Öğrenme yöntemleri temelde üç grupta toplanır [24, 25, 27].

- ❖ **Eğitici Öğrenme:** Dışarıdan bir eğitici sinir ağına müdahalesi söz konusudur. Eğitim işlemi, sinir ağına giriş ve çıkış bilgileri sunmaktan oluşur [30]. Eğitici, sinir ağının ilgili giriş için üretmesi gereken sonucu sinir ağı sistemine verir. Yani yapay sinir ağına giriş / çıkış ikilisinden oluşan örnekler sunulur. Bu ikili, ağın öğrenmesi gereken özellikleri temsil eder ve genellikle eğitim kümesi olarak tanımlanır. Ağ giriş kısmını alır ve o anki bağlantı ağırlıklarının temsil ettiği bilgi ile bir çıkış oluşturur. Bu çıkış, olması gereken çıkış ile karşılaştırılır ve aradaki hata tekrar ağa aktararak ağırlıklar bu hatayı azaltacak şekilde değiştirilirler.
- ❖ **Takviyeli Öğrenme:** Bu yöntemde de yine eğiticiye ihtiyaç vardır. Bu yöntemin eğitici öğrenme yönteminden farkı ise, eğitici sinir ağının üretmesi gereken sonuç yerine, onun ürettiği sonucun sadece doğru veya yanlış olduğunu söylemesidir. Bu işlem ağa bir takviye işaretinin gönderilmesi ile gerçekleştirilir. Bu yöntem, ilgili örnek için beklenen çıkışın oluşturulamadığı durumlarda çok faydalıdır.
- ❖ **Eğitici Özgün Öğrenme:** Bu yöntemde hiç bir eğiticiye ihtiyaç yoktur. Bundan dolayı çoğu zaman bu yöntem kendi kendine organize olmada denilmektedir. Ağ, kendine gösterilen örnekleri alır ve belli bir şarta göre sınıflandırır. Bu şart önceden bilinmeyebilir. Ağ, kendi öğrenme şartlarını kendisi oluşturmaktadır.

Öğrenme işleminin, başlangıçtaki rastgele seçilmiş ağırlıkların, belirli bir kurala göre değiştirilmesine dayandığı belirtilmişti. Bu öğrenme kurallarının önemli olanlarından bazıları aşağıda yer almaktadır. Bu algoritmalarda, $w_i = [w_{i1} \ w_{i2} \ ... \ w_{in}]$ vektörü ağdaki i .

hücreye gelen n tane girişin ağırlık katsayılarını, $x = [x_1 \ x_2 \ \dots \ x_n]$ vektörü ağırlık katsayıları w_i ile ağırlıklandırılarak y_i olarak üretilir. y_i ve d_i 'de sırasıyla i. hücrenin hesaplanan ve istenen çıkışı temsil etmektedirler.

➤ **Hebb Öğrenme Algoritması:**

Hebb öğrenme algoritmasında temel fikir, hücrenin çıkış üretmesine sebep olan bağlantıların ağırlık katsayılarını büyüterek etkilerini arttırmaktır. Buna göre bu değişim aşağıda Denklem 3.6. ve Denklem 3.7.deki gibi tanımlanmıştır.

$$w_i(t+1) = w_i(t) + \varepsilon \cdot f(w_i(t) \cdot x(t)) \cdot x(t) \quad (3.6)$$

veya

$$w_i(t+1) = w_i(t) + \varepsilon \cdot y_i(t) \cdot x(t) \quad (3.7)$$

Burada ε 'a öğrenme katsayısı veya öğrenme oranı denilmektedir. Dışarıdan istenilen bir çıkış uygulanmadığından, eğitici bir öğrenme algoritmasıdır.

➤ **Yarışmacı Öğrenme Algoritması:**

Bu algoritmada esas, giriş işaretine en yakın olan bağlantı ağırlık katsayılarını bulmaktır. Bu, w 'lara ilişkin nöron kazanan nöron olarak adlandırılmaktadır. Sadece kazanan nöron bir çıkış üretmekte ve bu nörona ilişkin w 'lar değişime uğramaktadır. Eğitici bir öğrenmenin tipik bir örneği olan bu algoritma sonuçta ağırlık katsayıları sınıflandırmaktadır.

$$w(t+1) = w(t) + \varepsilon \cdot (x(t) - w_m(t)) \quad (3.15)$$

Burada w_m kazanan hücrenin bağlantı ağırlık katsayısı olup aşağıdaki gibi tanımlanmıştır.

$$w_m \cdot x = \max(w_i \cdot x) \quad (3.16)$$

ve $i = 1, 2, \dots, p$ dir.

➤ **Algılayıcı (Perceptron) Öğrenme Algoritması:**

Rosenblatt tarafından önerilen algılayıcı öğrenme algoritmasında w değişimi, nöron çıkışı ve istenilen çıkışın farkı ile orantılıdır. Dolayısıyla eğitici bir öğrenme algoritmasıdır.

$$w_i(t+1) = w_i(t) + \varepsilon \cdot [d_i(t) - \text{sgn}(w_i(t) \cdot x(t))] \cdot x(t) \quad (3.8)$$

Ağın ürettiği çıkış istenilen çıkıştan farklı ise, hata azaltılacak yönde hareket edilmektedir. Hata sıfır olduğunda w değişimi durmaktadır.

➤ **Delta Kuralı (Eğim düşme) Öğrenme Algoritması:**

Eğim düşme (gradient descent) öğrenme algoritması veya “delta kuralı” öğrenme algoritması, sadece türetilabilir fonksiyonlara sahip olan ağlara uygulanabilmektedir. Eğitici bir öğrenme algoritmasıdır. w değişimi, hatanın eğiminin tersi yöndedir. Böylece, hata fonksiyonunun minimum olduğu yere doğru hareket edilmektedir.

$$w_i(t+1) = w_i(t) - \varepsilon \cdot \frac{\partial E(t)}{\partial w(t)} \quad (3.9)$$

Burada $E(t)$, hata fonksiyonunu temsil etmektedir ve aşağıdaki gibi tanımlanmıştır.

$$E(t) = \frac{1}{2} (d_i(t) - y_i(t))^2 \quad (3.10)$$

$$E(t) = \frac{1}{2} [d_i(t) - f(w_i(t).x(t))]^2 \quad (3.11)$$

$$\frac{\partial E(t)}{\partial w(t)} = -[d_i(t) - f(w_i(t).x(t))] \frac{\partial f(w_i(t).x(t))}{\partial (w_i(t).x(t))} . x(t) \quad (3.12)$$

Bilinen hata geriye yayılması algoritmasının esası da, bu öğrenme kuralına dayanmaktadır.

