

**ELEKTRONİK HARP DESTEK SİSTEMLERİNDE ANTEN  
TARAMA TİPİ SINIFLANDIRMASI**

**ANTENNA SCANNING TYPE CLASSIFICATION IN  
ELECTRONIC WARFARE SUPPORT SYSTEMS**

**EMİRHAN ÖZMEN**

**Dr. Öğr. Üyesi Yakup ÖZKAZANÇ**

**Tez Danışmanı**

Hacettepe Üniversitesi

Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin

Elektrik ve Elektronik Mühendisliği Anabilim Dalı için Öngördüğü

YÜKSEK LİSANS TEZİ olarak hazırlanmıştır.

2021

## ÖZET

# ELEKTRONİK HARP DESTEK SİSTEMLERİNDE ANTEN TARAMA TİPİ SINIFLANDIRMASI

Emirhan ÖZMEN

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Danışmanı: Dr. Öğr. Üyesi Yakup ÖZKAZANÇ

Aralık 2021, 81 sayfa

Bu çalışmada, farklı öznitelik çıkarma yöntemleriyle elde edilen öznitelik vektörleri gözetimli ve gözetimsiz öğrenme yöntemlerinde kullanılarak Elektronik Harp Destek Sistemlerinde kullanılan anten tarama tipleri belirlenmiştir. Ayrıştırılmış radar darbelerinin genlik örüntüleri önerdiğimiz modelleri besler ve ilgili tarama tipi otomatik olarak elde edilir. Gözetimli ve gözetimsiz öğrenme modellerine girdi sağlayan öznitelik vektörü farklı yöntemler kullanılarak oluşturulmuştur. Yapay veri setleri üzerinde yapılan deneyler önerilen Uzun-Kısa Süreli Bellek tabanlı Otomatik Kodlayıcı ile öznitelik çıkarma tekniğinin ele alınan problem için en iyi sonuçları verdiğini göstermektedir.

**Anahtar Kelimeler:** Elektronik Harp, Anten Tarama Tipi, Derin Öğrenme, Kümeleme, Otomatik Kodlayıcılar, Uzun-Kısa Süreli Bellek.

## **ABSTRACT**

# **ANTENNA SCANNING TYPE CLASSIFICATION IN ELECTRONIC WARFARE SUPPORT SYSTEMS**

**Emirhan ÖZMEN**

**Master of Science, Department of Electrical and Electronics Engineering**

**Supervisor: Asst. Prof. Yakup ÖZKAZANÇ**

**December 2021, 81 pages**

In this study, antenna scanning types used in Electronic Warfare Support Systems were determined by using the feature vectors obtained by different feature extraction methods in supervised and unsupervised learning methods. The amplitude patterns of the separated radar pulses feed into the patterns we propose and the corresponding scan type is automatically obtained. The feature vector that provides input to supervised and unsupervised learning models was created using different methods. Experiments on synthetic datasets show that the proposed feature extraction technique with Long-Short Term Memory based Autoencoder gives the best results for the problem under consideration.

**Keywords:** Electronic Warfare, Antenna Scanning Type, Deep Learning, Clustering, Autoencoders, Long-Short Term Memory.

## TEŞEKKÜR

Bu tezin hazırlanmasında bana yol gösteren danışmanım Dr. Yakup Özkazanç'a minnetle teşekkür ederim. Bana sadece araştırma yapmayı değil, hayattaki hedeflere nasıl ulaşacağımı öğretti. Üstün danışmanlık tecrübesi sayesinde kendi potansiyelimi keşfetmemi sağladığı için çok teşekkür ederim. Onun öğrencilerinden biri olduğum için kendimi her zaman gururlu ve şanslı hissedeceğim.

Son olarak, beni destekleyen, beni önemseyen ve beni her zaman seven sevgili eşim Aslınur Özmen'e, maddi ve manevi desteklerini esirgemeyen annem Huri Özmen ve babam Mehmet Özmen'e en derin şükranlarımı sunuyorum. Böyle harika bir ailenin parçası olduğum için kendimi çok şanslı hissediyorum.

# İÇİNDEKİLER

ABSTRACT .....	iii
TEŞEKKÜR.....	iv
İÇİNDEKİLER.....	v
ŞEKİLLER DİZİNİ.....	vii
TABLOLAR DİZİNİ .....	x
SİMGELER VE KISALTMALAR .....	xi
SÖZLÜKÇE.....	xiii
1. GİRİŞ.....	1
1.1 Tezin Amacı.....	3
1.2 Tezin Akışı.....	3
2. ELEKTRONİK HARP KAVRAMLARI .....	4
2.1 Radar Elektronik Harp Sistemi Parametreleri .....	5
2.1.1 Radyo Frekansı (RF).....	5
2.1.2 Darbe Tekrarlama Aralığı (PRI).....	5
2.1.3 Darbe Genişliği (PW) .....	6
2.1.4 Darbe Geliş Yönü (DOA).....	6
2.1.5 Darbe Genliği (PA) .....	6
2.1.6 Anten Tarama Tipi (ATT) ve Anten Tarama Periyodu (ATP).....	6
2.2 Anten Tarama Tipleri .....	8
2.2.1 Dairesel Tarama.....	10
2.2.2 Çift Yönlü Sektörel Tarama .....	10
2.2.3 Tek Yönlü Sektörel Tarama .....	11
2.2.4 Raster Tarama .....	11
2.2.5 Helezonik Tarama .....	12
2.2.6 Konik Tarama.....	12
2.2.7 Spiral Tarama.....	13
2.2.8 Lob Anahtarlama Tarama.....	13
2.2.9 Kesik Konik Tarama .....	14

3.	ANTEN TARAMA ÖRÜNTÜSÜ ÜRETECİ .....	15
3.1	EH Destek Sistemi Almaç Özellikleri.....	16
3.2	Radar Anteni Dönüş Özellikleri .....	16
3.2.1	Radar Anteni Konumu .....	17
3.3	Anten Tarama Tipi Listesi ve Veri Oluşturma .....	17
4.	ANTEN TARAMA TİPİ BELİRLEME ALGORİTMASI .....	20
4.1	Öznitelik Çıkarma Yöntemleri.....	25
4.1.1	İstatistiksel Öznitelik Çıkarma.....	25
4.1.2	Uzun-Kısa Süreli Bellek (UKSB) Tabanlı Otomatik Kodlayıcı ile Öznitelik Çıkarma .....	26
4.1.3	Evrişimsel Sinir Ağı (ESA) Tabanlı Otomatik Kodlayıcı ile Öznitelik Çıkarma .....	30
4.2	Gözetimsiz Öğrenme Yöntemi ile Anten Tarama Tipi Belirleme .....	32
4.2.1	K-Ortalamlar Kümeleme Algoritması .....	32
4.2.2	K-Ortalamlar++ Kümeleme Algoritması .....	33
4.2.3	Hiyerarşik Kümeleme Algoritması.....	34
4.2.4	Mesafe Ölçütleri.....	35
4.3	Gözetimli Öğrenme Yöntemi ile Anten Tarama Tipi Belirleme .....	37
4.3.1	Yapay Sinir Ağı (YSA) Tabanlı Sınıflandırma .....	37
4.3.2	Destek Vektör Makinesi (DVM).....	40
4.3.3	K-En Yakın Komşu (K-EYK) Algoritması .....	42
4.4	Benzetim Düzenekleri .....	43
4.4.1	Eğitim Düzenekleri.....	43
4.4.2	Test Düzenekleri.....	46
4.4.3	Öznitelik Vektörlerinin Görselleştirilmesi.....	48
5.	BENZETİM ÇALIŞMALARI .....	51
6.	SONUÇLAR.....	70
7.	KAYNAKLAR .....	72
	EKLER.....	75
	EK 1 - Tezden Türetilmiş Yayınlar .....	75
	EK 2 - Tezden Türetilmiş Bildiriler.....	75

## ŞEKİLLER DİZİNİ

Şekil 2.1.1 Radar parametrelerinin temsili gösterimi. ....	6
Şekil 2.1.2 Anten huzmesi ve 3dB huzme genişliği (BW).....	7
Şekil 2.2.1 Anten hareketleri ve tarama örüntüleri.....	9
Şekil 2.2.1.1. Dairesel Tarama Örüntüsü. ....	10
Şekil 2.2.2.1 Çift Yönlü Sektörel Tarama Örüntüsü.....	10
Şekil 2.2.3.1 Tek Yönlü Sektörel Tarama Örüntüsü.....	11
Şekil 2.2.4.1 Raster Tarama Örüntüsü.....	11
Şekil 2.2.5.1 Helezonik Tarama Örüntüsü.....	12
Şekil 2.2.6.1 Konik Tarama Örüntüsü.....	12
Şekil 2.2.7.1 Spiral Tarama Örüntüsü.....	13
Şekil 2.2.8.1 Lob Anahtarlamalı Tarama Örüntüsü.....	13
Şekil 2.2.9.1 Kesik Konik Tarama Örüntüsü.....	14
Şekil 3.1 Anten tarama örüntüsü üretici. ....	15
Şekil 3.1.1 EH sisteminin almaç özelliklerini giriş alanı.....	16
Şekil 3.2.1 Radar anteni dönüş özelliklerini giriş alanı.....	16
Şekil 3.3.1 Veri oluşturulması için listelenmiş anten tarama tipi özellikleri.....	17
Şekil 4.1 Ayırıştırma işlemi (Deinterleaving). ....	20
Şekil 4.2 Veri hazırlama işlemi.....	22
Şekil 4.3 Anten tarama tiplerinin zaman-frekans görüntüsü.....	23
Şekil 4.1.1.1 İstatistiksel öznitelik çıkarma yöntemi.....	25
Şekil 4.1.2.1 UKSB tabanlı Otomatik Kodlayıcılarla Öznitelik Çıkarılarak ATT'nin Belirlenmesi.....	27
Şekil 4.1.3.1 ESA tabanlı Otomatik Kodlayıcılarla Öznitelik Çıkarılarak ATT'nin Belirlenmesi.....	31
Şekil 4.3.1.1 Yapay Sinir Ağının Yapısı.....	38
Şekil 4.3.1.2 Yapay Sinir Ağı ile Çok Sınıflı Sınıflandırma.....	39
Şekil 4.3.2.1 Destek Vektör Makinesi ile Çok Sınıflı Sınıflandırma.....	42
Şekil 4.4.1.1 UKSBOK-ÖÇ modelinin Eğitimi.....	44
Şekil 4.4.2.1 UKSBOK-ÖÇ yöntemiyle Anten Tarama Tipinin Belirlenmesi.....	46
Şekil 4.4.2.2 ESAOK-ÖÇ yöntemiyle Anten Tarama Tipinin Belirlenmesi.....	47
Şekil 4.4.2.3 İÖÇ yöntemiyle Anten Tarama Tipinin Belirlenmesi.....	47
Şekil 4.4.2.4 ESAOK-ÖÇ yöntemiyle çıkarılan özniteliklerin t-SNE ile görselleştirilmesi. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.....	49
Şekil 4.4.2.5 İÖÇ yöntemiyle çıkarılan özniteliklerin t-SNE ile görselleştirilmesi. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.....	50

Şekil 4.4.2.6 UKSBOK-ÖÇ yöntemiyle çıkarılan özniteliklerin t-SNE ile görselleştirilmesi. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.....	50
Şekil 5.1 UKSBOK-ÖÇ yöntemi ile elde edilen özniteliklerin Mahalanobis mesafe ölçütü kullanılarak hiyerarşik kümeleme dendrogramı. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.....	52
Şekil 5.2 ESAOK-ÖÇ yöntemi ile elde edilen özniteliklerin Bray-Curtis mesafe ölçütü kullanılarak hiyerarşik kümeleme dendrogramı. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.....	54
Şekil 5.3 İÖÇ yöntemi ile elde edilen özniteliklerin Mahalanobis mesafe ölçütü kullanılarak hiyerarşik kümeleme dendrogramı. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.....	55
Şekil 5.4 Hiyerarşik Kümeleme Tabanlı Sınıflandırma (HKS) yöntemi.....	56
Şekil 5.5 İÖÇ yöntemi ile elde edilen özniteliklerin YSA ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.....	57
Şekil 5.6 İÖÇ yöntemi ile elde edilen özniteliklerin DVM ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.....	58
Şekil 5.7 İÖÇ yöntemi ile elde edilen özniteliklerin K-EYK ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.....	59
Şekil 5.8 İÖÇ yöntemi ile elde edilen özniteliklerin HKS ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.....	60
Şekil 5.9 ESAOK-ÖÇ yöntemi ile elde edilen özniteliklerin YSA ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.....	61
Şekil 5.10 ESAOK-ÖÇ yöntemi ile elde edilen özniteliklerin DVM ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.....	62
Şekil 5.11 ESAOK-ÖÇ yöntemi ile elde edilen özniteliklerin K-EYK ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.....	63
Şekil 5.12 ESAOK-ÖÇ yöntemi ile elde edilen özniteliklerin HKS ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.....	64



Şekil 5.13 UKSBOK-ÖÇ yöntemi ile elde edilen özneliklerin YSA ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlama{8}, Kesik Konik{9}. .....	65
Şekil 5.14 UKSBOK-ÖÇ yöntemi ile elde edilen özneliklerin DVM ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlama{8}, Kesik Konik{9}. .....	66
Şekil 5.15 UKSBOK-ÖÇ yöntemi ile elde edilen özneliklerin K-EYK ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlama{8}, Kesik Konik{9}. .....	67
Şekil 5.16 UKSBOK-ÖÇ yöntemi ile elde edilen özneliklerin HKS ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlama{8}, Kesik Konik{9}. .....	68

## TABLÖLAR DİZİNİ

Tablo 3.1 Üretilen Yapay Verilerin Özellikleri .....	18
Tablo 5.1 UKSBOK-ÖÇ yöntemi ile elde edilen öznitelik vektörünün farklı kümeleme yöntemlerindeki başarımlar yüzdesi.....	51
Tablo 5.2 ESAOK-ÖÇ yöntemi ile elde edilen öznitelik vektörünün farklı kümeleme yöntemlerindeki başarımlar yüzdesi.....	53
Tablo 5.3 İÖÇ yöntemi ile elde edilen öznitelik vektörünün farklı kümeleme yöntemlerindeki başarımlar yüzdesi.....	54
Tablo 5.4. Kullanılan ATT Belirleme yöntemlerinin zamansal performansları.....	69

## SİMGELER VE KISALTMALAR

EH	: Elektronik Harp
SIGINT	: Signals Intelligence
COMINT	: Communication Intelligence
ELINT	: Electronic Intelligence
RF	: Radyo Frekansı
DTA	: Darbe Tekrarlama Aralığı
DTK	: Darbe Tanımlayıcı Kelime
PW	: Pulse Width
PRI	: Pulse Repetion Interval
ATT	: Anten Tarama Tipi
ATP	: Anten Tarama Periyodu
FFT	: Fast Fourier Transform
UKSBOK	: Uzun-Kısa Süreli Bellek Tabanlı Otomatik Kodlayıcı
ESAOK	: Evrimsel Sinir Ağı Tabanlı Otomatik Kodlayıcı
ÖÇ	: Öznitelik Çıkarma
İÖÇ	: İstatiksel Öznitelik Çıkarma
t-SNE	: t-distributed Stochastic Neighbor Embedding
PDW	: Pulse Descriptor Word
TOA	: Time of Arrival
PRF	: Pulse Repetion Frequency
DOA	: Direction of Arrival
PA	: Pulse Amplitude
WGN	: White Gaussian Noise
STFT	: Short Time Fourier Transform

DFT	: Discrete Fourier Transform
NN	: Neural Network
DVM	: Destek Vektör Makinesi
YSA	: Yapay Sinir Ağları
SNR	: Signal to Noise Ratio
ReLU	: Rectified Linear Unit
BW	: Beam Width
YSA	: Yapay Sinir Ağı
HKS	: Hiyerarşik Kümeleme Tabanlı Sınıflandırma
K-EYK	: K-En Yakın Komşu
SVM	: Support Vector Machine

## SÖZLÜKÇE

Stokastik Komşu Gömme	: Stochastic Neighbor Embedding
Tekrarlayan	: Stagger
Değişken	: Agile
Anahtarlama	: Dwell-Switch
Elektronik Destek	: Electronic Support
Elektronik Taarruz	: Electronic Attack
Elektronik Korunma	: Electronic Protection
Darbe Genişliği	: Pulse Width
Darbe Tanımlayıcı Kelime	: Pulse Descriptor Word
Darbe Tekrarlama Aralığı	: Pulse Repetition Interval
Beyaz Gauss Gürültüsü	: White Gaussian Noise
Kısa Süreli Fourier Dönüşümü	: Short Time Fourier Transform
Çarpıklık	: Skewness
Basıklık	: Kurtosis
Yapay Sinir Ağı	: Artificial Neural Network
Destek Vektör Makinesi	: Support Vector Machine
Maksimum Ortaklama	: MaxPolling
Uzun-Kısa Süreli Bellek	: Long-Short Term Memory
Huzme Genişliği	: Beam Width
Discrete Fourier Transform	: Ayrık Fourier Dönüşümü



# 1. GİRİŞ

Elektronik Harp (EH), elektromanyetik spektrumun askeri alanlarda kullanılması olarak tanımlanır [1]. Elektronik Harp kavramının ilk oluştuğu zamanlarda tasarlanan EH sistemleri, kullanıcının EH sisteminin ölçtüğü ve hesapladığı parametreleri göz önüne alarak tahmin yapmasına dayanırken, yeni EH sistemleri kullanıcının becerilerine ve deneyimine olan bağımlılığı en aza indirmeyi amaçlamaktadır. Gelecekteki EH sistemlerinin, kullanıcıya ihtiyaç duymayan ve öğrenme işlemini kendi başına yapabilen bilişsel sistemler olması öngörülmektedir [2].