➤ **Widrow-Hoff Öğrenme Algoritması:**

Widrow-Hoff öğrenme algoritması, eğitici bir öğrenme algoritmasıdır. w değişimi ağırlık çıkışından bağımsızdır.

$$w_i(t+1) = w_i(t) + \varepsilon (d_i(t) - w_i(t).x(t)).x(t) \quad (3.13)$$

➤ **İltili Öğrenme Algoritması:**

Hebb öğrenme algoritmasının eğitici uyarlamasıdır. $f(w_i(t).x_i(t))$ yerine $d_i(t)$ istenen çıkış alındığında aşağıdaki denklem elde edilir.

$$w_i(t+1) = w_i(t) + \varepsilon . d_i(t).x(t) \quad (3.14)$$

➤ **Outstar Öğrenme Algoritması:**

Eğitici öğrenme algoritma türü olan bu öğrenme algoritmasında amaç, w 'ları istenen çıkışa benzetmektir.

$$w(t+1) = w(t) + \varepsilon . (d_i(t) - w(t)) \quad (3.17)$$

3.2. Uyarlamalı Ağ Tabanlı Bulanık Çıkarım Sistemi (UATBÇS)

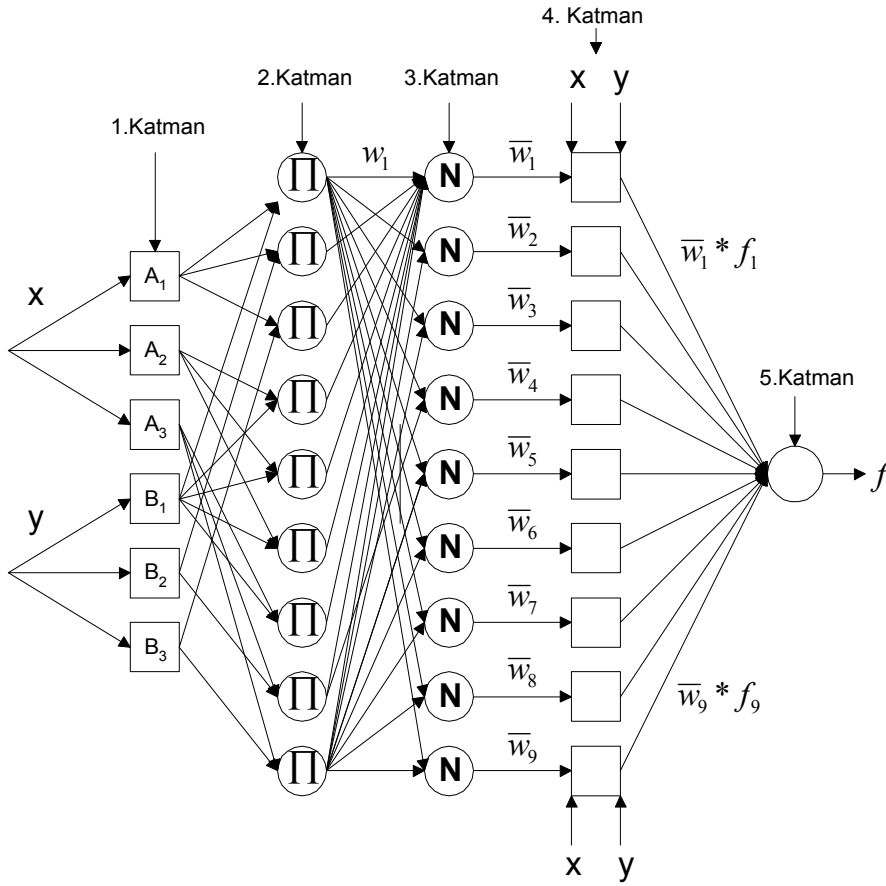
Uyarlamalı ağ tabanlı bulanık çıkarım sistemi, akıllı sistemlerin geliştirilmesinde kullanılan tamamlayıcı araçlardan biridir. Esasen UATBÇS yapısı, Sugeno tipi bulanık sistemlerin, sinirsel öğrenme kabiliyetine sahip bir ağ yapısı olarak temsilinden ibarettir. Bu ağ, her biri belli bir fonksiyonu gerçekleştirmek üzere, katmanlar halinde yerleştirilmiş düğümlerin birleşiminden oluşmuştur [32].

Yapay sinir ağları, ham verilerle uğraşıldığında iyi sonuçlar veren düşük seviyeli yapılardır. Bulanık mantık ise, uzman görüşü sonucu elde edilen dilsel bilgileri kullanarak daha yüksek seviyeli sonuçlar çıkarmaktadır. Aslında bulanık sistemlerin öğrenme kabiliyeti yoktur ve kendilerini yeni çevreye adapte edemezler. Diğer yandan yapay sinir ağları öğrenme kabiliyetine sahiptir; fakat kullanıcı tarafından anlaşılmazlar [31].

3.2.1. UATBÇS Mimarisi

Sinirsel bulanık sistemler, yapay sinir ağlarının paralel hesaplayabilme ve öğrenme kabiliyeti ile bulanık mantığın uzman bilgisini kullanarak sonuçlar çıkarabilme özelliklerinin birleşiminden oluşur. Böylece sinirsel bulanık sistemler sayesinde YSA daha anlaşılır bir hale gelir [33].

UATBÇS, ele alınan problem için oluşturulan yapıya göre olası tüm kuralları atayabilmekte veya kuralların veriler yardımıyla uzman tarafından atanmasına olanak vermektedir. UATBÇS'nin kural oluşturabilmesi veya kural oluşturulmasına olanak sağlaması uzman görüşlerinden faydalanması anlamına gelmektedir. UATBÇS 5 katmandan oluşmaktadır. Bu sistem Şekil 3.5'de gösterilmiştir. UATBÇS yapısındaki her katmana ait düğüm işlevleri ve katmanların işleyişi sırasıyla şöyledir [31].



Şekil 3.5. 2 girişli 9 kurallı bir UATBÇS sınıflandırıcı yapısı

Bu UATBÇS yapısı yukarıda da görüldüğü gibi 5 katmandan oluşmaktadır. Burada x ve y giriş, f ise çıkış olarak alınırsa temel kural yapısı şu şekilde yazılabilir:

$$\text{Eğer } x \text{ } A_1 \text{ ve } y \text{ } B_1 \text{ ise } f_1 = p_1 x + q_1 y + r_1 \quad (3.18)$$

Burada p, q ve r lineer çıkış parametreleridir. İki girişli ve bir çıkışlı bir UATBÇS' in temel yapısı Şekil 3.5 de görülmektedir. Bu yapı, 5 katman ve 9 adet eğer-ise kuralı kullanılarak oluşturulmuştur:

1.Katman: Bulanıklaştırma katmanı olarak adlandırılır. Giriş değerlerini bulanık kümelere ayırmada Jang'ın UATBÇS modeli, üyelik fonksiyonu şekli olarak geliştirilmiş Bell aktivasyon fonksiyonunu kullanmaktadır. Burada, her bir düğümün çıkışı, giriş değerlerine ve kullanılan üyelik fonksiyonuna bağlı olan üyelik derecelerinden oluşmaktadır ve 2. katmandan elde edilen üyelik dereceleri $\mu_{A_j}(x)$ ve $\mu_{B_j}(y)$ şeklinde gösterilir. Bu katmandaki hücre sayısı, iki giriş ve bu iki girişin her birine üç üyelik fonksiyonu tanımlandığına göre altı adettir ($i=6$ 'dır). Buna göre,

$$O_{1,i}=\mu_{Ai}(x), i=1,2,3 \text{ için } O_{1,i}=\mu_{Bi-3}(y), i=4,5,6 \text{ için} \quad (3.19)$$

Burada x ve y girişlerdir. Bu katmanın çıkışı kuralların varsayım (eğer) kısımlarının üyelik fonksiyonlarına olan üyelik dereceleridir. Kullanılan üyelik fonksiyonu Denklem 3.20 de verilmiştir. Buradan da görüldüğü gibi üyelik fonksiyonu olarak Gauss üyelik fonksiyonu kullanılmıştır.

$$\mu_{Ai}(x), \mu_{Bi-3}(y) = \exp(-(x_i - c_i)/(a_i))^2 \quad (3.20)$$

2. Katman: Kural katmanıdır. Bu katmandaki her bir düğüm, Sugeno bulanık mantık çıkarım sistemine göre oluşturulan kuralları ve sayısını ifade etmektedir. Her bir kural düğümünün çıkışı μ_i , 1. katmandan gelen üyelik derecelerinin çarpımı olmaktadır (Denklem 3.21).

$$O_{2,i}=w_i=\mu_{Ai}(x)*\mu_{Bi}(y), \quad i=1,2,3,\dots,9 \quad (3.21)$$

3. Katman: Normalizasyon katmanıdır. Bu katmandaki her bir düğüm, kural katmanından gelen tüm düğümleri giriş değeri olarak kabul etmekte ve her bir kuralın normalleştirilmiş ateşleme seviyesini hesaplamaktadır (Denklem 3.22).

$$O_{3,i}=\bar{w}_i=w_i/(w_1+w_2+\dots+w_9), \quad i=1,2,3,\dots,9 \quad (3.22)$$

4. Katman: Bu katmanda normalize edilmiş her bir kural kendine ait çıkış fonksiyonu ile çarpılır (Denklem 3.23).

$$O_{4,i}=\bar{w}_i * f_i=w_i*(p_i x + q_i y + r_i) \quad (3.23)$$

Buradaki p, q ve r lineer parametreleri sonuç parametreleri olarak adlandırılır.