Radarın işlevi (arama, tespit, takip, vb.) ve tehdit düzeyi, Anten Tarama Tipi (ATT) değerlendirilerek tahmin edilebilir. ATT, tehdit radarlarının tanımlanması için önemli bir özelliktir. Bu nedenle, başarılı ATT analizi, EH sistemlerinde kritik öneme sahiptir. Ancak literatürde ATT analizi ile ilgili çok az çalışma bulunmaktadır. Yakın zamanda yayınlanan en kapsamlı çalışmalardan biri Barshan ve arkadaşlarının [3] yaptığı çalışmadır. Bu çalışmada bazı ön işleme adımlarından sonra, radar sinyalinden öznelik çıkarma ve klasik örüntü tanıma tabanlı yöntemlerle ATT belirlenmiştir. Barshan ve arkadaşları dört klasik sınıflandırıcının (Naive Bayes, Destek Vektör Makinesi, Karar Ağacı, Sinir Ağı) performanslarını beş temel ATT (Dairesel, Konik, Sektörel, Helezonik ve Raster) için karşılaştırmıştır. Literatürde yer alan sonraki çalışmalar temel olarak [3]'e dayanmaktadır. Bu çalışmalardan biri Kim ve arkadaşları [4] tarafından yapılmıştır. Kim ve arkadaşları radarların yanlış tanımlanmasını önlemek için 'Tanımlanmamış' adlı yeni bir sınıf kullanmıştır. [5]'te Ayazgök ve arkadaşları, sınıflandırıcıya yeni anten tarama tipleri eklemiştirler. Yeni eklenen spiral ve elektronik tarama tipleriyle beraber korelasyona dayalı ana aydınlatma bulma yöntemi önermişlerdir. Bu alandaki en yeni çalışmalardan biri [6]'dır. Bu çalışmada sinyalin her bir aydınlatmasındaki en iyi tepe noktasının yerine belirlemek için yeni bir yöntem önerilmiş ve elde edilen öznelikler, sinir ağı tabanlı sınıflandırma yöntemi kullanılarak ATT belirlenmiştir. Yaptığımız literatür araştırmalarını göz önüne aldığımızda literatürde anten tarama tipini belirlemek için derin öğrenme tabanlı öznelik çıkarma yöntemi bulunmamaktadır. Geliştirilecek olan derin öğrenme tabanlı öznelik çıkarma yönteminin mevcut yöntemlerin bazı eksikliklerini giderebileceği düşünülmektedir.

İlk olarak, mevcut tüm yöntemler 4-5 tarama periyodu boyunca veri toplamayı gerektirir, bu da tipik 10 saniyelik bir anten tarama periyoduna sahip radar için 40-50 saniyelik bir veri toplama süresi anlamına gelir. Düzinelerce radarın aynı anda çalıştığı gerçek EH senaryoları göz önüne alındığında, EH sistem alıcısını tek bir radar için 40-50 saniye tahsis etmek EH sisteminin desteklediği misyonu tehlikeye atacaktır.

İkinci olarak, radar antenleri genellikle dar huzme genişliklerine sahiptir. Bu nedenle, EH sistemi radarın tarama süresinin belirli bir kısmı için, tipik olarak yüzde biri için veri alabilir. Tarama süresinin kalan kısımlarında radarın ana huzmesi farklı yönleri gösterdiği için radar sinyalleri alınamaz.

FFT tabanlı algoritmalar geleneksel olarak EH sistemlerinde anten tarama periyodu tahmini için kullanılır. Alınan verilerin boş kısımları, FFT analizini gerçekleştirmek için sıfırlar eklenerek doldurulur. FFT analizi için düzenli olarak örneklenen veriler gereklidir. Bu yüzden, FFT analizi yapılabilmesi için radarların sabit Darbe Tekrarlama Aralığı (PRI) değeriyle çalışması gerekmektedir. Fakat, operasyonel sahada kullanılan radarların çoğu kademeli (stagger), değişken (jitter) veya anahtarlamalı (dwell&switch) gibi karmaşık PRI türleri kullanır [1]. Mevcut yöntemlerde toplanan veriler, ek ön işleme gerektiren interpolasyon teknikleri kullanılarak düzenli olarak örneklendirilir. Bu nedenle, mevcut yöntemler gerçek zamanlı işleme senaryolarına doğrudan uygulanamaz.

Uzun-Kısa Süreli Bellek tabanlı Otomatik Kodlayıcıyla Öznitelik Çıkarma (UKSBOK-ÖÇ) yönteminde, EH sisteminin ölçtüğü ve hesapladığı parametreler kullanılarak ATT sınıflandırma işlemi için öznitelikler belirlenmiştir. Evrimsel Sinir Ağı tabanlı Otomatik Kodlayıcıyla Öznitelik Çıkarma (ESAOK-ÖÇ) ve İstatistiksel Temelli Öznitelik Çıkarma (İÖÇ) yöntemlerinde girdi verisinin bazı ön işleme adımlarından geçmesi gerekir. Önerilen UKSBOK-ÖÇ yönteminin en önemli avantajı, veri temizleme kısmı hariç başka bir ön işleme adımı olmadığı için zaman kaybı olmamasıdır.



## **1.1 Tezin Amacı**

Bu çalışma kapsamında, EH sisteminin düşman radarların anten tarama tiplerini hızlı ve doğru belirleyebilmesi için makine öğrenmesi tabanlı yöntemler araştırılmış ve yeni bir anten tarama tipi belirleme yöntemi geliştirilmesi amaçlanmıştır. Bu çerçevede, literatürdeki benzer çalışmalar etüd edilerek makine öğrenmesi konusunda sistematik yaklaşımlar ortaya konmuştur. Ek olarak; Uzun-Kısa Süreli Bellek tabanlı Otomatik Kodlayıcılar kullanılarak özgün bir öznitelik çıkarma yöntemi tasarlanmış ve problemin doğasına uygun olarak kullanılmıştır.

## **1.2 Tezin Akışı**

Bu çalışmanın ilk 3 bölümünde elektronik harp sistemleri hakkında bilgi sahibi olmak ve önerilen algoritmaları daha iyi anlayabilmek için literatür ve alan bilgisi sunulmuştur.

Bölüm 1’de elektronik harp sistemleri hakkında temel bilgiler verilmiş ve literatürdeki çalışmalar özetlenmiştir. Bölüm 2’de elektronik harp sistemleriyle ilgili alanlar hakkında bilgi verilmiştir. Bu bölümde ayrıca radar elektronik harp sisteminin ölçtüğü parametreler açıklanmıştır. Bu çalışmanın belirlemeye çalıştığı anten tarama tipleri detaylı olarak açıklanmıştır. Bölüm 3’te algoritma geliştirme çalışmalarında kullanılacak olan verileri oluşturmak için tasarlanan anten tarama örüntüsü üretici hakkında bilgiler verilmiştir. Bölüm 4’te ise anten tarama tipi belirleme algoritmasında kullanılacak öznitelik çıkarma, gözetimli ve gözetimsiz öğrenme yöntemleri detaylı olarak anlatılmıştır. Bu yöntemler ile kurulan benzetim düzenekleri hakkında bilgi verilmiştir. Bölüm 5’te önerilen algoritmalar çalıştırılarak alınan sonuçlar incelenmiş ve algoritmaların avantaj ve dezavantajları karşılaştırmalı olarak belirlenmiştir. Bölüm 6’da, yapılan çalışmalar hakkında genel değerlendirme yapılmış ve tasarlanan algoritmaların gerçek hayatta uygulaması hakkında öneriler verilmiştir.

## 2. ELEKTRONİK HARP KAVRAMLARI

Elektronik harp (EH), dost unsurları düşman unsurlardan korumak için elektromanyetik spektrumun kullanılmasıyla oluşan harp tekniğidir. Elektromanyetik spektrum, doğru akımdan ışık ve ötesine kadar geniş bir çerçevede değerlendirilir [7]. Böylece elektronik harp, tepe radyo frekansı spektrumunu, kızılötesi spektrumu, optik spektrumu ve ultraviyole spektrumunu kapsar. Elektronik harp sistemleri çalışma amaçlarına göre üç gruba ayrılırlar.

- Elektronik harp destek sistemleri; elektronik harp sisteminin alıcı kısmıdır. Bu sistemler herhangi bir spektrumda aktif olarak sinyal yaymazlar. Sadece dinleme görevi yaparlar.
- Elektronik harp taarruz sistemi; radarların, askeri iletişimin ve elektro optik sistemlerin çalışmasına müdahale etmek için kullanılan karıştırma sistemleridir. Aynı zamanda anti radyasyon silahları ve yönlendirilmiş enerji silahları da bu gruba dahildir.
- Elektronik korunma tedbirleri; elektronik taarruz sistemlerinin etkilerine karşı koymak için radarların veya iletişim sistemlerinin tasarımında veya işletiminde alınan önlemlerdir.

Elektronik harp destek sistemleri ve elektronik harp taarruz sistemleri beraber çalışabildiği gibi ayrı ayrı da çalışabilmektedir.

Tüm bu alanlar düşman iletimlerinin alınmasını içerse de elektronik destek sistemleri, sinyal istihbaratının (SIGINT) bir alt grubu olan haberleşme istihbaratı (COMINT) ve elektronik istihbarat (ELINT) sistemlerinden bazı farklılıklar içermektedir. Sinyallerin karmaşıklığı arttıkça giderek daha belirsiz hale gelen farklılıklar, sinyallerin alınma amaçlarına göre özel sistemler geliştirilmesini zorunlu hale getirmiştir.

- COMINT sistemleri, düşman unsurların sinyallerinden istihbarat çıkarmak amacıyla düşman iletişim sinyallerini alır ve işlerler.
- ELINT sistemleri, karşı önlemlerin geliştirilebilmesi için düşmanın radar sistemlerinin çalışma karakteristiklerinin ayrıntılarını belirlemek amacıyla düşmanın radar sinyallerini alır ve işlerler. Bu nedenle, ELINT sistemleri ayrıntılı bir analiz yapılabilmesi için uzun bir süre boyunca çok sayıda veri toplar.
- Elektronik destek sistemleri, sinyaller veya bu sinyallerle ilişkili silahlar hakkında hemen bir şeyler yapmak amacıyla düşman sinyallerini toplar. Alınan sinyaller hızlı bir şekilde işlenir ve düşman sisteme karşı müdahale yapılır. Alınan sinyaller durum

farkındalığı için de kullanılabilir. Yani, düşman kuvvetlerinin, silahlarının veya elektronik yeteneklerinin türlerini ve yerlerini belirlemek için kullanılabilir. Elektronik destek sistemleri, yüksek verim oranıyla daha dar bir alandan alınan verileri işlemek için çok sayıda sinyal verisi toplar. Elektronik destek sistemleri tipik olarak parametreleri bilinen düşman sistemlerinin hangisinin aktif olarak taktik sahada çalıştığı ve nerede olduğunu belirler.

Bu sistemlerin genel olarak amacı, radar ya da haberleşme sinyallerini yakalayıp analiz ederek bu sistemler hakkında bilgi toplamaktır. Bu sistemler operasyonel ortamdaki çalışma zamanlarının büyük bir kısmında pasif olarak çalışır. Bu sistemler ortamdaki topladığı sinyallerin Radyo Frekansı (RF), Darbe Genişliği (PW) ve Darbe Varış Zamanı (TOA) parametrelerini hesaplayarak radarların karakteristik bilgilerini karakterize eden Darbe Tanımlayıcı Kelimelere (DTK) dönüştürür. DTK'lerdeki TOA bilgisi kullanılarak darbe tekrarlama aralığını (PRI) veya darbe tekrarlama frekansını (PRF) belirlenir.

## **2.1 Radar Elektronik Harp Sistemi Parametreleri**

Bu bölümde Elektronik harp sistemleri tarafından ölçülen ve hesaplanan radar parametreleri hakkında bilgi verilmiştir. Bu parametreler kullanılarak radar sisteminin tespiti ve teşhisi yapılmaktadır. Şekil 2.1.1'de radar parametrelerinden bazıları temsili olarak gösterilmiştir.

### **2.1.1 Radyo Frekansı (RF)**

Her bir radar sistemi, tasarımına ve çalışma amacına göre belirlenmiş bir veya birkaç frekansta çalışabilir. Radar tarafından gönderilen her darbe, bu frekanslardan biriyle modüle edilir. Genel olarak radarların frekansı 1-40 GHz arasında değişmektedir. Bazı radarlar karıştırma sistemlerine karşı önlem almak için frekanslarında atlama ya da rastgele değişimler yapabilirler. Bu gibi yöntemler radar sisteminin EH sistemleri tarafından aldatılmasını önlemektedir.

### **2.1.2 Darbe Tekrarlama Aralığı (PRI)**

Bir radar darbesinin başlangıç zamanı ile bu radar darbesinin ardından gelen radar darbesinin başlangıç zamanı arasındaki farka Darbe Tekrarlama Aralığı (PRI) denir. Radarlar farklı türde PRI modelleri kullanabilirler. Bunlar sabit, kademeli (Stagger), değişken (Jitter), anahtarlamalı (Dwell&Switch) PRI modelleridir. Sabit PRI modelinde,

darbeler arası zaman farkı sabittir. Kademeli modelde ise PRI belirli bir örüntü içinde tekrar eder. Değişken modelde, PRI değeri rastgele olarak değişmektedir. Anahtarlamalı modelde ise belirli bir darbe adetinde PRI değeri sabitken bu adet kadar darbe gönderildikten sonra PRI değeri değiştirilmektedir.

### 2.1.3 Darbe Genişliği (PW)

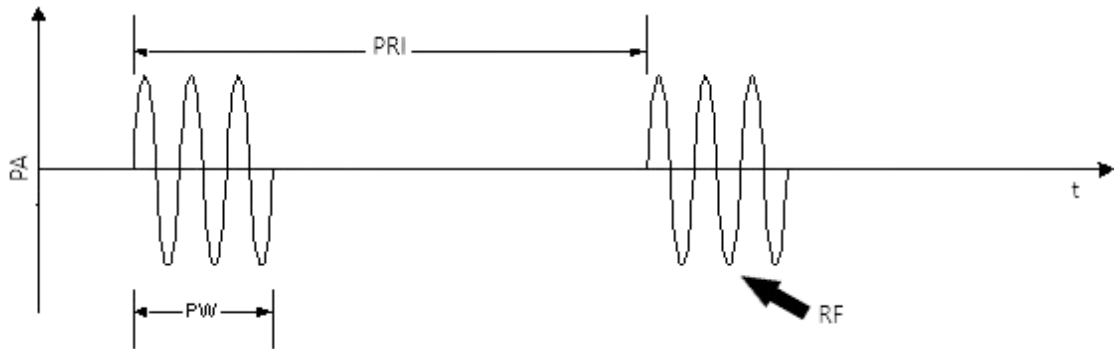
Darbenin başlama ve bitiş zamanları arasındaki farka Darbe Genişliği (PW) denir. Radar sistemleri darbe genişliğini uzaklık çözünürlüğünü iyileştirmek veya karıştırma sistemlerine karşı önlem alabilmek için kullanabilmektedir.

### 2.1.4 Darbe Geliş Yönü (DOA)

Radar sisteminin EH sistemine göre yanca ve yükselişteki açısına Darbe Geliş Yönü (DOA) denir. Darbe geliş yönünün bulunabilmesi için EH sistemlerinde kullanılan farklı algoritmalar mevcuttur. DOA genellikle EH platformundaki yönlü antenlerden alınan PA oranları kıyaslanarak bulunur. Bazı EH sistemleri, genlik karşılaştırma tekniklerine göre daha düşük hata oranına sahip olan faz karşılaştırmalı teknikleri de kullanmaktadır [8].

### 2.1.5 Darbe Genliği (PA)

Darbe Genliği (PA), EH sistemi tarafından alınan darbenin gücüdür. Bu parametre radar sisteminin çıkış gücü, anten kazancı, konumu, ortama göre oluşan yayılma kaybı ve EH sisteminin anten kazancı ile ilgilidir.



Şekil 2.1.1 Radar parametrelerinin temsili gösterimi.

### 2.1.6 Anten Tarama Tipi (ATT) ve Anten Tarama Periyodu (ATP)

Antenler radar sistemlerinin vazgeçilmez parçalarıdır. Elektromanyetik dalgaları almak veya göndermek için antenler kullanılır. Antenler kullanılma amaçlarına göre farklı

özelliklere sahiptirler. Radar sistemleri, farklı işlevleri yerine getirmek için farklı tasarım mimarilerine sahip antenler kullanırlar. Farklı anten tasarımlarının karakteristik özelliklerini gösteren parametreler vardır. Bu parametreler; anten radyasyon örüntüsü, kazanç, 3 dB huzme genişliği gibi parametrelerdir. Radyasyon örüntüsü, antenden çıkan gücün uzayda dağılımını gösterir. Kazanç, antenden çıkan toplam gücün ne kadarının kaybedildiğini gösterir. 3 dB huzme genişliği, yayılan gücün, antenin ön görüş yönünde yayılan maksimum gücün yarısından fazla olduğu açısız sektörü tanımlar. Şekil 2.1.2’de anten huzmesi ve 3dB huzme genişliği temsili olarak gösterilmiştir.

Bütün radarlar ilgilendiği bölgeyi taramak için farklı tarama yöntemleri kullanır. Bu tarama yöntemlerine anten tarama tipi denir. Yaygın olarak kullanılan anten tarama tipleri şunlardır: Dairesel, Tek Yönlü Sektörel, Çift Yönlü Sektörel, Raster, Helezonik, Konik, Spiral, Lob Anahtarlamalı ve Kesik Konik.

Anten tarama tipleri periyodik bir örüntüye sahiptir. Bu periyodiklik radar anteninin hedefini belirli zaman aralıklarıyla aydınlatmasının sonucunda oluşur. Aydınlatmaların tepe noktaları arasındaki zamana anten tarama periyodu (ATP) denir. Bölüm 2.2’de, her bir tarama tipi açıklanacak ve EH sistemi tarafından nasıl görüldüğü incelenecektir.



Şekil 2.1.2 Anten huzmesi ve 3dB huzme genişliği (BW).

## 2.2 Anten Tarama Tipleri

Genel olarak radar sistemleri, uzaysal çözünürlüğü ve algılama aralığını arttırmak için dar bir gönderici/alıcı anten kullanırlar. Hedefleri tespit etmek, arama modunda çalışan bir radarın başlıca görevidir. Bu yüzden radar anteninde huzme hareketi esastır. Bu radar huzme hareketine tarama, bu hareketten kaynaklanan örüntüye tarama örüntüsü denir. Tarama modeli, radarın gözetleme alanının bütününe kapsayacak şekilde tasarlanmıştır. Radar huzmesi mekanik veya elektronik olarak hareket edebilmektedir. Mekanik taramalı radarlarda, radar anteni fiziksel olarak döndürülerek huzme hedefe doğru yönlendirilir. Bu nedenle, huzme hareketi mekanik taramalı radarlarda periyodiktir. Elektronik tarama radarlarında, faz dizili antenler tarafından uzaysal örnekleme kullanılır. Huzme, her bir anten elemanının fazlarının yönlendirildiği yöne ve çalışma frekansına göre kaydırılarak yönlendirilir. Dolayısıyla anten hareketi periyodik olabilir veya olmayabilir. Mekanik taramalı radarlar mevcut radar sistemlerinin çoğunu oluştururken, elektronik taramalı radarlarının sayısı da hızla artmaktadır.

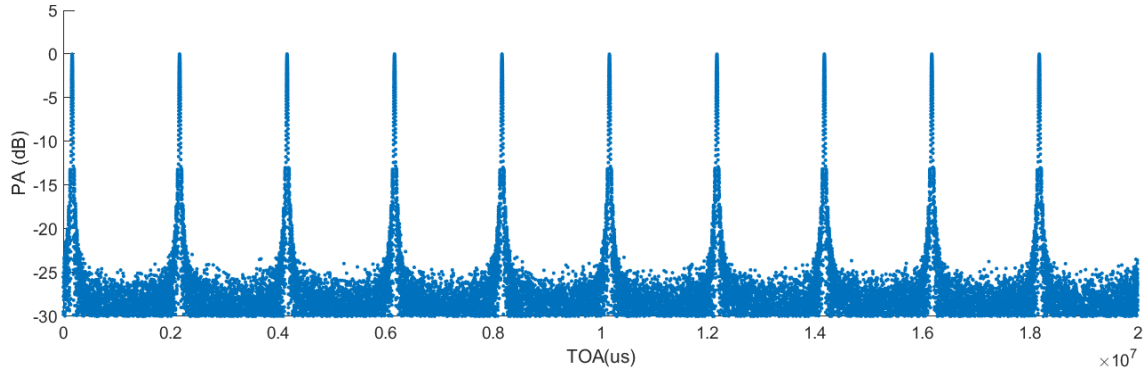
EH destek sistemi üzerinden geçen her radar huzmesine, radar sisteminin EH sistemi üzerindeki aydınlatması denir. Dairesel tarama tipinde bir periyod boyunca yalnızca bir aydınlatma alınır. Sektörel tarama tipinde, bir periyod boyunca eşit genlikte iki aydınlatma alınır. Radar sisteminin tarama sektöründe bulunan EH sisteminin konumu, EH sisteminin aldığı aydınlatmalar arasındaki zaman farkını belirler. Raster taramada bir periyod boyunca alınan aydınlatmaların genlik değerleri ve aydınlatmalar arasındaki zamansal farklar değişkenlik gösterir. Helezonik taramada bir periyod boyunca alınan aydınlatmaların genlikleri birbirinden farklıdır, fakat aydınlatmalar arasındaki zamansal farklar aynıdır. Konik ve Spiral taramalarda radar anteni EH Destek sistemini sürekli aydınlatmaktadır. Sürekli aydınlatma sırasında toplanan veriler sinüzoidal bir genlik örüntüsüne sahiptir. Konik taramada tepe genlikleri sabit iken, Spiral taramada tepe genlikleri değişkendir. Mevcut radar sistemlerinde kullanılan ve bu çalışmada belirlenmesi hedeflenen ATT'lerin anten hareketi modeli ve tarama örüntüleri Şekil 2.2.1'de verilmiştir.

Anten Tarama Tipi	Anten Hareketi	Anten Tarama Örüntüsü
Dairesel Tarama		
Sektörel Tarama		
Raster Tarama		
Helezonik Tarama		
Konik Tarama		
Spiral Tarama		
Lobe Anahtarlamalı Tarama		

Şekil 2.2.1 Anten hareketleri ve tarama örüntüleri.

### 2.2.1 Dairesel Tarama

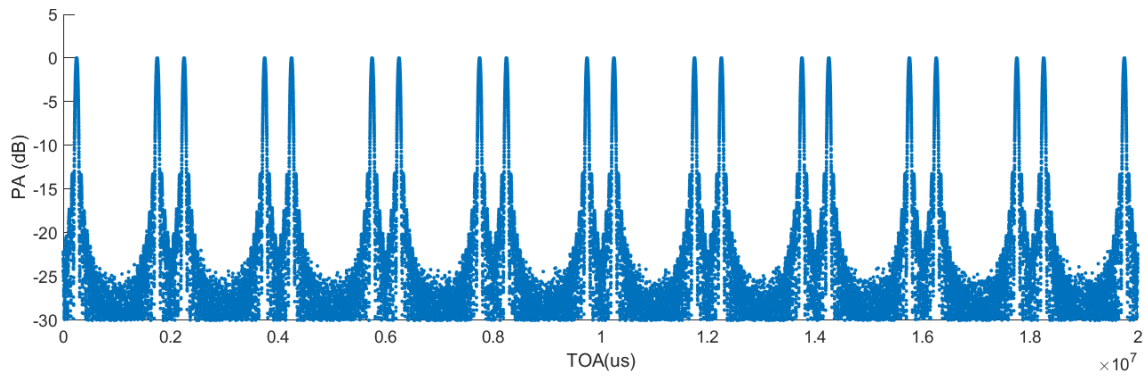
Arama radarlarında yaygın olarak kullanılan tarama tipidir. Anten, yarıda sabit bir hızda 360 ° tarar, yükselişte açısı sabit tutulur. Anten hızı, ilgilenilen bölgeyi kapsayacak şekilde yükselişte geniş ve iyi bir çözünürlük sağlamak için yarıda dardır. Bu tarama tipine sahip radarların menzil bilgisini ve yarıda açı bilgisini sağlaması beklenir, ancak yükseklik bilgisini sağlayamaz. Maksimum algılama mesafesine bağlı olarak, tarama süresi 1-10 saniye arasında değişebilir. Tarama örüntüsü Şekil 2.2.1.1’de gösterilmiştir.



Şekil 2.2.1.1. Dairesel Tarama Örüntüsü.

### 2.2.2 Çift Yönlü Sektörel Tarama

Çift Yönlü Sektörel tarama, belirlenmiş yarıda açı aralığında tarama yapan arama radarları tarafından kullanılır. Radar belirli bir açısal sektörü sabit açısal hızda ileri geri süpürür. Radar anteni, tarama sektörünün sonuna geldiğinde ana hızını ters yönde hareket ettirerek yalnızca ilgili sektör açı aralığı taramış olur. EH sistemi anteni, Şekil 2.2.2.1 'de gösterildiği gibi periyodik ve simetrik ana hızlar alır. Her tam periyod için eşit tepe noktalarına sahip iki ana hız alınır.

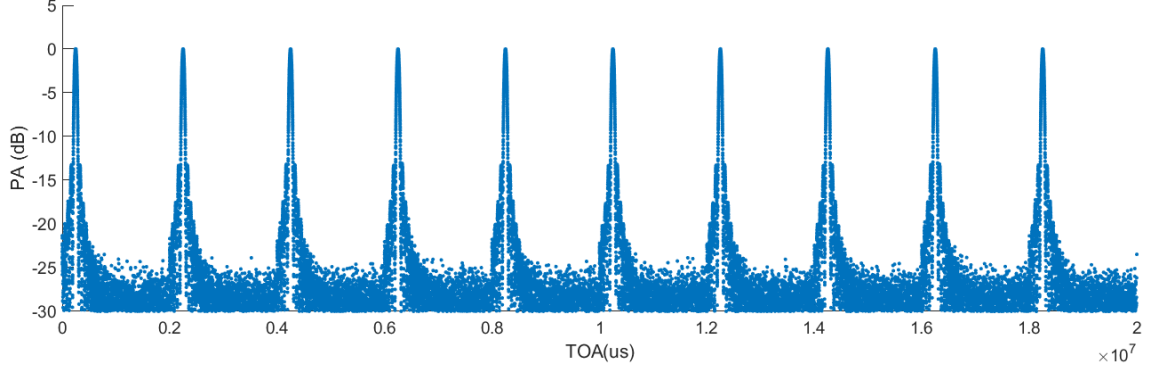


Şekil 2.2.2.1 Çift Yönlü Sektörel Tarama Örüntüsü.



### 2.2.3 Tek Yönlü Sektörel Tarama

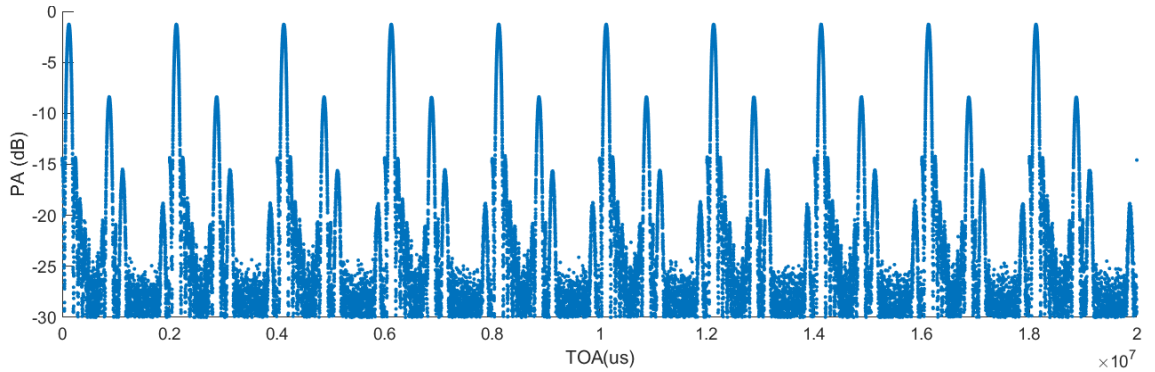
Çift Yönlü Sektörel taramadan ayrılan özelliği, radar anteni tarama bölgesinin sonuna geldiğinde ters yönde taramaya devam etmez ve taramaya başladığı konuma aydınlatma yapmadan döner. Radar anteni, sabit yükseliş açısına sahiptir, yükselişte tarama yapmaz. Anten tarama örüntüsü Şekil 2.2.3.1’de gösterilmiştir.



Şekil 2.2.3.1 Tek Yönlü Sektörel Tarama Örüntüsü.

### 2.2.4 Raster Tarama

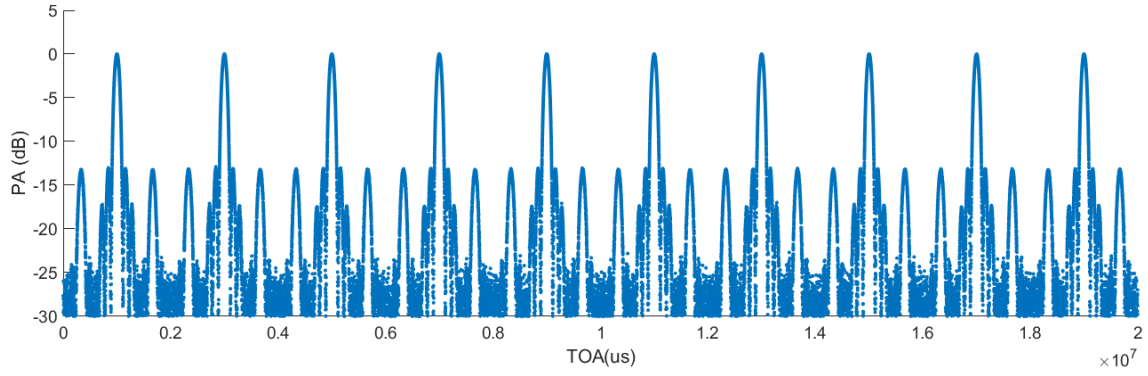
Raster taraması hem yarıda hem de yükselişte tarama yapar. Periyod değeri sabittir ve her tam periyotta, raster taramasının her kademesi için bir ana huzme yakalanır. Ancak, bu tarama tipinde yükselişte açı değiştiğinden, EH sisteminin her kademe aldığı PA değeri, yükselişteki açı değiştiği için farklı olabilir. Anten tarama örüntüsü Şekil 2.2.4.1’de gösterilmiştir.



Şekil 2.2.4.1 Raster Tarama Örüntüsü.

### 2.2.5 Helezonik Tarama

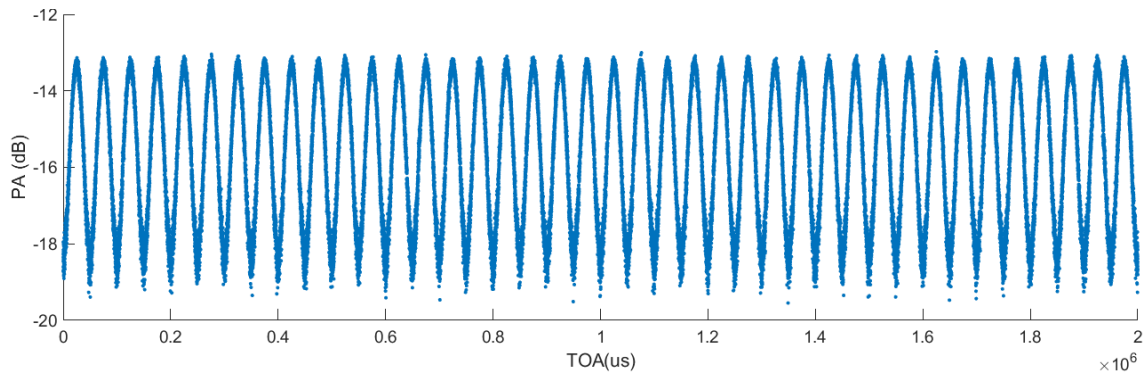
Helezonik tarama, genellikle tespit (acquisition) radarlarında kullanılır. Radar anteni yarıda 360 dereceyi sabit hızla tarar iken yükselişte belirli bir sektör açısını tarar. Bu nedenle, Helezonik tarama modeli sadece yarı açı bilgisini değil aynı zamanda yükseliş açısı bilgisini de içerir. Arama işleminde oluşan boşluğu azaltmak için Helezonik tarama işlemi sırasında daha yüksek bir tarama frekansı kullanılabilir. Helezonik tarama örüntüsü Şekil 2.2.5.1'de gösterilmiştir.



Şekil 2.2.5.1 Helezonik Tarama Örüntüsü.

### 2.2.6 Konik Tarama

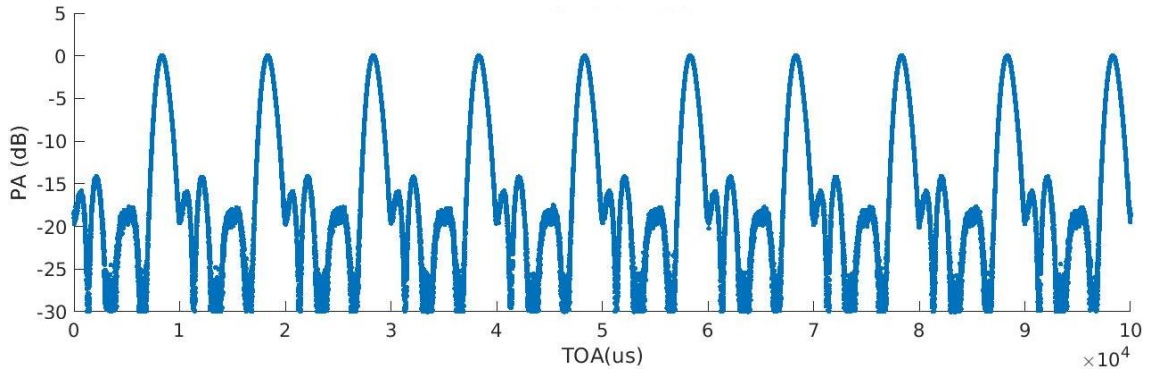
Konik tarama modeli genellikle hedef takibi için kullanılır ve alınan sinyalin genliği, EH sisteminin konumu ile önemli ölçüde ilişkilidir. Radarın anten tarama tipi konik tarama olduğunda ve EH sisteminin antenin bakış açısı konik tarama içinde kalıyorsa, alınan sinyalin genlik örüntüsü sinüzoidal bir dalga formu olan dairesel taramaya benzemektedir [5]. EH sisteminin antenin bakış açısı dairenin merkezinde olduğunda EH sisteminin gözlemlediği tarama örüntüsü sabit genliklidir. Konik tarama örüntüsü Şekil 2.2.6.1'de gösterilmiştir.