5. Katman: Toplam katmanıdır. Bu katmanda sadece bir düğüm vardır ve Σ ile etiketlenmiştir. Burada, 5.katmandaki her bir düğümün çıkış değeri toplanarak sonuçta, UATBÇS sisteminin gerçek değeri elde edilir (Denklem 3.24).

$$O_{5,i}=\text{toplam çıkış} = \frac{\sum_i w_i f_i}{\sum_i w_i} \quad (3.24)$$

3.2.2. UATBÇS İçin Geri Yayımlı Öğrenme Algoritması

Geri yayımlı öğrenme algoritmasında ağın çıkışından elde edilen hata değeri giriş katmanına kadar geriye doğru yansıtılarak gerekli ağırlık değişkenlerinin ayarlanması gerçekleştirilir [34]. Amaç E hata ölçütünü öğrenme işlemi sonunda bütün giriş örnekleri için sıfıra götürmektir. Ağın çıkışında hata, istenen değer d ile gerçek çıkış f arasındaki farktır ve Denklem 3.25. deki şekilde ifade edilir,

$$e = d - f \quad (3.25)$$

Bu hata değerine en küçük kareler yöntemi uygulanarak hata ölçütü E Denklem 3.26. da verildiği gibi hesaplanır.

$$E = \frac{1}{2} e^2 \quad (3.26)$$

Hata ölçütü kullanılarak hata geriye doğru katman katman aşağıdaki gibi yansıtılır.

5. Katman: Bu katmanda herhangi bir ağırlık değeri ayarlaması yoktur, sadece çıkıştaki hatanın Denklem 3.27. deki gibi katman çıkışına yansıtılması hesaplanır,

$$\delta^5 = -\frac{\partial E}{\partial f} = -\left[\frac{\partial E}{\partial e} \frac{\partial e}{\partial f} \right] = e \quad (3.27)$$

4. Katman: Bu katmanda p_i , q_i ve r_i sonuç değişkenlerinin ayarlanması için gerekli hesaplamalar gerçekleştirilir.

$$\Delta p_i = -\frac{\partial E}{\partial p_i} = \left[-\frac{\partial E}{\partial f} \right] \left[\frac{\partial f}{\partial p_i} \right] \quad (3.28)$$

$$f = \sum_i \bar{\mu}_i (p_i x + q_i y + r_i) \quad i = 1, 2 \quad (3.29)$$

$$\frac{\partial f}{\partial p_i} = \bar{\mu}_i x \quad (3.30)$$

buradan,

$$\Delta p_i = \delta^5 \bar{\mu}_i x \quad (3.31)$$

Aynı işlemler diğer sonuç değişkenlerine de uygulanarak değişimler hesaplanır. Buna göre,

$$\Delta q_i = \delta^5 \bar{\mu}_i y \quad (3.32)$$

$$\Delta r_i = \delta^5 \bar{\mu}_i \quad (3.33)$$

olur.

Elde edilen bu ifadeler öğrenme oranı eklenerek üçgen (delta) kuralı uygulanır.

$n = \{p, q, r\}$ olarak alınırsa bu ifadeler tek bir eşitlik olarak aşağıdaki gibi gösterilir.

$$n_i(k+1) = n_i(k) + \eta_n \Delta n_i(k+1) \quad (3.34)$$

3. Katman: Bu katmanda çıkış katmanından yansıyan hata değeri hesaplanır herhangi bir ağırlık hesaplaması yoktur.

$$\delta_i^3 = -\frac{\partial E}{\partial \bar{\mu}_i} = \left[-\frac{\partial E}{\partial f} \right] \left[\frac{\partial f}{\partial \bar{\mu}_i} \right] \quad i = 1, 2 \quad (3.35)$$

$$f = \sum_i \bar{\mu}_i f_i \quad (3.36)$$

$$\frac{\partial f}{\partial \bar{\mu}_i} = f_i \quad (3.37)$$

$$\delta_i^3 = \delta^4 f_i \quad (3.38)$$

2. Katman: Bu katmanda çıkış katmanından yansıyan hata değeri hesaplanır herhangi bir ağırlık hesaplaması yoktur.

$$\delta_i^2 = -\frac{\partial E}{\partial \mu_i} = \left[-\frac{\partial E}{\partial f} \frac{\partial f}{\partial \bar{\mu}_i} \right] \left[\frac{\partial \bar{\mu}_i}{\partial \mu_i} \right] \quad i = 1, 2 \quad (3.39)$$

$$\bar{\mu}_i = \frac{\mu_i}{\sum_i \mu_i}, \text{ bu ifade de } a = \sum_i \mu_i \text{ olarak alınırsa,}$$

$$\frac{\partial \bar{\mu}_i}{\partial \mu_i} = \left[\frac{a - \mu_i}{a^2} \right] \quad (3.40)$$

olarak bulunur. Bu ifade Denklem 3.41. de yerine konularak aşağıdaki son durum elde edilir.

$$\delta_i^2 = \delta_i^3 \left[\frac{a - \mu_i}{a^2} \right] \quad (3.41)$$

1. Katman: Bu katmanda hem hata değeri hem de giriş üyelik işlevlerine ait değişkenlerin ayarlanması gerçekleştirilir.

$$\delta_i^1 = -\frac{\partial E}{\partial \mu_{Ai}} = -\left[\frac{\partial E}{\partial f} \frac{\partial f}{\partial \bar{\mu}_i} \frac{\partial \bar{\mu}_i}{\partial \mu_i} \right] \left[\frac{\partial \mu_i}{\partial \mu_{Ai}} \right] \quad i = 1, 2 \quad (3.42)$$

$$\frac{\partial \mu_i}{\partial \mu_{Ai}} = \mu_{Bi} \quad (3.43)$$

$$\delta_i^1 = \delta_i^2 \mu_{Bi} \quad (3.44)$$

$$\delta_{i+2}^1 = -\frac{\partial E}{\partial \mu_{Bi}} = -\left[\frac{\partial E}{\partial f} \frac{\partial f}{\partial \bar{\mu}_i} \frac{\partial \bar{\mu}_i}{\partial \mu_i} \right] \left[\frac{\partial \mu_i}{\partial \mu_{Bi}} \right] \quad (3.45)$$

$$\frac{\partial \mu_i}{\partial \mu_{Bi}} = \mu_{Ai} \quad (3.46)$$

$$\delta_{i+2}^1 = \delta_i^2 \mu_{Ai} \quad (3.47)$$

$$\mu_{Ai} = \exp \left[- \left(\frac{x - m_i}{\sigma_i} \right)^2 \right] \quad v_i = - \left[\frac{x - m_i}{\sigma_i} \right]^2 \quad \mu_{Ai} = \exp[v_i] \quad (3.48)$$

Çan eğrisi şekilli üyelik işlevlerinin orta noktalarındaki değişim aşağıdaki gibi hesaplanır.