Şekil 2.2.6.1 Konik Tarama Örüntüsü.

### 2.2.7 Spiral Tarama

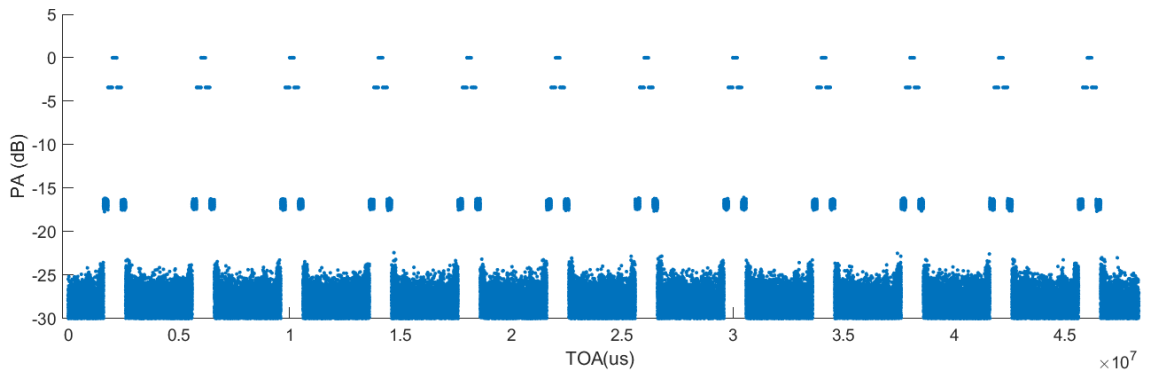
Spiral tarama, belirli bir dairesel bölgeyi taramak için kullanılan bir tür anten taramasıdır. Anten huzmesi ilgili bölgenin merkezinden taramaya başlar ve huzmesini dışa doğru hareket ettirir [9]. Tarama bölgesi tamamen tarandıktan sonra, anten huzmesi tarama bölgesinin merkezine geri döner. Bazı radar tasarımlarında, dışa doğru tarama tamamlandıktan sonra taramaya ters yönde içe doğru devam eder. Spiral tarama, huzme hareketinde ve tarama modelinde konik taramaya benzer. Anten huzmesinin açısı spiral taramada artar ve azalır, konik taramada açı sabit kalır. Spiral tarama örüntüsü Şekil 2.2.7.1’de gösterilmiştir.



Şekil 2.2.7.1 Spiral Tarama Örüntüsü.

### 2.2.8 Lob Anahtarlamalı Tarama

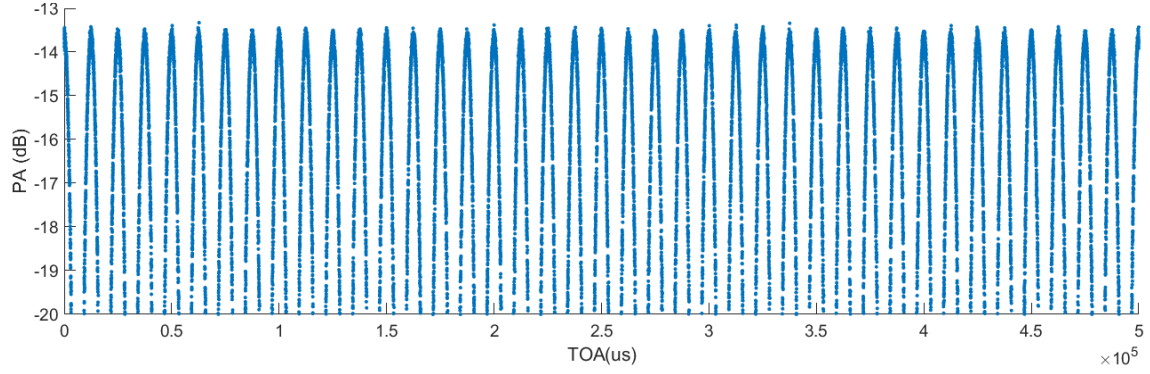
Lob Anahtarlamalı Tarama, takip radarlarının takip doğruluğunu arttırmak için ana huzmelerini farklı açılarla konumlandırması sonucu oluşur. Konik taramadaki sürekli dönen huzme hareketi yerine farklı pozisyonlarda huzme oluşturur. Çoğunlukla iki veya dört loba sahiptir. Lob Anahtarlamalı tarama örüntüsü Şekil 2.2.8.1’de gösterilmiştir.



Şekil 2.2.8.1 Lob Anahtarlamalı Tarama Örüntüsü.

## 2.2.9 Kesik Konik Tarama

EH sisteminin anteninin bakış açısı radar sisteminin konik tarama alanı dışında kalıyorsa sinüzoidal dalga örüntüsünde kesilmeler oluşur. Bu kesilmelerin sebebi EH sisteminin aldığı bazı darbelerin gücünün EH sisteminin almacının duyarlılığından düşük olmasıdır. Şekil 2.2.9.1 'de gösterilen örüntü tipine Kesik Konik tarama denilmiştir.

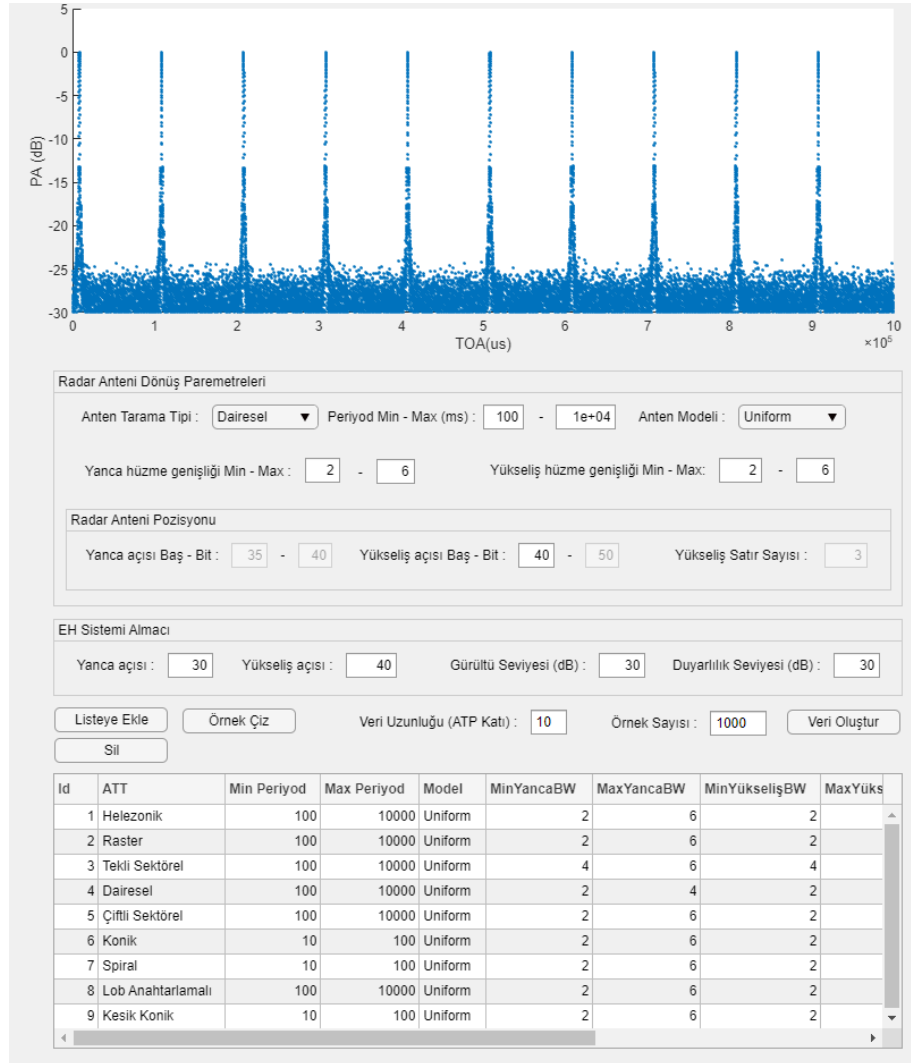


Şekil 2.2.9.1 Kesik Konik Tarama Örüntüsü.

### 3. ANTEN TARAMA ÖRÜNTÜSÜ ÜRETECİ

Gerçek radar verileri radarın sahada çalışma karakteristiğini gösterdiğinden dolayı bu verilere ulaşmak kolay değildir. Bu çalışmada kullanılmak üzere anten tarama örüntüsü üretici tasarlanmıştır. Bu bölümde, MATLAB kullanılarak tasarlanan üretici detaylı olarak anlatılmıştır.

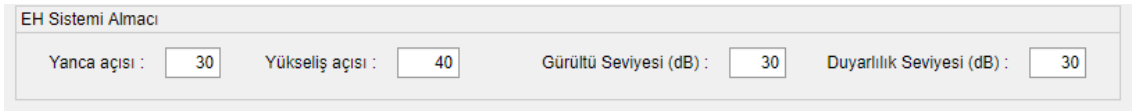
Üreticinin arayüz ekranı Şekil 3.1’ de gösterilmiştir. Bu arayüz üzerinden radar anteninin dönüş parametreleri ve EH sisteminin almaç özellikleri girilerek örnek tarama örüntüsü çizdirilebilmektedir. Tarama tiplerine göre istenilen alanlar doldurularak liste oluşturulur. Bu listede sıralanan tarama tipleri için girilen parametreler dahilinde yapay anten tarama örüntüleri oluşturulur. Oluşturulan tarama örüntüleri ve etiketleri veri tabanına kaydedilir.



Şekil 3.1 Anten tarama örüntüsü üretici.

### 3.1 EH Destek Sistemi Almaç Özellikleri

Bu kısım tasarlanan arayüz üzerinde Şekil 3.1.1’de gösterilmiştir. Bu kısımda EH sisteminin anteninin baktığı yanca ve yükseliş açısı derece olarak girilmiştir. EH sisteminin ve radar sisteminin simülasyon süresi boyunca sabit olduğu varsayılmaktadır. EH sisteminin almamacının gerçek ortamdan aldığı sinyallerde ortam koşullarından veya donanımsal nedenlerden dolayı gürültü olabilmektedir. Çalışmamızın gerçeğe yakın sonuçlar verebilmesi için bu durumu göz önüne almamız gerekmektedir. Bunun için sinyal seviyesine istenilen güçte Gauss gürültüsü eklenir. Gürültü seviyesi, 30dB sinyal gürültü oranını göstermektedir. EH sistemi almamacının belirli bir seviyeden düşük sinyalleri tespit edememe olasılığı vardır. Bu durumu göz önünde bulundurarak almamacın hassasiyet seviyesinin altındaki genliğe sahip sinyaller, gürültü olma ihtimalleri oldukça yüksek olduğundan, göz ardı edilecektir. Şekil 3.1.1’de duyarlılık seviyesi 30 dB seçilmiş ve maksimum genlik değerinden 30 dB değerinden fazla farkı olan darbeler göz ardı edilmiştir.



Şekil 3.1.1 EH sisteminin almaç özelliklerini giriş alanı.

### 3.2 Radar Anteni Dönüş Özellikleri

Anten dönüş parametrelerinin arayüze girildiği kısım Şekil 3.2.1’de gösterilmiştir. Bu kısımda ilk olarak anten tarama tipi seçilir. Daha sonra minimum veya maksimum periyod değerleri belirlenir. Antenin modeli ve huzme genişliği girilerek seçilen anten tarama tipi için anten pozisyonu belirleme işlemine geçilir.



Şekil 3.2.1 Radar anteni dönüş özelliklerini giriş alanı.

Simülâtörde beş farklı anten modeli bulunmaktadır. Bu modeller; Uniform, Triangular, Hamming, Poisson ve Blackman modelleridir. Bu modellere göre ana aydınlatmaların yanında çıkan yan aydınlatmaların PA seviyeleri farklılık gösterebilir. Kullanıcı ayrıca huzme genişliklerini yanca ve yükseliş açısı olarak girebilir. Simülâtör, belirli bir anten modelinde huzme genişliğini elde etmek için gereken eleman sayısını hesaplar. Bu hesaplamadan sonra simülâtör, yanca ve yükseliş için bir model oluşturur. EH sisteminin konumuna göre yanca ve yükselişte örüntü kazanımlarını hesaplar daha sonra yanca ve yükselişteki genel örüntüyü bulmak için bunları toplayarak EH sisteminin gözlemlendiği anten örüntüsünü oluşturur.

### 3.2.1 Radar Anteni Konumu

EH sisteminin radar sinyallerini tespit edebilmesi için radar antenin konumu çok önemlidir. EH sisteminin antenin baktığı alan radar antenin yanca ve yükselişte tarama açısı aralığına girmelidir. Şekil 3.2.1’de radar anteni konum parametreleri giriş ekranı gösterilmiştir. Anten tarama tiplerine göre istenilen parametreler doldurularak radar anteni dönüş parametreleri tamamlanmış olur.

### 3.3 Anten Tarama Tipi Listesi ve Veri Oluşturma

Şekil 3.1’de gösterilen arayüzde EH sistemi almaç özellikleri ve radar anteni dönüş parametreleri girildikten sonra ‘Listeye Ekle’ tuşu kullanılarak belirlenen özelliklerdeki radar anteni tipi Şekil 3.3.1’de gösterilen listeye eklenir. Yapay anten tarama örüntüsü oluşturmak istenen tiplerin tümü listeye eklendikten sonra veri uzunluğu ve her bir anten tarama tipi için oluşturulacak tarama örneği sayısı belirlenerek veri oluşturma işlemi tamamlanır.

Id	ATT	Min Periyod	Max Periyod	Model	MinYancaBW	MaxYancaBW	MinYükselişBW	MaxYüks
1	Helezonik	100	10000	Uniform	2	6	2	
2	Raster	100	10000	Uniform	2	6	2	
3	Tekli Sektörel	100	10000	Uniform	4	6	4	
4	Dairesel	100	10000	Uniform	2	4	2	
5	Çiftli Sektörel	100	10000	Uniform	2	6	2	
6	Konik	10	100	Uniform	2	6	2	
7	Spiral	10	100	Uniform	2	6	2	
8	Lob Anahtarlamalı	100	10000	Uniform	2	6	2	
9	Kesik Konik	10	100	Uniform	2	6	2	

Şekil 3.3.1 Veri oluşturulması için listelenmiş anten tarama tipi özellikleri.

Tablo 3.1 Üretilen Yapay Verilerin Özellikleri.

ATT	Periyod (s)		Model	Yan.BW		Yük. BW		Yan.Açı		Yük.Açı	
	Min	Maks		Min	Maks	Min	Maks	Baş	Bit	Baş	Bit
Helezonik	0.1	10	Uniform	2	6	2	6	0	360	30	50
Raster	0.1	10	Uniform	2	6	2	6	20	40	30	50
Tekli Sektör	0.1	10	Uniform	4	6	4	6	25	40	40	40
Dairesel	0.1	10	Uniform	2	6	2	6	0	360	40	40
Çiftli Sektör	0.1	10	Uniform	2	6	2	6	25	40	40	40
Konik	0.01	0.1	Uniform	2	6	2	6	30	30	40	40
Spiral	0.01	0.1	Uniform	2	6	2	6	30	30	40	40
Lob Anahtar.	0.1	10	Uniform	2	6	2	6	30	35	40	45
Kesik Konik	0.01	0.1	Uniform	2	6	2	6	30	30	40	40

Anten tarama tipi belirleme algoritmasında kullanılacak olan yapay veriler Tablo 3.1’de verilen özelliklere göre üretilmiştir. EH sisteminin anteninin yanca açısı 30, yükseliş açısı 40 derece olarak şekilde sabit tutulmuştur. Her bir sınıf için 10000 örnek üretilmiştir.

Belirlenen özelliklerde veri oluştururken verinin gerçeğe en yakın olması yapılan testlerin gerçek ortam testlerine yakın olmasını sağlar. Bu yüzden yabancı darbe eklemesi ve eksik darbe oluşması için darbe silme işlemi yapılır. EH sisteminin almacına ulaşan radar sinyali  $P_r(t)$  ise, bu eşitlik (3.1)’de gösterilmiştir.

$$P_r(t) = \frac{P_t(t) G_r(t) \lambda^2}{(4\pi R(t))^2} G_t[\theta(t), \phi(t)] \quad (3.1)$$

Formülde  $P_t(t)$ ,  $G_r(t)$ ,  $R(t)$ ,  $G_t[\theta(t), \phi(t)]$  sırasıyla radar sinyalinin çıkış gücü, alıcı anten kazancı, EH sistemi ile radar sistemi arasındaki uzaklık, radar anten kazancını göstermektedir. Bu formül (3.2)’de dB cinsinden gösterilmiştir.

$$P_{r,dB}(t) = P_{t,dB}(t) + G_{r,dB}(t) + G_{t,dB} \quad (3.2)$$



Radar Sisteminin gönderdiği gücün büyük bir kısmı ana huzmenin yönünde yoğunlaştığından,  $P_{r,dB}(t)$  en yüksek değerini EH sistemini aydınlatırken alır, ana huzmeden uzaklaştıkça  $P_{r,dB}(t)$  değeri düşmektedir [10].  $P_{r,dB}(t)$ , gerçek EH ortamında gürültü, eksik veya sahte darbeler tarafından bozulabilir. Bu bozulma daha çok genlik örüntüsünde olacağından alınan genliğin ( $P_{rb,dB}(t)$ ) modellenmesi gerekmektedir. Bu model (3.3)'te verilen eşitlikle ifade edilebilir [11].