$$\Delta m_i = - \frac{\partial E}{\partial m_i} = - \left[\frac{\partial E}{\partial f} \frac{\partial f}{\partial \bar{\mu}_i} \frac{\partial \bar{\mu}_i}{\partial \mu_i} \frac{\partial \mu_i}{\partial \mu_{Ai}} \right] \left[\frac{\partial \mu_{Ai}}{\partial v_i} \frac{\partial v_i}{\partial m_i} \right] \quad i = 1, 2 \quad (3.49)$$

$$\frac{\partial \mu_{Ai}}{\partial v_i} = \exp[v_i] = \mu_{Ai}$$

$$(3.50) \quad \frac{\partial v_i}{\partial m_i} = 2 \left[\frac{x - m_i}{\sigma_i^2} \right]$$

(3.51)

$$\Delta m_i = 2 \left[\frac{x - m_i}{\sigma_i^2} \right] \delta_i^1 \mu_{Ai} \quad \delta_i^1 = \delta_i^2 \mu_{Bi} \quad (3.52)$$

$$\Delta m_i = 2 \left[\frac{x - m_i}{\sigma_i^2} \right] \delta_i^2 \mu_{Ai} \mu_{Bi} \quad \mu_i = \mu_{Ai} \mu_{Bi} \quad (3.53)$$

$$\Delta m_i = 2 \left[\frac{x - m_i}{\sigma_i^2} \right] \delta_i^2 \mu_i \quad (3.54)$$

Aynı şekilde Δm_{i+2} aşağıdaki şekilde elde edilir,

$$\Delta m_{i+2} = 2 \left[\frac{y - m_{i+2}}{\sigma_{i+2}^2} \right] \delta_i^2 \mu_i \quad (3.55)$$

Çan eğrisi şekilli üyelik işlevlerinin standart sapmalarındaki değişim aşağıdaki gibi hesaplanır.

$$\Delta \sigma_i = - \frac{\partial E}{\partial \sigma_i} = - \left[\frac{\partial E}{\partial f} \frac{\partial f}{\partial \bar{\mu}_i} \frac{\partial \bar{\mu}_i}{\partial \mu_i} \frac{\partial \mu_i}{\partial \mu_{Ai}} \right] \left[\frac{\partial \mu_{Ai}}{\partial v_i} \frac{\partial v_i}{\partial \sigma_i} \right] \quad i = 1, 2 \quad (3.56)$$

$$\frac{\partial v_i}{\partial \sigma_i} = 2 \left[\frac{(x - m_i)^2}{\sigma_i^3} \right] \quad (3.57)$$

$$\Delta \sigma_i = 2 \left[\frac{(x - m_i)^2}{\sigma_i^3} \right] \delta_i^2 \mu_i \quad (3.58)$$

Aynı şekilde $\Delta \sigma_{i+2}$ aşağıdaki şekilde elde edilir.

$$\Delta \sigma_{i+2} = 2 \left[\frac{(y - m_{i+2})^2}{\sigma_{i+2}^3} \right] \delta_i^2 \mu_i \quad (3.59)$$

Elde edilen bu değişimlere üçgen kuralı uygulanarak üyelik işlevine ait değişkenlerin ayarlanması aşağıdaki şekilde gerçekleştirilir.

$$m_i(k+1) = m_i(k) + \eta_m \Delta m_i(k+1) \quad i = 1,2,3,4 \quad (3.60)$$

$$\sigma_i(k+1) = \sigma_i(k) + \eta_m \Delta \sigma_i(k+1) \quad i = 1,2,3,4 \quad (3.61)$$

Burada toplam 8 değişken ayarlanmaktadır.

3.3. Bulanık C-Ortalamlar Sınıflandırıcısı (BCO)

Bulanık C-Ortalamlar (BCO) algoritması, bulanık bölünmeli kümeleme tekniklerinden en iyi bilinen ve yaygın kullanılan yöntemdir. Bulanık C-Ortalamlar algoritması 1973 yılında Dunn tarafından ortaya atılmış ve 1981’ de Bezdek tarafından geliştirilmiştir [35]. Bulanık C-Ortalamlar algoritması, bir bulanık amaç fonksiyonunu optimize etmeyi amaçlar ve veri noktalarının kümelendirilmesi için kullanılır. Bu sınıflandırıcı aynı zamanda serbest model tahmin edici ve BCO kümelemenin genel kavramı üzerinde esaslanmıştır. Sınıflandırıcı eğitim verisi üzerinde üyelik fonksiyonlarını hesaplar ve kümeleme merkezinin yerini belirler. Yeni verinin sınıflandırılması öğrenilen üyelik fonksiyonları üzerine temellenmiştir [36].

3.3.1. Matematiksel Temel

Bulanık c-ortalamlar metodu, nesnelerin iki veya daha fazla kümeye ait olabilmesine izin verir. Bulanık mantık prensibi gereği her veri, kümelerin her birine $[0,1]$ arasında değişen birer üyelik değeri ile aittir. Bir verinin tüm sınıflara olan üyelik değerleri toplamı “1” olmalıdır. Nesne hangi küme merkezine yakın ise o kümeye ait olma üyeliği diğer kümelere ait olma üyeliğinden daha büyük olacaktır. Amaç fonksiyonun belirlenen minimum ilerleme değerine yakınsaklaşmasıyla kümeleme işlemi tamamlanır.

Her bir küme merkezi, v_i bir üyelik fonksiyonu veya U ayrılma (bölünme) matrisi ile karakterize edilir. U'nun bir elemanı olarak, i kategorisindeki k örüntüsünün üyelik derecesi μ_{ik} ifadesi olarak yazılır. Bu üyelik derecesi ifadesi aşağıda verilen Denklem 3.62 ile bulunur.

$$\mu_{ik} = \frac{\left(\frac{1}{\|x_k - v_i\|^2} \right)^{\frac{1}{m-1}}}{\sum_{j=1}^{N_C} \left(\frac{1}{\|x_k - v_j\|^2} \right)^{\frac{1}{m-1}}} \quad (3.62)$$

Burada, N_C kategorilerin sayısıdır, m ; $1 < m < \infty$ arasında bir bulanıklaştırma parametresidir ve x_k k. eğitim örneğidir. Çift çizgi sembolü bir uzaklık ölçüsüdür. Mevcut uygulamalarda, öklit uzaklığı olarak benimsenmektedir. x_k örneği v_i merkezine yaklaştıkça da üyeliği 1'e yaklaşır, uzağında ise üyeliği 0'a yaklaşır. Yukarıdaki denklemin payı i kümesine uygun olma (dahil olma) miktarı olarak yorumlanabilir ve payda ise kümelerin seçilen gruba uygun olma seçeneğinin toplam miktarı olarak yorumlanabilir. Bu nedenle, yukarıdaki denklem üyeliğin bir ortalamasıdır. i. küme merkezi şu şekilde verilir:

$$v_i = \frac{1}{\sum_{k=1}^n (\mu_{ik})^m} \sum_{k=1}^n (\mu_{ik})^m x_k \quad (3.63)$$

Burada, n eğitim örneklerinin sayısıdır. Küme merkezi eğitim örneklerinin bir ortalaması olarak düşünülebilir, üyeliklere uygun olarak ağırlıklanır. Aynı zamanda sorgulamadaki kategori ile uygun olarak $\sum_{k=1}^n (\mu_{ik})^m$ örneğinin bulanıklık değeri ile normalize edilir.

3.3.2. Eğitim

Eğitimde en küçük kareler yönteminin genellemesi olan aşağıdaki amaç fonksiyonunu öteleyerek minimize etmek için çalışır [35].

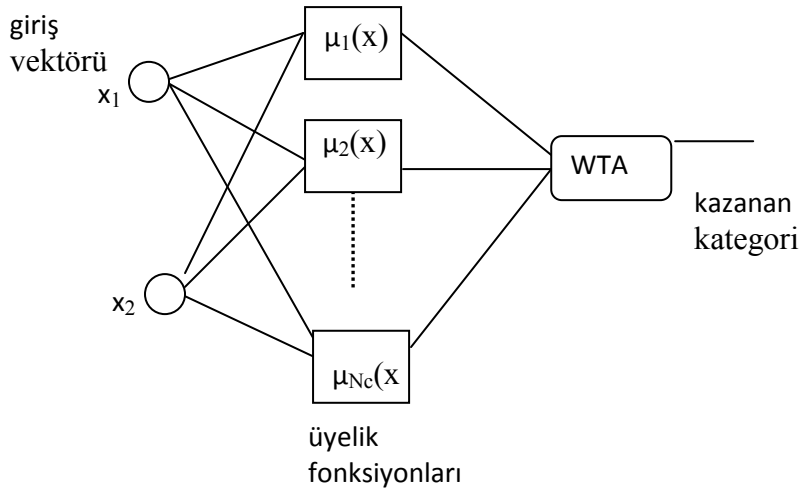
$$\sum_{i=1}^{N_C} \sum_{k=1}^n (\mu_{ik})^m \|x_k - v_i\|^2 \quad (3.64)$$

v_i küme merkezlerini, μ_{ik} ise üyelik fonksiyonlarını temsil etmektedir. Bu iki parametreden minimizasyon esnasında yararlanılabilir. Amaç fonksiyonunun eğimi, üyelik fonksiyonları ve küme merkezleri μ_{ik} ve v_i denklemleri ile tanımlanan sistem kazancıdır.

Bu sistem denklemleri, bulanık kümeleme algoritması olarak bilinen ardışık bir prosedür tarafından çözülebilir.

3.3.3. Test Etme

Yeni veri verildiğinde, çeşitli kümeler üyelikler hesaplanır. Bir kısmi matrisden ziyade bir kısmi vektör, sorguda yalnız bir örnek alınarak hesaplanır. En yüksek üyelik ile kümeleme Şekil 2.18. deki sınıflandırıcı kararı olarak seçilir.



Şekil 3.6. Bulanık C-Ortalama kümeleyicisi

Kümeleme, verilerin benzerlikleri ölçütünde alt kümeler veya sınıflara ayırma işlemi olarak tanımlanır. BCO algoritması iteratif bir yapıda olup, aşağıdaki denklemde verilen J değerinin minimize edilmesi için üyelik derecesi olan u_{ij} ve kümeleme merkezleri olan v_j lerin hesaplanmasına dayanır. Kümelerin başlangıç merkezleri rastgele verilebilir [37-40].

$$J = \sum_{i=1}^M \sum_{j=1}^R (u_{ij})^m |x_i - v_j|^2 \quad (3.65)$$

Burada m, 1 den büyük herhangi bir gerçekte sayıdır ve $1 \leq m < \infty$ dir [36, 37]. M giriş veri sayısını, R ise küme sayısını göstermektedir. u_{ij} , j kümesindeki x_i nin üyelik derecesidir. x_i ise d boyutlu verinin i.elemanıdır. v_j kümenin d boyutlu merkezidir. || ise giriş verisi ile merkez arasındaki benzerliği ifade eder. Bulanık kümeleme yukarıda verilen fonksiyonu, u_{ij} üyelik derecesinin ve v_j nin aşağıda belirtilen fonksiyonlarla güncellenerek

ve iterasyonlar yaparak en uygun şekle sokar [37]. Dikkat edilmesi gereken önemli nokta

$\sum_{j=1}^R u_{ij} = 1$ olması gereksinimidir.

$$u_{ij} = \left[\sum_{k=1}^R \left(\frac{|x_i - v_j|^2}{|x_i - v_k|^2} \right)^{\frac{1}{m-1}} \right]^{-1} \quad (3.66)$$

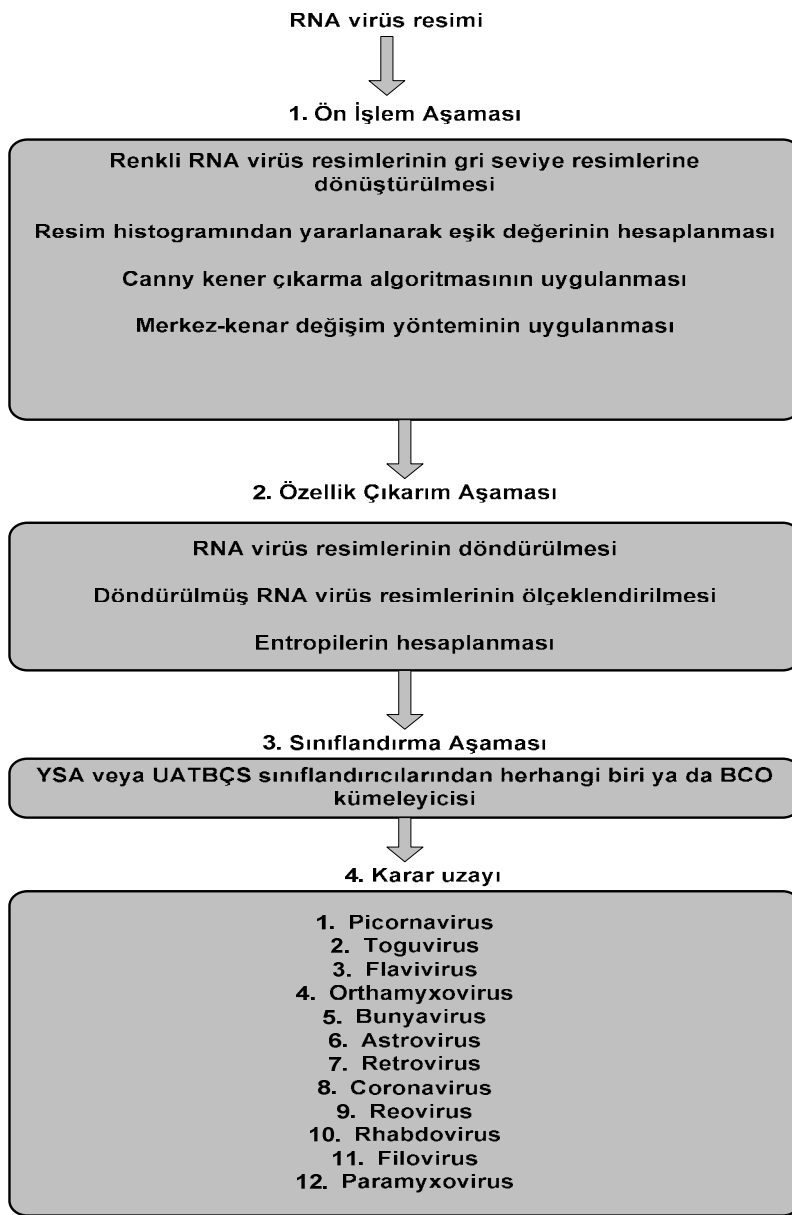
Elde edilen üyelik derecelerinden yeni küme merkezleri aşağıdaki denkleme göre yeniden hesaplanmaktadır.

$$v_j^{new} = \frac{\sum_{i=1}^M x_i (u_{ij})^m}{\sum_{i=1}^M (u_{ij})^m} \quad (3.67)$$

Elde edilen yeni küme merkezi ile bir önceki küme merkezi arasındaki fark istenen değerde ise döngü işlemi bitirilir.