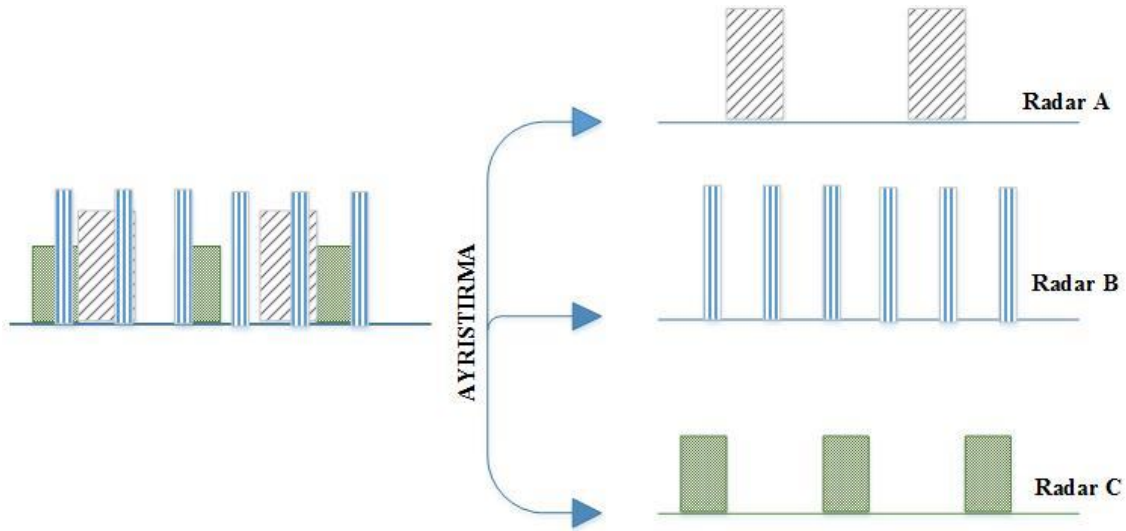
$$P_{rb,dB}(t) = (1 - X_k)(1 - X_s)P_{r,dB}(t) + N_{dB}(t) + (1 - X_k)X_sP_{s,dB}(t) \quad (3.3)$$

Burada  $X_k$  ve  $X_s$ , sırasıyla eksik ve sahte darbelerin olasılığını ifade eden Bernoulli dağılımlı rastgele değişkenlerdir.  $P_{s,dB}(t)$  ve  $N_{dB}(t)$ , sırasıyla sahte darbeyi ve gürültünün gücünü gösteren beyaz gauss gürültüsü (WGN) dağılımlı rastgele değişkenlerdir.

#### 4. ANTEN TARAMA TİPİ BELİRLEME ALGORİTMASI

EH sistemleri, ortamdaki radarlardan yayılan sinyalleri alarak darbe genliği (PA), radyo frekansı (RF), darbe genişliği (PW), varış zamanı (TOA) gibi sinyal parametrelerini ölçmektedir. Ölçülen RF, TOA, PW ve PA bilgileri grubuna Darbe Tanımlama Kelimesi (DTK) denir. Yakalanan sinyalin sinyal-gürültü oranı (SNR), yalnızca bir DTK'nin tespit edilip edilememesini etkiler, ancak tespitten sonraki işlemleri etkilemez.

EH ortamında birçok radar aynı anda çalışabilmektedir. Şekil 4.1'de gösterildiği gibi farklı radarlardan yayılan DTK'ler gösterilmiştir. EH sisteminin radarın kimliğini bulabilmesi için DTK'leri kullanarak kimliklendirme yapması gerekmektedir. Bu nedenle, önce farklı radarların darbe dizileri ayrılmalıdır. Bu süreç EH literatüründe "Deinterleaving" olarak adlandırılmaktadır [1]. Ayrıştırılmış DTK'lerin farklı bir radar olarak raporlanması ve her bir raporun sonraki aydınlatmalarda güncellenmesine kimliklendirme denir. Ayrıştırma ve kimliklendirme yöntemleri iki temel elektronik destek sistemi sürecini oluşturur, ancak bu algoritmalar bu çalışmada tartışılmayacaktır. Bu çalışmada, farklı radarlardan alınan sinyallerin birbirinden %100 başarıyla ayrıştırıldığı ve darbe dizileri olarak raporlandığı varsayılmıştır.



Şekil 4.1 Ayrıştırma işlemi (Deinterleaving).

Bölüm 2'de açıklanan anten tarama örüntüleri üzerinde inceleme yapıldığında ATT'lerin, ana huzme sayısı, ana huzmeler arasındaki genlik değişim ve ana huzmeler arasındaki zaman farkları açısından birbirinden ayrıldığı görülmektedir. Uzman personel tarafından

bu tiplerin ayrıştırılabilmesi, yapay zeka kullanılarak da ayrıştırılabileceği düşüncesini oluşturmuştur. İlk olarak yapay zeka tarafından algılanabilecek öznelikler belirlenmiş ve bu özneliklerden çıkarılacak ayırt edici bilgiler sayesinde ATT belirleme işlemi yapılmıştır. Her DTK için, darbenin genliği (PA) ve varış zamanı (TOA) temel öznelikler olarak kullanılır. Bu öznelikler kullanılarak biriktirilmiş DTK'lardan tarama örüntüsü öznelikleri çıkarılmıştır. Bölüm 3'de anlatılan anten tarama örüntüsü üretici ile oluşturulan bir anten tarama örüntüsü örneği darbe genliği (PA) ve darbe varış zamanı (TOA) bilgilerini içeren en az N adet DTK içermektedir. N değeri bu çalışma için 1000 olarak seçilmiştir.

Her anten tarama örneğinin eşit sayıda aydınlatma içerecek kadar biriktirilmesi gerekmektedir. Aydınlatma sayısı bu çalışma için 10 olarak belirlenmiştir. Her tarama örneği için en az 10 aydınlatma gözlemlenecek şekilde yapay darbe dizileri üretilmiştir. Anten tarama örneğinde bazı eksik ve sahte darbeler veya aydınlatmalar olabilir. İlk on aydınlatma seçilmeden önce darbeler ve aydınlatmalar temizlenmelidir. Bu temizleme süreci dört aşamadan oluşur;

1. Beş darbeden daha az darbeye sahip aydınlatmalar temizlenir.
2.  $M$  adet genlik değerine sahip  $X$  verisindeki  $i$ . genlik değeri  $X_i$  ise;

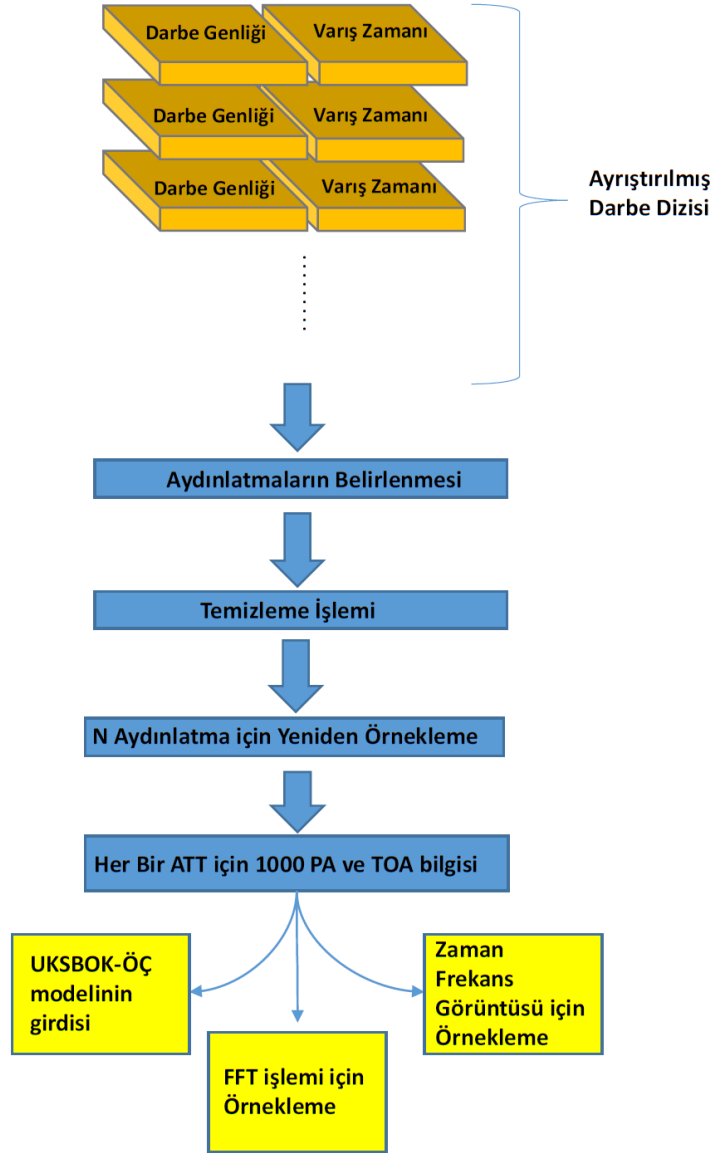
$$W = \begin{cases} 1, & M < 20 \\ 2, & M < 100 \\ 3, & M < 200 \\ 3 + \text{floor}\left(\frac{M}{200}\right), & M \geq 200 \end{cases} \quad \text{olduğu durumda}$$

$$\frac{(X_{i-1} + \dots + X_{i-W})}{W} - 3dB < X_i < \frac{(X_{i+1} + \dots + X_{i+W})}{W} + 3dB$$

,eşitsizliğini sağlamayan  $i$ . genlik değerleri temizlenir.

3. 1. Adım tekrar edilir.

Bu temizleme işlemlerinden sonra her tarama örüntüsü örneği için 10 aydınlatma seçilir. Üretilen her tarama örüntüsü örneğinin PRI değerleri farklı olabilir. Bu nedenle, aydınlatma sayısı sabit tutulduğunda, her bir tarama örneğinin sahip olduğu darbe sayısı farklılık gösterebilir. Öznelik çıkarımı için kullanılan yöntemlerde her tarama örneği için eşit sayıda darbe gerektiğinden, her bir tarama örneği için darbe sayısı yeniden örnekleme yapılarak eşitlenmiştir.

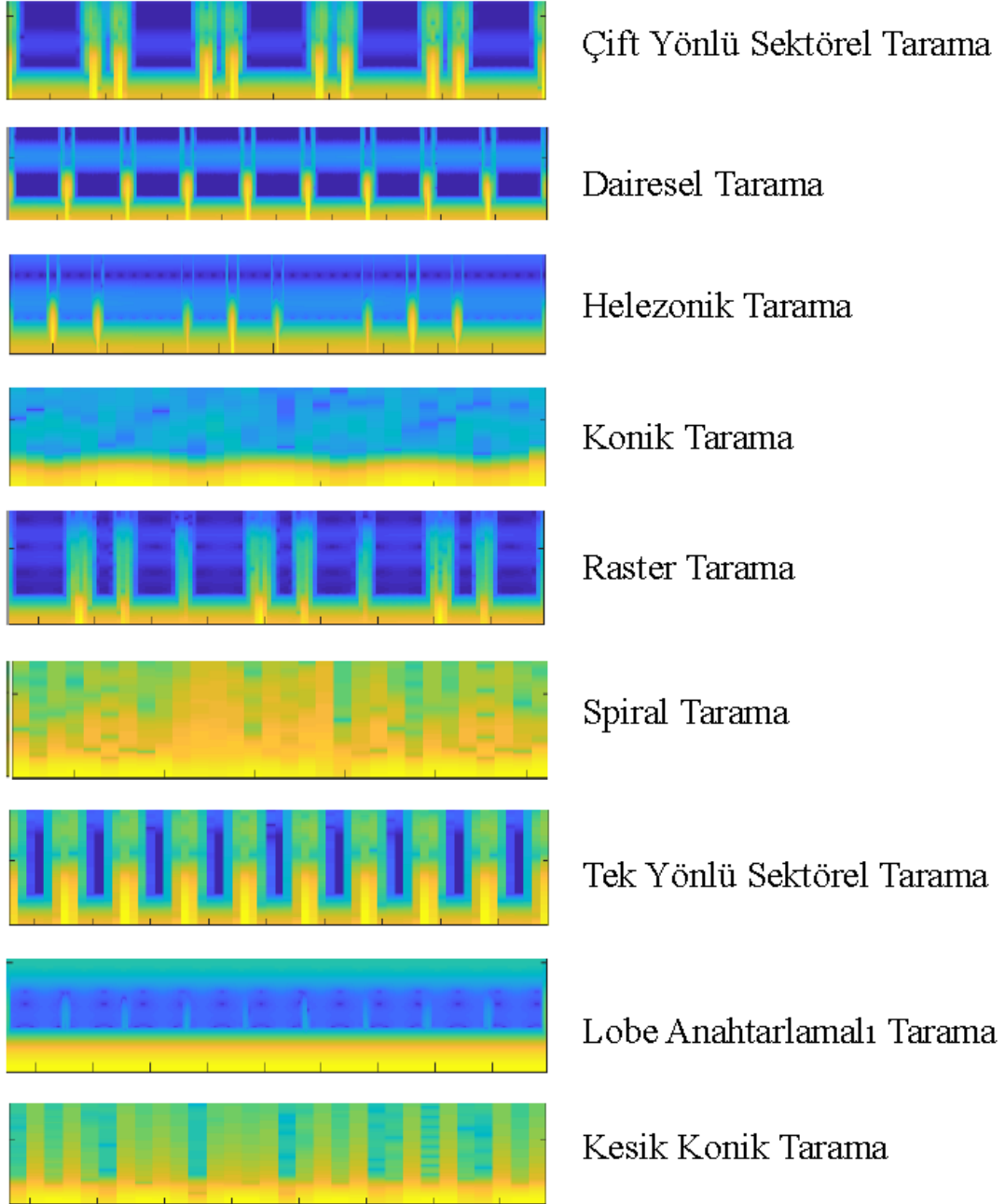


Şekil 4.2 Veri hazırlama işlemi.

Ön işleme adımları detaylı olarak Şekil 4.2’de gösterilmiştir. Veriler hazırlanırken darbe genlikleri (PA) önce dBm'den mV'ta dönüştürülür. Daha sonra toplanan en büyük darbe genliği değeri ile bütün darbeler normalize edilir. Benzer şekilde varış zamanı (TOA) değeri de sistem tarafından ölçülen darbelerin en büyük TOA değerine göre normalize edilir. Bölüm 4.1’de detaylı olarak anlatılan İstatistiksel Öznitelik Çıkarma (İÖÇ) ve Uzun-Kısa Süreli Bellek tabanlı Otomatik Kodlayıcılar (UKSBOK) ile öznitelik çıkarma yöntemleri normalize edilmiş PA ve TOA değerlerini kullanır.

Diğer öznitelik çıkarma yöntemi olan Evrişimsel Sinir Ağı tabanlı Otomatik Kodlayıcılar (ESAOK) ile öznitelik çıkarma yönteminin girdisi zaman-frekans görüntüsüdür (TFI).

TFI, ön işleme yöntemleri sonucunda oluşturulan 1000 adet Darbe Genliği (PA) ve Darbe Varış Zamanı (TOA) kullanılarak elde edilir. Şekil 4.3'te gösterilen her bir anten tarama tipinin zaman-frekans görüntüsünü elde etmek için Short Time Fourier Transform (STFT) kullanılmıştır.



Şekil 4.3 Anten tarama tiplerinin zaman-frekans görüntüsü.

STFT'de sinyal küçük çerçevelere bölünür ve bu çerçeve anlarında sinyalin durağan olduğu varsayılır. Bu çerçeveler, sinyale örtüşen pencere fonksiyonu uygulanarak elde

edilir. Bu teknik, Fourier Dönüşümünün lokalizasyonu fikrine dayanmaktadır ve dönüşüm işlemi, ilgilenilen yerde uygun bir pencere seçilerek gerçekleştirilir [12].

STFT yönteminde, zaman ekseninde kaydırılan pencere fonksiyonunun ( $g(t)$ ) mevcut pencere merkezi değeri  $\tau$  ise pencereli zaman serisi aşağıdaki denklem ile ifade edilebilir:

$$g(t) = f(t)g(t - \tau) \quad (4.1)$$

Fourier Dönüşümü, pencereli zaman serisinin bu ilgili zaman aralığında alınır.

$$STFT(\tau, f) = \int_{-\infty}^{+\infty} g(t) \exp(-j2\pi ft) dt \quad (4.2)$$

Daha sonra pencere yeni bir zaman değerine kaydırılır ve yine bu pencereli zaman serisinin Fourier Dönüşümü alınır. Bu süreç devam ederken, iki boyutlu bir zaman-frekans görüntüsü oluşturulur. Pencereyi zaman serisi üzerinde kaydırarak ve her pencereli zaman serisinin Fourier Dönüşümünü alarak bir zaman-frekans görüntüsü oluşturulur.