4. RNA VİRÜS RESİMLERİNİN AKILLI SINIFLANDIRMA VE KÜMELEME UYGULAMALARI

Bu çalışmada kullanılan Çoklu Entropi-Yapay Sinir Ağları (ÇE-YSA), Çoklu Entropi-Uyarlamalı Ağ Tabanlı Bulanık Çıkarım Sistemi (ÇE-UATBÇS) ve Çoklu Entropi-Bulanık C-Ortalamalar (ÇE-BCO) algoritmalarının yapısı Şekil 4.1’de verilmiştir.



Şekil 4.1. Bu çalışmada kullanılan ÇE-YSA, ÇE-UATBÇS ve ÇE-BCO algoritmalarının yapısı

4.1. Doğru Sınıflandırma ve Kümeleme Başarımının Testi Aşaması

Bu aşama yukarıdaki ÇE-YSA, ÇE-UATBÇS ve ÇE-BCO algoritmalarının ayrı ayrı kullanılması ile gerçekleştirilen sınıflandırma ve kümeleme sonuçlarının doğruluğunu test etmektedir. Bu amaçla daha önceden belirtildiği gibi özellik çıkarım safhasında elde edilen 1440×3 ($12 \times 120 \times 3$) özellik vektörünün yarısı (720×3) sınıflandırma ve kümelemede eğitim aşamasında, diğer yarısı (720×3) ise eğitilmiş yapay sinir ağlarının, uyarlamalı ağ tabanlı bulanık çıkarım sisteminin ve bulanık c-ortalamaların doğru sınıflandırma ve kümeleme başarımları için kullanılmıştır.

4.1.1. YSA İle Adaptif Özellik Çıkarma ve Sınıflama

Bu aşama sınıflandırma safhasıdır. Burada, özellik çıkarım safhasında elde edilen 1440×3 ($12 \times 120 \times 3$) özellik vektörünün yarısı yani 720×3 'lük özellik vektörü kullanılarak, yapay sinir ağları ile akıllı bir sınıflandırma yapılmıştır. Kullanılan çok katmanlı algılayıcı yapılı yapay sinir ağının eğitim parametreleri ve yapısı Tablo 4.1.'de gösterilmektedir.

Tablo 4.1. Çok katmanlı yapay sinir ağının yapısı ve eğitim parametreleri

Katman sayısı	3
Katman hücre sayıları	Giriş: 3, Gizli Katman: 10, Çıkış: 1
Aktivasyon fonksiyonları	Logaritmik sigmoid
Öğrenme algoritması	Geriye yayılım algoritması
Toplam karesel hata	0.00001

Bu değerler, örneğin gizli katman sayısı, gizli katmanlardaki hücre sayısı, öğrenme oranı ve moment sabitinin değeri ve aktivasyon fonksiyonu, birkaç farklı denemeden sonra en iyi performansı verebilecek şekilde seçilmiştir. RNA virüs türlerinin otomatik sınıflandırılması için geliştirilen bu yöntem, Çoklu Entropi ve Yapay Sinir Ağları (ÇE-YSA) olarak adlandırılmıştır.

Doğru sınıflandırma başarımları testi sonucunda elde edilen YSA için doğru sınıflandırma oranları Tablo 4.2. de gösterilmiştir.

Tablo 4.2. YSA için doğru sınıflandırma oranları

RNA Virüs Tipi	Doğru Sınıflandırılmış RNA Virüs Sayısı	Yanlış Sınıflandırılmış RNA Virüs Sayısı	Doğru Sınıflandırma Oranını (%)
PV	58	2	96.66
TV	57	3	95
FV	57	3	95
OV	60	0	100
BV	55	5	91.66
AV	54	6	90
RV	59	1	98.33
CV	60	0	100
REV	57	3	95
RHV	50	10	83.33
FLV	51	9	85
PRV	59	1	98.33
Toplam	677	43	94.02

Yapay sinir ağlarının RNA virüs görüntülerini tanıma başarımı Tablo 4.3. de gösterilmiştir.

Tablo 4.3. YSA' nın RNA virüs görüntülerini tanıma başarımı

	PV	TV	FV	OV	BV	AV	RV	CV	REV	RHV	FLV	PRV
PV	58	-	-	1	-	-	1	-	-	-	-	-
TV	-	57	-	-	1	-	1	-	-	-	-	1
FV	1	-	57	-	-	1	-	-	1	-	-	-
OV	-	-	-	60	-	-	-	-	-	-	-	-
BV	-	-	-	-	55	1	-	2	-	2	-	-
AV	-	-	3	-	-	54	-	-	1	-	-	2
RV	-	1	-	-	-	-	59	-	-	-	-	-
CV	-	-	-	-	-	-	-	60	-	-	-	-
REV	-	-	-	2	-	1	-	-	57	-	-	-
RHV	-	3	-	2	-	-	3	-	1	50	-	1
FLV	2	-	-	-	-	3	-	1	-	3	51	-
PRV	-	-	-	-	1	-	-	-	-	-	-	59

4.1.2. UATBÇS İle Adaptif Özellik Çıkarma ve Sınıflama

Burada, özellik çıkarım safhasında elde edilen 1440 x 3 (12 x 120 x 3) özellik vektörünün yarısı yani 720 x 3'lük özellik vektörü kullanılarak uyarlamalı ağ tabanlı çıkarım sistemi ile akıllı bir sınıflandırma yapılmıştır. Kullanılan çok katmanlı algılayıcı yapılı uyarlamalı ağ tabanlı çıkarım sisteminin eğitim parametreleri ve yapısı Tablo 4.4.'de gösterilmektedir.

Tablo 4.4. UATBÇS modelinin yapısı ve eğitim parametreleri

Katman sayısı	3
Katman hücre sayıları	Giriş: 3, Kural Sayısı: 8, Çıkış: 1
Aktivasyon fonksiyonları	Logaritmik sigmoid
Eğitim parametreleri Öğrenme kuralı	Melez Öğrenme Algoritması (Doğrusal olmayan giriş parametreleri için Geri yayılım ve doğrusal çıkış parametreleri için En küçük kareler yöntemlerinin birleşimleri)
Toplam karesel hata	0.000000001

Doğru sınıflandırma başarımlarını test sonucunda elde edilen UATBÇS için doğru sınıflandırılmış oranlar Tablo 4.5.'de gösterilmiştir.

Tablo 4.5. UATBÇS için doğru sınıflandırılmış oranlar

RNA Virus Tipi	Doğru Sınıflandırılmış RNA Virüs Sayısı	Yanlış Sınıflandırılmış RNA Virüs Sayısı	Doğru Sınıflandırma Oranı(%)
PV	57	3	95
TV	58	2	96.66
FV	57	3	95
OV	60	0	100
BV	58	2	95
AV	56	4	93.33
RV	60	0	100
CV	59	1	98.33
REV	57	3	95
RHV	55	5	91.67
FLV	50	10	83.33
PRV	59	1	98.33
Toplam	686	34	95.12

Uyarlamalı ağ tabanlı çıkarım sisteminin RNA virüs görüntülerini tanıma başarımı Tablo 4.6. da gösterilmiştir.

Tablo 4.6. UATBÇS'nin RNA virüs görüntülerini tanıma başarımı

	PV	TV	FV	OV	BV	AV	RV	CV	REV	RHV	FLV	PRV
PV	57	-	-	1	-	-	-	2	-	-	-	-
TV	-	58	-	-	-	-	1	-	-	-	-	1
FV	-	-	57	-	-	2	-	-	1	-	-	-
OV	-	-	-	60	-	-	-	-	-	-	-	-
BV	-	-	-	-	58	-	-	2	-	-	-	-
AV	-	-	2	-	-	56	-	-	1	-	-	1
RV	-	-	-	-	-	-	60	-	-	-	-	-
CV	-	-	1	-	-	-	-	59	-	-	-	-
REV	-	-	-	2	-	-	-	-	57	-	1	-
RHV	2	1	-	-	2	-	-	-	-	55	-	-
FLV	-	3	-	1	-	2	1	-	2	1	50	-
PRV	-	-	-	-	-	-	-	-	-	1	-	59

4.1.3. BCO İle Adaptif Özellik Çıkarma ve Kümeleme

Burada, özellik çıkarım safhasında elde edilen 1440×3 ($12 \times 120 \times 3$) özellik vektörünün yarısı yani 720×3 'lük özellik vektörü kullanılarak Bulanık C-Ortalamlar ile akıllı bir kümeleme yapılmıştır.