STFT yönteminde uygun pencerenin seçilmesi, sonucu etkileyen önemli bir faktördür. Örneğin, dikdörtgen bir pencere, düşük frekanslı bir çözünürlük sunar. Üçgen bir pencere, azalan bir frekans spektrumu verir ve dikdörtgenden daha iyi bir pencere olarak kabul edilir. Bir fonksiyonun STFT pencere işlevi olarak kullanılabilmesi için belirli enerjiye sahip ve entegre edilebilir olması gerekir. STFT algoritmalarında beş tür pencere kullanılabilmektedir. Bunlar dikdörtgen, hanning, hamming, gaussian, blackman pencereleridir. Bu çalışmada hamming penceresi seçilmiştir. Hamming penceresinin seçilmesinin sebebi bütün anten tarama tipleri için en iyi zaman-frekans görüntü çözünürlüğünü vermesidir. STFT yönteminde kısa süreli bir pencere seçilirse görüntü çözünürlüğü artar, frekans çözünürlüğü azalır. Uzun süreli bir pencere seçilirse tam tersi durum gözlenir. Zaman-frekans görüntülerinin oluşturulması için MATLAB spectrogram fonksiyonu kullanılmıştır. Çalışmada kullanılan DFT nokta sayısı (4.3)'te gösterilen eşitlikle belirlenmiştir.

$$nfft = \max\left(256, 2^{\left(\log_2\left(\frac{N}{4.5}\right)\right)}\right) \quad (4.3)$$

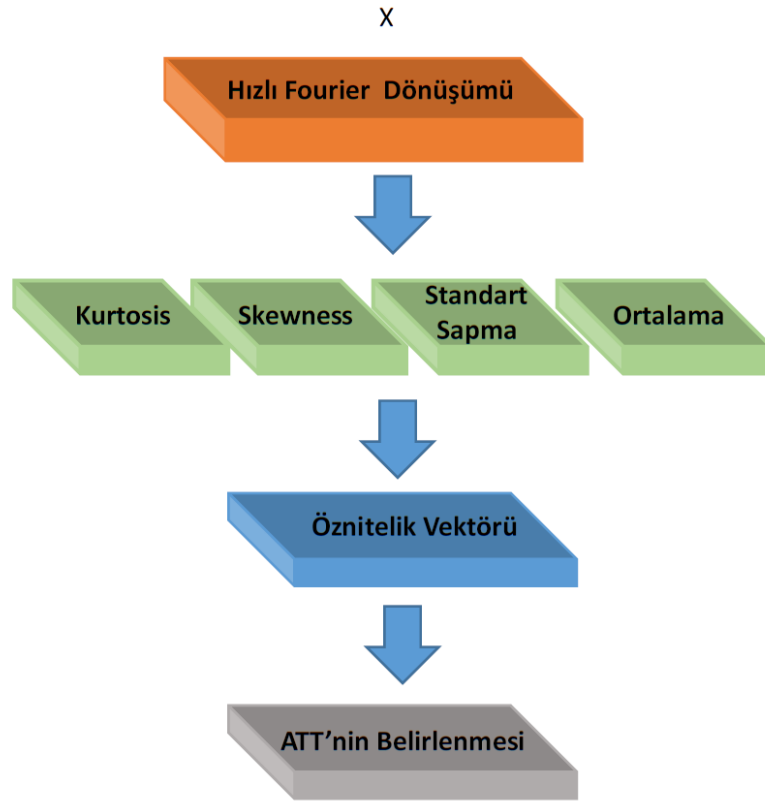
N girdi olan sinyalin örnek sayısını, nfft ise DFT nokta sayısını ifade eder.

## 4.1 Öznitelik Çıkarma Yöntemleri

Bu bölümde anten tarama tipi belirleme probleminde kullanılmak için tasarlanan öznitelik çıkarma yöntemleri detaylı olarak anlatılacaktır.

### 4.1.1 İstatistiksel Öznitelik Çıkarma

İstatistiksel yöntemler kullanılarak öznitelik çıkarma veri analizinde kullanılan en yaygın yöntemlerden biridir. İstatistiksel yöntemler kullanarak öznitelik çıkarılması üzerinde yapılan en son çalışmalardan biri [14]'tür. Şekil 4.1.1.1'de gösterilen X, Hızlı Fourier Dönüşümünün (FFT) girdisini temsil eder.



Şekil 4.1.1.1 İstatistiksel öznitelik çıkarma yöntemi.

FFT, bir dizinin Ayırık Fourier Dönüşümü (DFT) değerini hesaplayan bir algoritmadır. Veri üzerinde FFT uygulandıktan sonra öznitelik çıkarmak için istatistiksel yöntemler kullanılarak FFT uygulanmış veri analiz edilir. DFT, (4.4)'te formüle edilmiştir.

$$X(k) = \sum_{n=0}^{N-1} x_n \exp(-j2\pi kn/N) \quad (4.4)$$

Bu formülün gerçekleşmesi  $N^2$  işlemi gerektirir. Burada  $N$  adet  $X_k$  çıktısı vardır ve her çıktı bir toplam  $N$  terim gerektirir.

FFT ise  $N \log(N)$  işlemlerde aynı sonuçları hesaplamak için kullanılan bir yöntemdir [13]. FFT işleminin çıktısının ortalama (mean), standart sapma (standard deviation), çarpıklık (skewness) ve basıklık (kurtosis) değerleri hesaplanmıştır. Ortalama ve standart sapma sırasıyla (4.5) ve (4.6)'da tanımlanmıştır. (4.5)'te rastgele değişken  $A$  vektörü  $N$  skaler gözlemden oluşur.

$$\lambda = \sum_{i=1}^N \frac{A_i}{N} \quad (4.5)$$

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^N |A_i - \lambda|^2} \quad (4.6)$$

Bir dağılımın çarpıklık değerinin 3.dereceden kökü şu şekilde tanımlanır:

$$s = \sqrt[3]{\frac{E(x - \lambda)^3}{\sigma^3}} \quad (4.7)$$

Burada  $x$ 'in ortalaması  $\lambda$ ,  $x$ 'in standart sapması  $\sigma$  ve  $E(t)$ ,  $t$  vektörünün beklenen değerini (Expected Value) temsil eder. Benzer şekilde, bir dağılımın basıklık değerinin 4.dereceden kökü şu şekilde tanımlanır:

$$k = \sqrt[4]{\frac{E(x - \lambda)^4}{\sigma^4}} \quad (4.8)$$

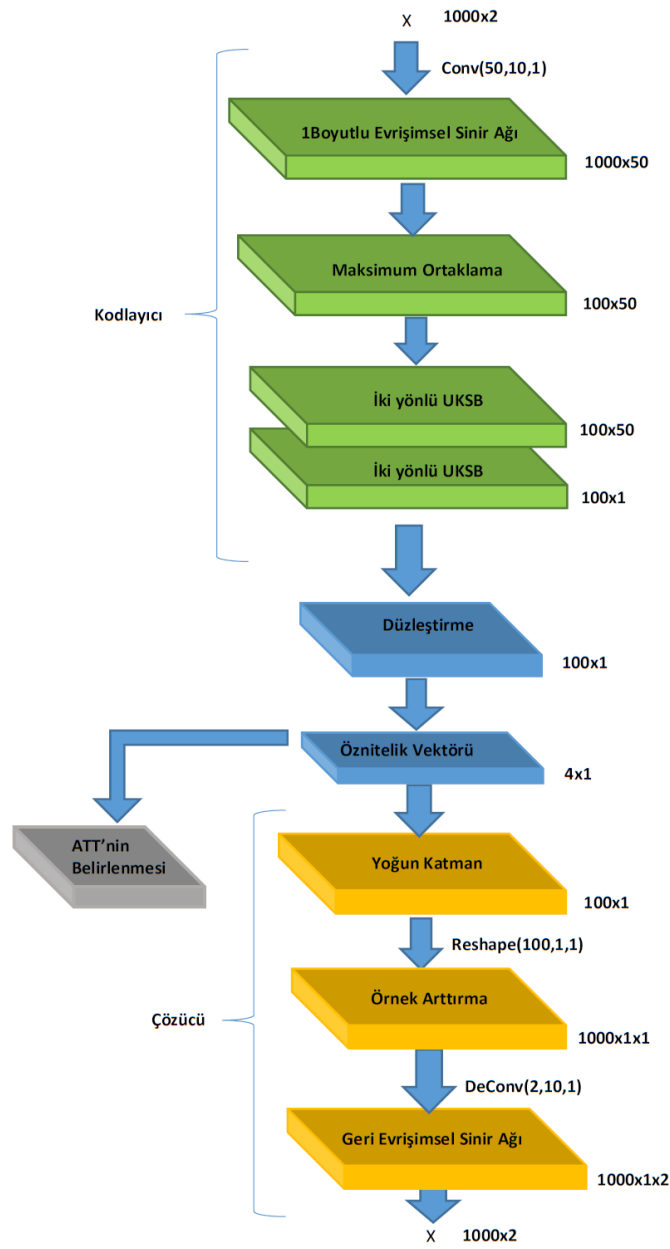
Bu işlemler sonucunda öznelik vektörü  $[\lambda; \sigma; s; k]$  her bir anten tarama örneği için oluşturulmuştur.

#### 4.1.2 Uzun-Kısa Süreli Bellek (UKSB) Tabanlı Otomatik Kodlayıcı ile Öznelik Çıkarma

Bu çalışmada kullanılan öznelik çıkarma yöntemlerinden biri olan UKSB tabanlı Otomatik Kodlayıcılar ile öznelik çıkarma (UKSBOK-ÖÇ) yöntemi Şekil 4.1.2.1'de verilmiştir. İlk katman 1 Boyutlu Evrişimsel Sinir Ağı katmanına ve ReLU aktivasyon



fonksiyonuna sahiptir. Sonraki katman olan, Maksimum Ortaklama katmanı bir pencere içine düşen her öznelik için maksimum değeri seçer. Maksimum Ortaklama pencere boyutu 10 x 1'dir. Sonraki iki katman, Çift Yönlü UKSB'lerden oluşur. Bu katmanlarla ağ, darbelerin oluşturduğu örüntüyü öğrenmeye çalışır. Şekil 4.1.2.1'de gösterilen X, girdiyi ve otomatik kodlayıcı çıktısını; Conv(filtre,kernel,stride), evrişim işlemi; DeConv(filtre,kernel,stride), geri evrişim işlemi temsil eder. Filtre, kernel ve stride sırasıyla uygulanan filtre sayısı, filtre boyutu, adım sıklığını belirtir. Ara katmandan elde edilen öznelik vektörü, gözetimli ve gözetimsiz öğrenme modellerinde kullanılır.



Şekil 4.1.2.1 UKSB tabanlı Otomatik Kodlayıcılarla Öznelik Çıkarılarak ATT'nin Belirlenmesi.

*1 Boyutlu Evrişimsel Sinir Ağı Katmanı:* Evrişimsel Sinir Ağları (ESA) bir, iki veya üç boyutlu olmalarına bakılmaksızın aynı şekilde çalışır. Aradaki fark, girdi matrisinin yapısı ve öznitelik algılayıcı olarak da bilinen filtrenin veriler arasında nasıl hareket ettiğidir[15]. Bir boyutlu evrişimsel sinir ağı katmanı, çıktı vektörü üretmek için tek bir uzaysal boyut üzerinden geçen tek boyutlu bir filtreden oluşmaktadır. Bir boyutlu ESA katmanları iki boyutlu ESA katmanlarından daha az parametre içereceğinden eğitilmesi daha kolaydır. 1 boyutlu ESA katmanları daha basit bir yapıya sahip olduğundan hesaplama maliyeti daha düşüktür. Bu durum hızlı çalışması gereken modellerde bir boyutlu ESA katmanını vazgeçilmez yapmaktadır.

*1 Boyutlu Maksimum Ortaklama Katmanı:* Maksimum ortaklama, bir pencere içerisinde kalan veri noktalarından maksimum değerli olan noktanın seçilmesi işlemidir. Amaç, girdi olan veri noktalarının bütünü temsil eden noktayı seçerek girdi verisinin boyutunu azaltmaktır. Bu yöntem girdi verisinin özellikleri üzerinde tahmin yapılmasına izin vermektedir.

*İki Yönlü UKSB Katmanı:* UKSB ilk olarak Hochreiter ve Schmidhuber [16] tarafından geliştirilmiştir. Bu katmanın girdi vektörü  $x = (x_1, x_2, \dots, x_n)$ 'dir. Burada  $n$ , girdi vektörünün uzunluğudur. UKSB'in merkez yapısında üç kontrol kapısı bulunur ve bu kontrol kapıları bellek hücresinin aktivasyon vektörünü ( $c$ ) kontrol etmek için kullanılırlar. Bu kapılardan biri olan unutmama kapısı, mevcut zaman hücre durumundan ( $c_t$ ) ne kadar önceki hücre durumuna ( $c_{t-1}$ ) kadar olan bilgilerin unutulmayacağını belirler. Girdi kapısı mevcut zaman hücre durumundaki ( $c_t$ ) ağın girdilerinden ( $x_t$ ) kaydedilenlerin kapsamını belirler. Çıktı kapısı, mevcut zaman hücre durumundaki ( $c_t$ ) UKSB ağının çıktı ( $h_t$ ) olarak gönderdiği veri sayısını tanımlar. Bu üç kapı da yoğun katmandan oluşur ve girdileri vektör, çıktıları  $[0,1]$  arasında reel sayıdır.

$$\text{girdi kapısı ; } i_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i)$$

$$\text{unutma kapısı ; } f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f)$$

$$\text{çıktı kapısı ; } o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o) \quad (4.9)$$

$$\text{hücre durumu ; } c_t = f_t * c_{t-1} + i_t * \tanh(W_{cx}x_t + W_{ch}h_{t-1} + b_c)$$

$$\text{hücre çıktısı ; } h_t = o_t * \tanh(c_t)$$

Burada;  $\sigma$ , lojistik sigmoid işlevini,  $x_t$ , bir vektörün  $t$ . elemanını ve  $h_t$ , yoğun katmanı belirtir.  $W$  terimleri ve  $b$  terimleri, üç kapı için sırasıyla ağırlık matrislerini ( $W_{fx}$  unutmaya kapısı ağırlık matrisi) ve eşik vektörlerini ( $b_i$  girdi kapısı eşik vektörü) temsil eder.

Schuster ve Paliwal [17], aynı çıktıyı veren ve zıt yönlerde çalışan iki ayrı UKSB ağını birleştirerek iki yönlü UKSB yapısını geliştirmişlerdir. Bu yapı ile çıktı katmanı, hem geçmiş hem de gelecek hakkındaki bilgileri kullanabilir.

Bir İki Yönlü UKSB,  $x = (x_1, x_2, \dots, x_n)$  girdi dizisini, ters yönde ileriye doğru gizli dizi  $h_t = (h_1, h_2, \dots, h_n)$  ve geriye doğru gizli dizi  $h'_t = (h'_1, h'_2, \dots, h'_n)$  'ye göre hesaplar. Kodlanmış vektör  $y_t$ , son ileri ve geri çıktılarının birleştirilmesiyle oluşturulur,  $y_t = [h_t, h'_t]$ .

$$\begin{aligned} h_t &= \sigma(W_{hx}x_t + W_{hh}h_{t-1} + b_h) \\ h'_t &= \sigma(W_{h'x}x_t + W_{h'h'}h'_{t+1} + b_{h'}) \\ y_t &= W_{yh}h_t + W_{yh'}h'_t + b_y \end{aligned} \quad (4.10)$$

(4.10)'da  $y = (y_1, y_2, \dots, y_t, \dots, y_n)$ , iki yönlü UKSB katmanının çıktısıdır.

*Düzleştirme Katmanı:* UKSB katmanı ile yoğun katman arasında bir 'Düzleştirme' katmanı vardır. Düzleştirme işlemi UKSB katmanının çıktısının 4 hücreye sahip öznitelik vektörü katmanını besleyebilen şekle dönüştürür.

*Yoğun Katman:* Bu katman öznitelik vektörünün oluşturulduğu katmandır. Otomatik kodlayıcıda kodlayıcı ve çözücü işlevi gören katmanların arasında bulunur. Örnekleri birbirinden ayırt eden bilgiler bu katmanda bulunur. Bu katmanın boyutu öznitelik vektörünün boyutunu belirlemektedir.

*Örnek Arttırma:* Maksimum ortaklama katmanının tersi örnek arttırmadır. Örnek arttırma yapılabilmesi için maksimum ortaklama çıktısı olan veri noktasının, vektör ya da matris üzerindeki pozisyonu hafızada saklanmalıdır [18]. Örnek arttırma işleminin girdisi olan veri noktası hafızada saklanan pozisyon değerine göre matris yada vektör üzerine yerleştirilir. Maksimum örnekleme katmanının girdi boyutu ile örnek arttırma katmanını çıktı boyu eşit olması gerektiğinden vektör yada matris üzerine yerleştirdiğimiz veri noktasının komşu noktaları sıfır ile doldurulur. Geri evrimsel katmanların aksine, örnek arttırma katmanları parametre sayısını arttırmaz. Sadece kullanılan hafızayı arttırır. Ancak yeni oluşturulan matris ya da vektörde pozisyonu tutulan veri noktası dışındaki alanlar sıfır ile doldurulduğundan yeniden oluşturulan matris ya da vektörün taşıdığı bilgi

çok azdır. Bu nedenle, örnek arttırma katmanları genellikle parametre sayısını arttıran geri evrişimsel sinir ağı katmanlarıyla beraber kullanılırlar.

*Geri Evrişimsel Sinir Ağı Katmanı:* Geri Evrişimsel yöntemi ilk olarak Long ve arkadaşları tarafından evrişimsel sinir ağı katmanlarında kullanılmıştır [19]. Burada evrişimsel bir katmanın çıktısına geri evrişimsel katman uygulanarak evrişimsel katmanın girdisi elde edilmeye çalışılır. Bunun için, tek yönde  $1/f$  'e eşit adımla geri evrişimsel filtresi uygulanır.  $f$  değeri evrişimsel katmanın fitre adımını (stride) ifade eder. Geri evrişimsel filtrelerinin eğitilmesi için ağı daha derin hale getirilmesi gerekir. Bu hem bir avantaj hem de bir dezavantajdır çünkü ağı derinleştirilmesi, daha fazla verinin iyi eğitileceğini gösterirken daha fazla zaman ve işlem harcanacağından maliyeti arttırmaktadır.

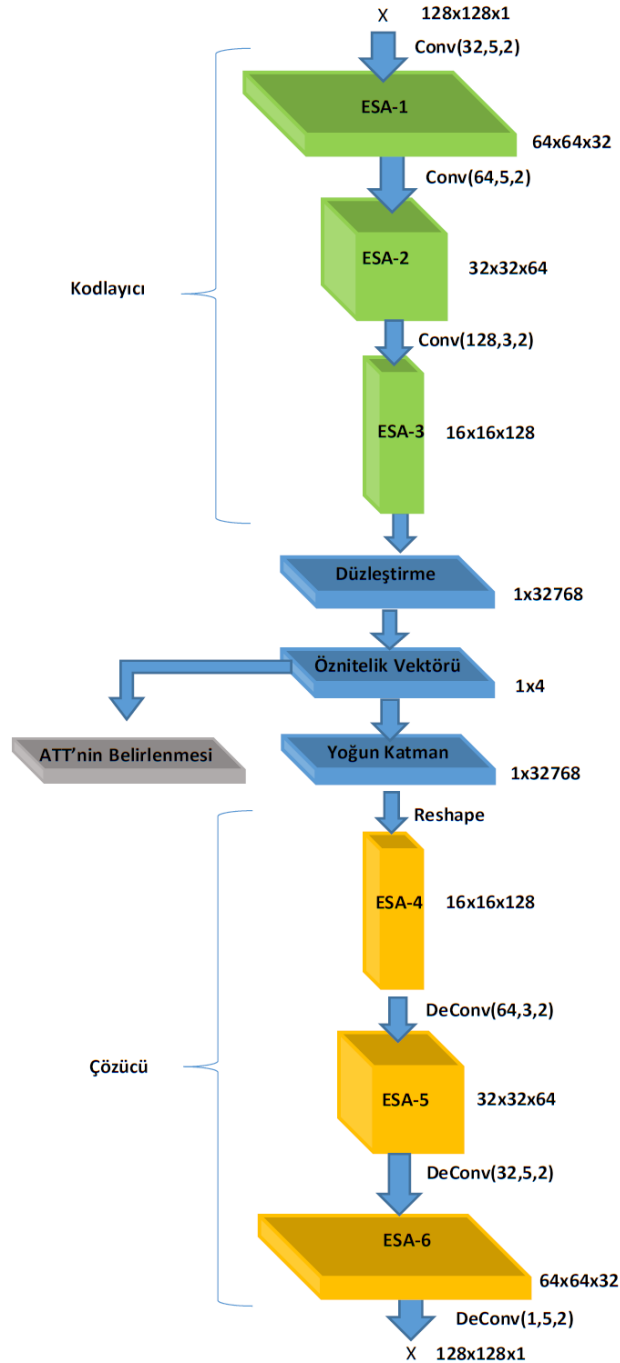
#### **4.1.3 Evrişimsel Sinir Ağı (ESA) Tabanlı Otomatik Kodlayıcı ile Öznitelik Çıkarma**

Otomatik kodlayıcılar, veri noktaları arasındaki karmaşık doğrusal olmayan ilişkileri öğrenmek için tasarlanmış sinir ağı modelleridir. Genellikle, otomatik kodlayıcılar birden fazla sinir ağı katmanından oluşur ve çıktı olarak girdiyi yeniden yapılandırmak için eğitilir [20].

Otomatik kodlayıcının girdinin düşük boyutlu bir temsilini öğrenmesini sağlamanın basit bir yolu, yoğun katmandaki hücre sayısını sınırlamaktır. Otomatik kodlayıcının artık sınırlı sayıda hücre kullanarak girişi yeniden yapılandırması gerektiğinden, girişin en önemli yönlerini öğrenecek ve verilerdeki küçük değişiklikleri görmezden gelecektir.

ESA'lar birçok bilgisayarla görme görevinde iyi performans gösterdiğinden dolayı otomatik kodlayıcıların evrişimsel katmanları kullanması doğaldır [21]. Yoğun katman, genellikle otomatik kodlayıcılarda düzleştirme katmanından sonra gelir. Şekil 4.1.3.1'de gösterilen X, zaman-frekans görüntüsü olan girdi ve otomatik kodlayıcı çıktısını; Conv(filtre,kernel,stride), evrişim işlemini; DeConv(filtre,kernel,stride), geri evrişim işlemini temsil eder. Filtre, kernel ve stride sırasıyla uygulanan filtre sayısı, filtre boyutu, adım sıklığını belirtir. Stride değeri bu yöntem için 2 seçilmiştir. Bunun nedeni, evrişimsel katmanların çıktısının boyutunu azaltarak özet bilgiye ulaşmayı kolaylaştırmaktır. Evrişimsel otomatik kodlayıcının çıktısı ile girdisi aynı boyutta olması gerektiğinden, kod çözücü katmanların yeniden boyutlandırılması gerekir.

Bu işlem eğitilebilir yukarı örnekleme işlemi kullanılarak yapılır. Bunun için geri evrişimsel sinir ağı katmanları kullanılır. Şekil 4.1.3.1’de ESA-1, ESA-2 ve ESA-3 evrişimsel katmanların çıktısı iken ESA-4, ESA-5 ve ESA-6 ise geri evrişimsel katmanların çıktısıdır.



Şekil 4.1.3.1 ESA tabanlı Otomatik Kodlayıcılarla Öznitelik Çıkarılarak ATT'nin Belirlenmesi.

Evrişimsel otomatik kodlayıcılar genellikle görüntü sıkıştırma ve gürültü azaltma için kullanılır. Gürültü azaltma durumunda, ağ gürültü giderici otomatik kodlayıcı olarak adlandırılır ve standart otomatik kodlayıcıdan farklı şekilde eğitilir. Girdi sinyali uygun bir gürültü sinyali tarafından bozulur ve otomatik kodlayıcının gürültü sinyalini öğrenmesi sağlanır [22]. Ara katmanlar genellikle özet bilgi içerdiğinden evrişimsel otomatik kodlayıcılar görüntü arama uygulamalarında da kullanılabilir. Bu nedenle, ara katmanlar üzerinde benzerlik araması, ham görüntü piksellerinde benzerlik aramasından daha iyi sonuçlar vermektedir. Otomatik kodlayıcılar sayesinde arama işlemi daha dar bir alanda yapıldığından arama hızı önemli ölçüde artmaktadır. Bu çalışmada, otomatik kodlayıcının öznitelik vektörü katmanındaki bilgiler kullanılarak gözetimli ve gözetimsiz öğrenme yöntemleriyle ATT belirlenmiştir.

## **4.2 Gözetimsiz Öğrenme Yöntemi ile Anten Tarama Tipi Belirleme**

Bu bölümde anten tarama tipi belirleme algoritmasında kullanılan kümeleme yöntemleri ve bu yöntemlerde kullanılan farklı mesafe ölçütleri detaylı olarak anlatılacaktır.

### **4.2.1 K-Ortalamlar Kümeleme Algoritması**

K-Ortalamlar algoritması, gözetimsiz öğrenme ve kümeleme algoritmasıdır. Stuart Lloyd [23] tarafından geliştirilen algoritma, kümelerin dağılımı hakkında herhangi varsayımda bulunmaz. Bununla birlikte algoritma, küme sayısının sabit olduğu veya önceden bilindiği varsayımı altında çalışır ve maliyeti en aza indirmeye çalışacak şekilde örnekleri kümelere atar. K-Ortalamlar algoritmasındaki K, algoritmanın sonucunda ulaşılabilecek küme sayısını belirler ve algoritma K değerini girdi olarak alır. Algoritmaya girdi olarak verilen K değeri kullanılarak rastgele K tane merkez noktası seçilir. Her bir veri noktası ile rastgele seçilen merkez noktaları arasındaki mesafeler hesaplanır ve veri noktası en yakın merkez noktasına ait kümeyle atanır. Daha sonra bütün kümeler için merkez noktaları tekrardan hesaplanır ve güncel merkez noktaları için kümeleme tekrar edilir.

Merkez noktalarında değişim olmayana kadar bu işlemler devam eder. Algoritmanın adımları şu şekildedir;

1. Rastgele küme merkezleri seçilir.

2. Her veri örneğinin küme merkezlerine olan mesafesi hesaplanır.
3. Her veri örneği en yakın kümeye atanır.
4. Küme elemanlarının değerlerinin ortalaması alınarak küme merkezleri yeniden hesaplanır.
5. Küme merkezlerinde herhangi bir değişiklik olmayana kadar 2, 3 ve 4 numaralı adımlar tekrarlanır.

Lloyd algoritmasının etkinliği üzerinde birçok çalışma yapılmıştır. Bunlardan en önemlisi de Ostrovsky ve arkadaşlarının [24] yaptığı çalışmadır. Ostrovsky ve arkadaşları Lloyd algoritmasına benzer algoritmaların K-Ortalamlar algoritması üzerindeki etkilerini incelemiştir.

#### **4.2.2 K-Ortalamlar++ Kümeleme Algoritması**

Günümüzde kullanılan veri kümelerinin boyutu oldukça büyüktür. Buna rağmen K-Ortalamlar algoritması, günümüzde kullanılan gözetimsiz öğrenme yöntemler içinden en çok tercih edilen yöntemlerden biridir. Basit tekrarlayan yapısından dolayı büyük veri tabanları üzerinde yapılacak işlemler çok hızlı yapılabilmektedir. İlk olarak küme merkezi belirlenir ve her bir nokta için hangi küme merkezinin bu noktaya en yakın olduğuna karar verilir. İşlem yapılan veri noktası kendisine en yakın küme merkezine sahip olan kümeye atanır. Daha sonra en iyi küme merkezin hesaplanması, küme içindeki veri noktalarının ortalaması alınarak yapılabilir. Arthur ve Vassilvitskii [25], hem algoritmanın başarı oranını arttıran hem de iyi bir başlangıç küme merkezi noktası sayesinde Lloyd'un [23] algoritmasının çalışma süresini iyileştiren bir yöntem önermişlerdir. Bu yöntem K-Ortalamlar++ olarak adlandırılmıştır. Birçok kümeleme algoritması kullanılarak Bahmani ve arkadaşları [26] tarafından yapılan bir incelemede, K-Ortalamlar++ algoritmasının K-Ortalamlar algoritmasındaki ilk küme-merkezlerini tanımlama yönteminin geliştirilmesiyle bazı problemlerin başarılı bir şekilde üstesinden gelindiği söylenmiştir. Lee ve arkadaşları [27] fotoğraflara eklenen enlem ve boylam bilgilerini kullanarak coğrafi fotoğraf kümeleri oluşturmak için K-Ortalamlar++ algoritmasını kullanmışlardır. K-Ortalamlar++ algoritmasında bütün küme merkezlerinin belirlenebilmesi için veri üzerinden küme sayısı kadar geçilmesi gerekir. Bu yüzden, büyük veri tabanlarında algoritma çok yavaş davranabilir.

K-Ortalamlar algoritması gibi, K-Ortalamlar++ algoritmasının da uygulanması basittir. K-Ortalamlar++ algoritması, küme sayısının K olduğu varsayılırsa küme merkezlerini aşağıdaki gibi seçer.

1. X veri kümesinden rastgele bir veri noktası seçilir. Seçilen veri noktası birinci küme merkezidir ve  $c_1$  ile gösterilir.
2. Her veri noktası ile  $c_1$  arasındaki mesafeler hesaplanır.  $c_j$  ile veri noktası  $m$  arasındaki mesafe  $d(x_m, c_j)$  olarak belirtilir.
3. (4.11)'de hesaplanan olasılık değerinin maksimum olduğu  $x_m$  veri noktası ikinci küme merkezi  $c_2$  olarak seçilir.

$$\frac{d^2(x_m, c_1)}{\sum_{j=1}^n d^2(x_j, c_1)} \quad (4.11)$$

4.  $m = 1 \dots n$ ,  $p = 1 \dots j - 1$  değerlerinde j.küme merkezini seçmek için (4.12)'de hesaplanan olasılık değerinin maksimum olduğu  $x_m$  veri noktası j. küme merkezi  $c_j$  olarak seçilir.

$$\frac{d^2(x_m, c_p)}{\sum_{h=1}^n d^2(x_h, c_p)} \quad (4.12)$$

5. K adet kümenin merkez noktası bulunana kadar 4. adım tekrarlanır.

Arthur ve Vassilvitskii [25], birkaç küme dağılımı için bir benzetim çalışması yaparak noktadan küme merkezine kareler toplamı uzaklığı için K-Ortalamlar++ algoritmasının Lloyd'un [23] algoritmasından daha hızlı yakınsama sağladığını göstermiştir.

### 4.2.3 Hiyerarşik Kümeleme Algoritması

Hiyerarşik kümeleme yöntemleri özellikle az sayıda elemana sahip problemler için uygundur. Hiyerarşik kümeleme yöntemi, farklı kademelerdeki kümeleri birleştirerek yeni kümeler oluşturan ve bu kümelerin birbirleri arasındaki mesafeyi hesaplayan teknik analizdir. Ağaç diyagramlarıyla temsil edilen hiyerarşik kümeleme tekniklerinin sonuçlarına dendrogram denir [28]. Verilerin her biri, ilk aşamada bir küme olarak kabul edilir. Mesafe ölçütü kullanılarak hesaplanan mesafeye göre en yakın iki küme birleştirilir. Bu işlem kümelenecek veri kalmayana kadar devam eder. Hiyerarşik



kümeleme algoritmasında kümeler arasındaki benzerlikler ve yakınlıklar farklı yöntemlerle belirlenebilir [29]. Bunlar; tam bağlantı kümeleme, tek bağlantı kümeleme, ortalama bağlantı kümeleme ve Ward'ın minimum varyans yöntemidir. Tam bağlantı kümelemede birleştirilmeye aday küme çifti arasındaki yakınlık değeri, küme çiftlerinin elemanları arasındaki maksimum mesafe değeridir. Tek bağlantı kümelemede birleştirilmeye aday küme çifti arasındaki yakınlık değeri, küme çiftlerinin elemanları arasındaki minimum mesafe değeridir. Ortalama bağlantı kümelemede, küme çiftlerinin farklarını hesaplar ve bu farkların ortalamasını iki küme arasındaki mesafe olarak kabul eder. Ward yönteminde her küme için varyans değeri hesaplanır ve varyans değeri minimize edilerek her adımda en az varyans değerine sahip iki küme birleştirilir. Bu çalışmada Ortalama bağlantı yöntemi kullanılmıştır. Algoritmanın işleyişi aşağıda listelenmiştir.

1. Algoritma  $n$  örnek ve  $n$  küme ile başlar.
2. Kümeler arasındaki mesafe hesaplanır.
3. Birbirine en yakın iki küme belirlenir ve bu kümeler birleştirilir.
4. Toplam küme sayısı güncellenir.
5. 2, 3 ve 4'üncü adımlar,  $n-1$  kez tekrarlanır.

Bu algoritma işlemleri sonucunda, bütün veriler tek bir kümeye kadar atanır. Daha sonra bu kümeden en farklı iki alt küme seçilir ve bunların arasındaki mesafe ölçülerek bir ilişki kurulur. Bu işlemler belirlenen küme sayısına ulaşıncaya kadar devam edilir. Kümeler arasındaki ilişki dendrogram denilen ağaç diyagramlarıyla gösterilebilir.

#### 4.2.4 Mesafe Ölçütleri

Mesafe ölçütleri, kümeleme algoritmalarının performansını etkileyen en kritik faktördür [30]. Bu bölümde farklı mesafe ölçüleri açıklanmıştır.