Doğru sınıflandırma başarımları testi sonucunda elde edilen BCO için doğru kümelendirilmiş oranlar Tablo 4.7.'de gösterilmiştir.

Tablo 4.7. BCO için doğru kümelendirilmiş oranlar

RNA Virus Tipi	Doğru Sınıflandırılmış RNA Virüs Sayısı	Yanlış Sınıflandırılmış RNA Virüs Sayısı	Doğru Sınıflandırma Oranı (%)
PV	52	8	86.66
TV	55	5	91.67
FV	54	6	90
OV	57	3	95
BV	50	10	83.33
AV	52	8	86.66
RV	60	0	100
CV	49	11	81.66
REV	51	9	85
RHV	50	10	83.33
FLV	54	6	90
PRV	60	0	100
Toplam	644	76	89.44

Bulanık c-ortalamaların RNA virüs görüntülerini tanıma başarımı Tablo 4.8. de gösterilmiştir.

Tablo 4.8. BCO'nun RNA virüs görüntülerini tanıma başarımı

	PV	TV	FV	OV	BV	AV	RV	CV	REV	RHV	FLV	PRV
PV	52	-	2	1	-	3	-	-	2	-	-	-
TV	-	55	-	-	3	-	-	2	-	-	-	-
FV	-	-	54	-	4	-	-	-	-	-	2	-
OV	1	-	2	57	-	-	-	-	-	-	-	-
BV	-	4	-	-	50	-	5	-	-	-	1	-
AV	-	-	-	3	-	52	-	3	-	-	2	-
RV	-	-	-	-	-	-	60	-	-	-	-	-
CV	-	-	-	4	-	3	-	49	1	3	-	-
REV	-	3	-	-	-	-	4	-	51	2	-	-
RHV	3	-	-	-	2	-	-	-	-	50	-	5
FLV	-	-	3	-	-	-	-	3	-	-	54	-
PRV	-	-	-	-	-	-	-	-	-	-	-	60

5. SONUÇLAR VE TARTIŞMA

Kümeleme algoritmaları kullanılarak tıbbi imgelerin sınıflandırılması çalışmalarında, en önemli iki sorun: Güçlü ve etkili özellik çıkarımı ve buna bağlı olarak elde edilen sınıflandırma ve kümeleme sonuçlarının güvenilirliğidir. Bu tez çalışmasında görüntülerin döndürülmesi, döndürülmüş görüntülerin ölçeklendirilmesi ve bunların entropilerinin hesaplanması özellik çıkarımı için kullanılmıştır. Buna göre her bir farklı RNA virüs türü görüntüsü için farklı açı ve büyüklükte 120 adet resim elde edilmiştir. Elde edilen resimlerin her biri için 3 adet entropi değeri hesaplanarak özellik vektörü elde edilmiştir. Bu yöntemlerin etkinliği, gerçek zaman uygulamaları ve üstün genelleme yetenekleri ile örüntü sınıflandırıcıları arasında önemli bir yere sahip olan YSA, UATBÇS sınıflayıcıları ve BCO kümeleyicileri ile ayrı ayrı değerlendirilmiştir.

Bu çalışma MATLAB ortamında bilgisayar simülasyonları sistemini yerine getiren ve RNA virüsleri için önerilen otomatik algılama performansını kullanıp test etmek amacıyla tasarlanmış ÇE-YSA, ÇE-UATÇBS ve ÇE-BCO sistemleridir.

5.1. Sonuçların Değerlendirilmesi

- Geliştirilen özellik çıkarım algoritmalarındaki çalışmaların aksine, görüntüyü doğrudan özellik uzayı olarak tanımlamak yerine, daha ayrıntılı ve tutarlı özellikler elde etmek için, ölçeklendirme ve döndürme ile görüntüye birden çok perspektiften bakılarak elde edilen her bir görüntü bileşeninin entropi değerleri hesaplanmıştır. Böylece görüntünün özellik uzayı, görüntünün farklı bakış açılarından çıkarılan her bir bileşeninin sahip olduğu üçer adet entropi değerlerinin oluşturduğu küme ile tanımlanmıştır.
- Kullanılan görüntülerin döndürülmesi, döndürülmüş görüntülerin ölçeklendirilmesi özellik çıkarım yapılarında görüntünün entropi değerleri hesaplanmıştır. Bu yöntemlerin YSA ve UATBÇS ile sınıflama başarımları ortalama olarak sırasıyla %94 ve %95 civarındadır. Ayrıca BCO kümeleyicisinin kümeleme başarımlar ortalaması %89 civarında başarımlar seviyesine ulaşmıştır.

- RNA virüslerinin yanlış sınıflandırılma ve kümelemelerinin temel sebebi birbirine benzer şekillerin olmasıdır. Bu hata payını en aza indirmek için RNA virüs görüntülerinde kullanılan akıllı görüntü tanıma sistemlerinin girişine, RNA virüs görüntülerini doğrudan vermek yerine, önerilen ön işlem süreci ile bu sistemlerin başarımına katkıda bulunulmuştur. RNA virüs görüntülerine uygulaması yapılan ön işlem sürecinde, gürültüden ayrışım, normalizasyon gibi aşamalar ile görüntü içerisinde bulunan serbest iç ve dış dinamiklere müdahale edilerek, görüntü tanıma sisteminin başarımına olumsuz yönde olan etkileri minimum düzeye çekilmiştir.

5.2. Öneriler

1. RNA virüs görüntüleri için yapılan uygulamada, YSA ve UATBÇS sınıflandırıcıları ile BCO kümeleyicisinin eğitimi için kullanılan veri örneklerinin sayısı arttırılarak, geliştirilen RNA virüslerinin sınıflandırılması sisteminin başarımı daha da yükseltilebileceği gibi RNA virüsleri hakkında ayrıntılı bilgilerde otomatik olarak çıkartılabilir.
2. Bu tez çalışmasında kullanılan ve geliştirilen algoritmalarda veri alma, özellik çıkarma, sınıflama ve kümeleme işlemlerinin daha hızlı yapılabilmesi problemi ilerde farklı yöntemler kullanılarak giderilebilir.

KAYNAKLAR

1. **Sammouda, R., Sammouda, M.,** 2003. Improving the performance of Hopfield Neural Network to segment pathological liver color images. CARS 2003, s.232-239.
2. **Esgiar AN, Naguib RNG, Sharif BS, Bennett MK, Murray A.,** 2002. Fractal analysis in the detection of colonic cancer images. IEEE Trans Info Technol Biomed; **6(1)**:54-8.
3. http://tr.wikipedia.org/wiki/Virus,_Virüs. 12 Mart 2009.
4. <http://www.webturkiyeportal.com/webforum/113459-rna-virusleri.html>, RNA Virüsleri. 12 Mart 2009.
5. http://complex.upf.es/~ricard/RNA_SITE.html, Virus Site. 29 Ocak 2010.
6. <http://tr.wikipedia.org/wiki/Retroviridae>, Retroviridae. 17Mart 2009.
7. **Demuth, H., Beale, M., Hagan, M.,** 2006. Neural Network Toolbox, for use with MATLAB, User Guide Version 5, Mathworks.
8. **Park J. W., Kim J. S., Cho W. J. Yoon W. J.,** 2004. Novel virus-inducible stress protein required for rhabdovirus growth, Science and Technology, KORUS 2004. The 8th Russian-Korean International Symposium on.
9. **Karaboga D., Basturk B.,** 2005. Diferansiyel Gelişim Algoritması ile İmge Bölütleme, IEEE 13. Sinyal İşleme ve İletişim Uygulamaları Kurultayı SİU-2005, Erciyes Üniversitesi, 16-18 Mayıs, Kayseri, Türkiye.
10. **Binczak, S., Sliwa, T., Jacquir, S. Bilbault J.M.,** 2010, Reaction–diffusion network for geometric multiscale high speed image processing, Image and Vision Computing, 28(6), s.914-926.
11. **Jurjo, D.L.B.R., Magluta, C., Roitman, N., Gonçalves, P.B.,** 2010. Experimental methodology for the dynamic analysis of slender structures based on digital image processing techniques, Mechanical Systems and Signal Processing, 24(5), s.1369-1382.
12. **Brunelli, R., Mich, O.,** 2001. Histograms analysis for image retrieval Pattern Recognition, 34(8)s.1625-163.
13. **Gonzalez, R.C. and Woods, R.E.,** 1993. Digital Image Processing, Addison-Wesley Publishing Company.

14. **Ding, L., ,Goshtasby, A.,** 2001. On The Canny Edge Detector, Pattern Recognition, 34(3), s.721-72.
15. **Kulkarni, A., D.,** 2001, Computer Vision and Fuzzy – Neural Systems, Prentice Hall PTR, USA, s.509.
16. **Shannon, C.E.,** 1948. A mathematical theory of communication, Bell System Technology Journal, 27, s. 379-423.
17. **Principe, J.C., Euliano, N.R. and Lefebvre, W.C.,** 2000. Neural and Adaptive Systems, John Wiley & Sons, 1. Baskı, New York, s. 656.
18. **Overwijk, M.H.F., and Reefman, D.,** 2000. Maximum-entropy deconvolution applied to electron energy-loss spectroscopy, Pergamon Micron, 31, s. 325–331.
19. **Li, X.,** 2000. Edge directed statistical inference with applications to image processing”, *Doktora Tezi*, Princeton Üniversitesi.
20. **Coifman, R.R. and Wickerhauser, M.V.,** 1992. Entropy-based algorithms for best basis selection, IEEE Transaction on Information Theory, 38 (2), s.713-718.
21. **Özbay, Y.,** 1999. EKG Aritmilerini Hızlı Tanıma, *Doktora Tezi*, Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya.
22. **Demuth, H., Beale, M., and Hagan, M.,** 2006. Neural Network Toolbox, for use with MATLAB, User Guide Version 5, Mathworks.
23. **Türkoğlu, İ.,** 2002. Durağan Olmayan İşaretler İçin Zaman-Frekans Entropilerine Dayalı Akıllı Örüntü Tanıma, *Doktora Tezi*, Fırat Üniversitesi Fen Bilimleri Enstitüsü, s. 112, Elazığ.
24. **Kelesoglu, O., ve Fırat, A.,** 2006. Tuğla Duvardaki ve Tesisattaki Isı Kaybının Yapay Sinir Ağları ile Belirlenmesi, Fırat Üniversitesi Fen ve Mühendislik Bilimleri Dergisi, 18(1), s.133-141.
25. **Türkoğlu, İ.,** 1996. Yapay Sinir Ağları ile Nesne Tanıma, *Yüksek Lisans Tezi*, Fırat Üniversitesi Fen Bilimleri Enstitüsü, Elazığ.
26. **Civalek, Ö.,** 1998. Plak ve Kabukların Nöro-Fuzzy Tekniği ile Lineer ve Non-Linear Statik-Dinamik Analizi, *Yüksek Lisans Tezi*, Fırat Üniversitesi, Elazığ.
27. **Feixue, W., Wenxian, Y., Guirong, G.,** 1996. Recurrent neural network for high-resolution radar ship target recognition, Radar, 1996. Proceedings., CIE International Conference of , 8-10 Ekim 1996, s.200 – 203.
28. **Zongquan, Y.,** 1998, Two-level classification of target recognition based on neural network, Microwave and Millimeter Wave Technology Proceedings, 1998. ICMMT '98. 1998 International Conference on, 18-20 Ağustos 1998, s.460 – 462.

29. **Daqing, C., Zheng, B.,** 1996. High range resolution radar target identification using multilayer feedforward neural network, Radar, Proceedings., CIE International Conference of , 8-10 Ekim 1996, s 215 – 218.
30. **Elmas, Ç.,** 2007. Yapay Zeka Uygulamaları, Seçkin Yayınları, Ankara, s.88.
31. **Ozçalık H. R., Uygur, A.F.,** 2003. Dinamik Sistemlerin Uyumlu Sinirsel-Bulanık Ağ Yapısına Dayalı Etkin Modellenmesi, KSÜ Fen ve Mühendislik Dergisi 6(1), Kahramanmaraş.
32. **Erkal B.,** 2009, Eğitim Amaçlı Sinirsel Bulanık Denetimine Dayalı Bir manyetik Askı Sisteminin Tasarlanması ve Simulink Ortamında Benzetimi, 5. Uluslar arası İleri Teknolojiler Sempozyumu Karabük, Türkiye.
33. **Jang, J.S.R.,** 1993. ANFIS: Adaptive-Network-Based Fuzzy Inference System, IEEE Trans. Systems, Man, Cybernetics, 23(5/6):s.665-685.
34. **Elmas, Ç.,** 2003, Bulanık Mantık Denetleyiciler, Seçkin Yayınları, Ankara, s.230.
35. **Höppner, F., Klawonn, F., Kruse, R., and Runkler, T.,** 2000. Fuzzy Cluster Analysis, John Wiley&Sons, Chichester.
36. **Türkoğlu, İ.,** 1999, Örüntü Tanıma Yöntemlerinin İncelenmesi, *Doktora Semineri*, Fırat Üniversitesi Fen Bilimleri Enstitüsü, Elazığ.
37. **Işık, M., Çamurcu, A. Y.,** 2007. K-MEANS K-MEDOIDS ve Bulanık C-Means Algoritmalarının Uygulamalı Olarak Performanslarının Tespiti, İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi, 6 (11) s.31-45.
38. **Seul, M., O’Gorman L., Sammon M.J.,** 2000. Practical Algorithms for Image Analysis, Cambridge University Press, United Kingdom, s.71-72.
39. **Dunn, J. C.,** 1973. A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters", Journal of Cybernetics 3: 32-57.
40. **Bezdek, J. C.,** 1981. Pattern Recognition with Fuzzy Objective Function Algoritms, Plenum Press, New York.

ÖZGEÇMİŞ

Öznur ERKUŞ

Fırat Üniversitesi
Teknik Eğitim Fakültesi
Elektronik ve Bilgisayar Eğitimi Bölümü
Elazığ

Tel: 424 – 2334883
E.posta: oznurerkus@hotmail.com

1985 Elazığ ‘da doğdu.

1999 – 2002 Elazığ Mehmet Akif Ersoy Lisesini tamamladı.

2003 – 2007 Fırat Üniversitesi Teknik Eğitim Fakültesi Elektronik – Bilgisayar Eğitimi Bilgisayar Öğretmenliği Bölümünden mezun oldu.

2007 – T.C. M.E.B. da öğretmen olarak çalışmaktadır.

2008 – Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik – Bilgisayar Eğitimi Anabilim Dalında yüksek lisans tez çalışmasına başladı.