*Öklid:* Öklid ölçütü, makine öğrenmesi algoritmalarında en çok kullanılan mesafe ölçütüdür. Öklid mesafesi iki nokta arasındaki doğrusal mesafeyi hesaplamak için kullanılır.  $a$  ve  $b$  noktaları arasındaki Öklid mesafesi, (4.13)'e göre hesaplanır.  $k$  değeri  $a$  ve  $b$  noktalarının boyut sayısını gösterir.

$$d(a, b) = \sqrt{\sum_{j=1}^k (a_j - b_j)^2} \quad (4.13)$$

*Kosinüs:* Kosinüs ölçütü, iki vektör arasındaki benzerliği ölçen bir ölçüttür. Kosinüs ölçütü, vektörler arasındaki açının kosinüsüdür. Bu ölçüt genellikle vektörlerin büyüklüğü önemli olmadığı durumlarda mesafeyi ölçmek için kullanılır. Örneklerin bazı özelliklerinin farklı bir anlam ifade etmeden ağırlıkların daha büyük olabilmesini sağladığı durumlarda Kosinüs ölçütünü uygulamak benzerlik oranının doğru hesaplanmasını sağlar.

$$d(a, b) = \frac{\sum_{j=1}^k a_j b_j}{\sqrt{\sum_{j=1}^k a_j^2 \sum_{j=1}^k b_j^2}} \quad (4.14)$$

*Manhattan:* Manhattan metriği,  $a$  ve  $b$  noktalarının her boyut için mutlak farkların toplamıyla hesaplanır. City block ölçütü olarak adlandırılmaktadır. Genellikle regresyon analizi, sıkıştırılmış algılama ve frekans dağılımı hesaplamalarında kullanılmaktadır.

$$d(a, b) = \sum_{j=1}^k |a_j - b_j| \quad (4.15)$$

*Bray-Curtis:* Bray-Curtis ölçütü, farklı bölgelerden toplanan veriler için örnekler arasındaki benzerliği ölçmenin en yaygın yoludur [31]. Ekoloji ve biyolojide, iki bölgede bulunan türler açısından bu iki bölgenin benzerliğinin ölçmek için kullanılır. Bray-Curtis ölçütü, iki bölgenin eşit büyüklükte olduğunu varsayar. Çünkü eğer bir alan diğerinden daha büyükse, kaplanacak çok daha fazla alan olduğu için doğal olarak daha büyük alanda daha küçük alana kıyasla daha fazla tür sayılacaktır.

$$d(a, b) = \frac{\sum_{j=1}^k |a_j - b_j|}{\sum_{j=1}^k a_j + b_j} \quad (4.16)$$

*Canberra:* Canberra mesafe ölçütü, iki veri kümesinin ne kadar benzer veya farklı olduğunu göstermek için kullanılan bir ölçüttür. Canberra mesafesi, Manhattan mesafesinin ağırlıklandırılmış bir versiyonudur. Sıfıra yakın değerler arasındaki ölçümlerde çok hassas olduğu için genellikle bir orijin etrafına dağılmış veriler için kullanılır.

$$d(a, b) = \sum_{j=1}^k \frac{|a_j - b_j|}{|a_j| + |b_j|} \quad (4.17)$$

*Minkowski*: Genelleştirilmiş mesafe ölçütü olarak da bilinir [32]. (4.18)'de  $p$  pozitif bir reel sayıdır.  $p = 2$  olduğunda Öklid mesafesi,  $p = 1$  olduğunda Manhattan mesafesi,  $p = \infty$  olduğunda ise Chebyshev mesafesi olarak tanımlanır.

$$d(a, b) = \left( \sum_{j=1}^k |a_j - b_j|^p \right)^{1/p} \quad (4.18)$$

*Chebyshev*: Chebyshev mesafesi, Minkowski mesafe ölçütündeki  $p = \infty$  olduğu özel bir durumdur ve iki nokta arasındaki mutlak farkların maksimumu olarak tanımlanır.

$$d(a, b) = \max(|a_j - b_j|) \quad (4.19)$$

*Mahalanobis*: Mahalanobis mesafesi, bir  $a$  noktası ile ortalaması  $b$  olan bir dağılım arasındaki mesafenin ölçüsüdür.  $a = [a_1, a_2 \dots a_k]$  noktasının, merkezi  $b = [b_1, b_2 \dots b_k]$  olan bir dağılımdan ne kadar uzakta olduğu (4.20) kullanılarak hesaplanabilir. Burada  $S$ , merkezi  $b$  olan dağılımın kovaryans matrisidir.

$$d(a, b) = \sqrt{(a - b)^T S (a - b)} \quad (4.20)$$

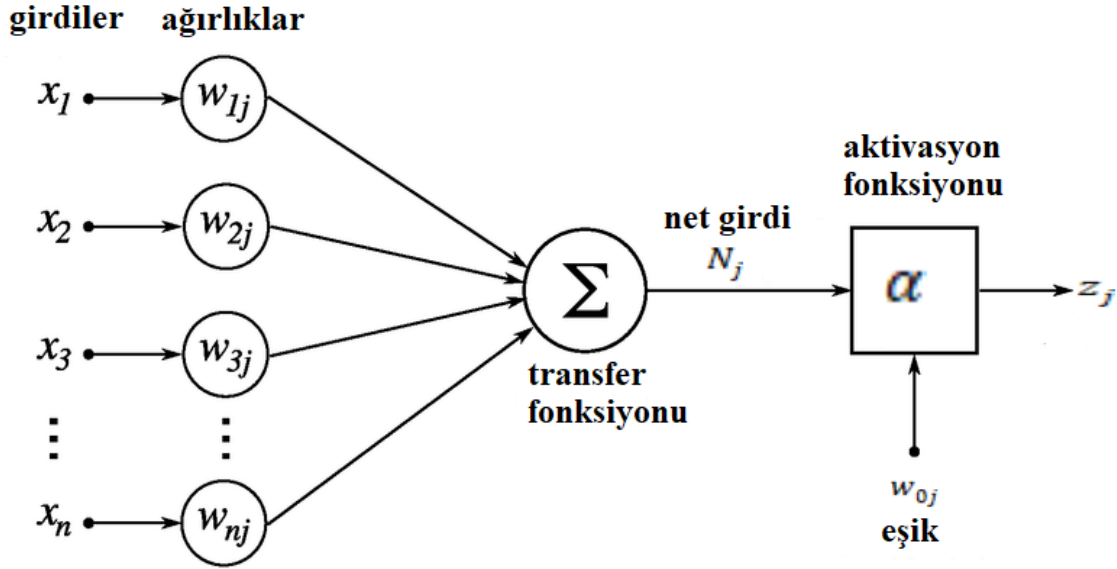
### 4.3 Gözetimli Öğrenme Yöntemi ile Anten Tarama Tipi Belirleme

Bu bölümde anten tarama tipi belirleme algoritması için kullanılan gözetimli öğrenme yöntemleri detaylı olarak anlatılmıştır.

#### 4.3.1 Yapay Sinir Ağı (YSA) Tabanlı Sınıflandırma

Yapay sinir ağları, makine öğrenimi uygulamalarının ve derin öğrenme algoritmalarının temelini oluşturmaktadır. Yapay sinir ağları (YSA), girdi katmanı, yoğun katman ve çıktı katmanından oluşur [20]. Bu katmanlar yapay sinir hücrelerinden oluşmaktadır. Yapay nöron olarak da adlandırılan bu hücreler diğer katmandaki hücelere bağlanır. Bu hücreler arasında ağırlık ve eşik değerleri vardır. Herhangi bir hücrenin çıktısı belirtilen eşik

değerinin üzerindeyse o hücre etkinleştirilir ve ağına bir sonraki katmanına veri gönderilir. Aksi durumda, ağıdaki sonraki katmana veri gönderilmez.



Şekil 4.3.1.1 Yapay Sinir Ağının Yapısı.

Yapay sinir ağlarında girdi katmanı belirlendikten sonra ağırlıklar atanır. Bu ağırlık değerleri, belirtilen hücrenin önemini gösterir ve büyük ağırlık değerine sahip olan hücreler diğer hücrelere göre çıktıyı daha fazla etkiler. Yoğun katmanın sahip olduğu sinir hücresinin yapısı Şekil 4.3.1.1’de gösterilmiştir. Tüm girdi değerleri ilgili ağırlıklarla çarpılır ve ardından toplanır. Bu işlemler (4.21)’de verilmiştir.

$$N_j = \sum_{n=1}^d w_{nj}x_n \quad (4.21)$$

$x_n, n = 0, 1, \dots, d$  girdileri ve  $N_j, j = 1, \dots, H$  çıktıları gösterir.  $w_{nj}, x_n$  girişinden  $N_j$  çıkışına olan bağlantının ağırlığıdır.

Daha sonra  $N_j$ , katmanın çıktısını belirleyen aktivasyon fonksiyonundan geçirilir. Bu çıktı belirli bir eşiği aşarsa, hücreyi etkinleştirir ve verileri ağıdaki bir sonraki katmana iletir. Bu durum, bir hücrenin çıktısının bir sonraki hücrenin girdisi olmasına neden olur. Bu işlemler (4.22) ve (4.23)’de verilmiştir.

$$z_j = \alpha(N_j, w_{0j}) \quad (4.22)$$

$\alpha$ , aktivasyon fonksiyonunu göstermektedir. Bu fonksiyon modelin tasarımına göre sigmoid, ReLU, vb. fonksiyonlar olabilir.  $z_j$  aktivasyon fonksiyonunun çıktısıdır.

$$y_i = \sum_{j=1}^H v_{ji} z_j + v_{0i} \quad (4.23)$$

Aktivasyon fonksiyonun çıktısı olan  $z_j$  değerini girdi olarak alan ikinci katmanın çıktısı  $y_i$ 'dir. Bu katmandaki hücrelerin ağırlıkları  $v_{ji}$ , eşik değeri ise  $v_{0i}$ 'dir. Aynı işlemler son katmana kadar devam eder. Son katmanda softmax aktivasyon fonksiyonu kullanılarak sınıfların olasılıkları belirlenir. Softmax fonksiyonu (4.24)'te verilmiştir.

$$f(y_i) = \frac{e^{y_i}}{\sum_{j=0}^K e^{y_j}} \quad (4.24)$$

$K$  değeri çıktı sınıf sayısını,  $y_i$  ise softmax fonksiyonundan bir önceki yoğun katmanın çıktısını belirtir. Bu çıktının olasılık değeri ile girdi verisinin etiketi hata fonksiyonunda kullanılarak hata değeri hesaplanır. Ağ üzerinde geriye doğru hesaplama yapılarak katmanlardaki ağırlıklar güncellenir. Bu çalışmada kullanılan hata fonksiyonu (4.25)'te verilmiştir.

$$L = -\frac{1}{N} \sum_{i=1}^N \left[ \sum_{j=1}^M [B_{ij} \log(\hat{B}_{ij})] \right] \quad (4.25)$$

Burada  $L$ , hesaplanan hata değerini,  $M$  sınıf sayısını,  $N$  ise örnek sayısını gösterir.  $B_{ij}$  ve  $\hat{B}_{ij}$  ise sırasıyla örneğin etiketi ve modelin çıktısıdır.



Şekil 4.3.1.2 Yapay Sinir Ağı ile Çok Sınıflı Sınıflandırma.

Sinir ağırları, uygulandıkları problem bazında başarımlarını arttırmak için eğitim verileri kullanırlar. Bununla birlikte, en iyi çalışma parametreleri seçildiğinde bu algoritmalar sayesinde büyük veri kümeleri yüksek bir hızda sınıflandırılabilir. Sinir ağırlarıyla sınıflandırma yapmak için tasarlanan model yapısı Şekil 4.3.1.2’de gösterilmiştir. Bu modelde girdi katmanında 20 hücre, 3 adet yoğun katmanın her birinde 30 hücre, çıktı katmanında 9 hücre vardır. Girdi ve yoğun katmanlarda ReLU aktivasyon fonksiyonu kullanılmıştır. Çıktı katmanında ise softmax aktivasyon fonksiyonu kullanılmıştır.

### 4.3.2 Destek Vektör Makinesi (DVM)

Destek vektör makinesi algoritmasının amacı, N-boyutlu uzaydaki farklı kümelere ait verileri birbirinden ayıran hiper sınır bulmaktır. Bir veri kümesinin sahip olduğu iki sınıfı birbirinden ayırmak için kullanılacak birçok hiper sınır vardır [33]. Algoritma, bu iki sınıfı en iyi şekilde ayıran bir sınır bulur. Her sınıf için bu sınır çizgisinden belirli marj değerinde destek vektörleri oluşturulur. Bu marj mesafesini en iyi değerinde hesaplamak algoritmanın sınıflandırma performansı için önemli bir iyileştirme sağlar. Destek vektörleri, hiper sınıra yakın olan ve hiper sınırın konumunu ve yönünü etkileyen veri noktalarıdır. Bu destek vektörleri kullanarak sınıflandırıcının marjı maksimize edilir.

Hiper sınırlar, örneklerin sınıflandırılmasına yardımcı olan karar sınırlarıdır. Hiper sınırın her iki tarafına düşen örnekler farklı sınıflara ait olabilir. Ayrıca, hiper sınırın boyutu, öznitelik vektörü boyutuna bağlıdır. Öznitelik vektörü boyutu 2 ise, hiper sınır sadece bir çizgidir. Öznitelik vektörü boyutu 3 ise, hiper sınır iki boyutlu bir düzlem olur. Öznitelik vektörü boyutu  $b$  ise hiper sınır  $b - 1$  boyutludur [34].

DVM algoritmasında girdi  $x = x_1, x_2, \dots, x_n$  ise;

$$0 \leq \alpha_i \quad , \quad \sum_{i=1}^n \alpha_i z_i = 0 \quad \text{olduğu durumda}$$

$$L_D(\alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j z_i z_j K(x_i, x_j) \quad (4.26)$$

(4.26) eşitliği optimize edilerek destek vektörleri ( $x_i$ ) ve destek vektör ağırlıkları ( $\alpha_i$ ) hesaplanır.  $K(x_i, x_j)$ , kernel fonksiyonunu ifade etmektedir. Bu çalışma için kernel fonksiyonu (4.27)'de verilen gaussian fonksiyonu seçilmiştir.

$$K(x_i, x) = \exp\left(-\frac{1}{2}\|x_i - x\|^2\right) \quad (4.27)$$

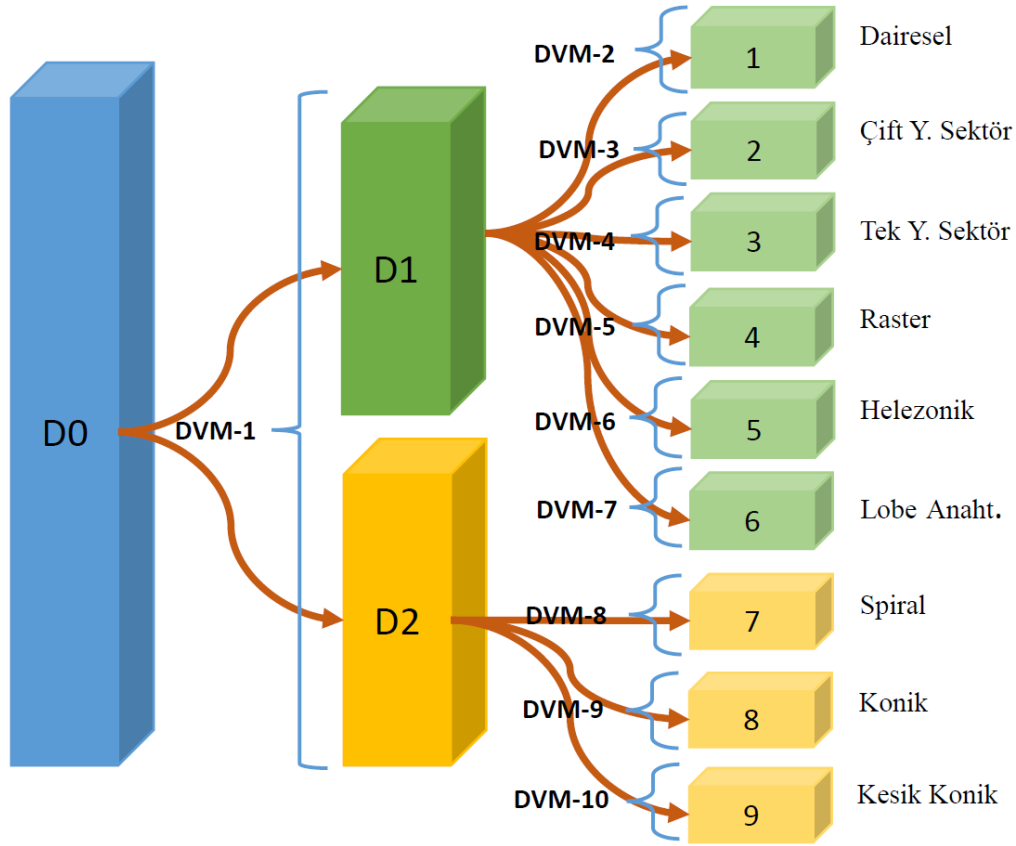
Test örneğinin sınıfı (4.28)'de verilen formüle göre bulunur.  $S$ , destek vektörleri kümesini,  $y$  çıktı sınıfını göstermektedir.

$$g(x) = \sum_{x_i \in S} \alpha_i z_i K(x_i, x), \quad y = \begin{cases} 1 & g(x) > 0 \\ 2 & g(x) \leq 0 \end{cases} \quad (4.28)$$

Basit DVM yapısı çok sınıflı sınıflandırmayı desteklemez. İkili sınıflandırmayı ve veri noktalarını iki sınıfa ayırmayı destekler. Çok sınıflı sınıflandırma için, çoklu sınıflandırma problemini çok sayıda ikili sınıflandırma problemlerine böldükten sonra aynı prensip uygulanır.

Çoklu sınıflandırma işlemlerinde DVM iki türlü kullanılır. Birincisi bire-bir (one-vs-one) yaklaşımı olarak adlandırılan yöntemdir. Bu yöntemde, her iki sınıf arasında karşılıklı sınır belirlemek için veri noktaları yüksek boyutlu uzaya eşlenir. Her sınıf çifti için bir DVM kullanılır. İkincisi ise bire-diğerleri (one-vs-all) yaklaşımıdır. Bu yaklaşımda her sınıf için bir DVM kullanılır. Bire-diğerleri yaklaşımında, bir sınıfla diğer sınıfları aynı anda ayırmak için bir hiper sınır kullanılır. Bu hiper sınır veri kümesini iki gruba ayırır. Bunlar; seçilen sınıf için bir grup ve diğer tüm sınıflar için bir gruptur. Bu çalışmada DVM ile çok sınıflı sınıflandırma yapılması için Python sklearn kütüphanesine ait *SVC* ve *OneVsRestClassifier* methodları kullanılmıştır [35].

Şekil 4.3.2.1'de gösterilen destek vektör makineleri kullanılarak çok sınıflı sınıflandırma yönteminde bire-diğerleri yaklaşımı kullanılmıştır. DVM-1, sürekli aydınlatmalı ve taramalı aydınlatmalı anten tarama tiplerini birbirinden ayırmaktadır. DVM-2'den DVM-10'a kadar olan işlemlerde bire-diğerleri yaklaşımı kullanılarak sınıflandırma yapılmıştır. D0, algoritmaya girdi olan örnekleri; D1, Taramalı aydınlatma örneklerini; D2, Sürekli aydınlatmalı örnekleri ifade eder.



Şekil 4.3.2.1 Destek Vektör Makinesi ile Çok Sınıflı Sınıflandırma.

Modelin eğitimi sonunda DVM-1 için destek vektör sayısı 100-130 arasında, DVM-2 için destek vektör sayısı 90-110 arasında, DVM-3 için destek vektör sayısı 70-95 arasında, DVM-4 için destek vektör sayısı 40-55 arasında, DVM-5 için destek vektör sayısı 60-80 arasında, DVM-6 için destek vektör sayısı 130-160 arasında, DVM-7 için destek vektör sayısı 30-55 arasında, DVM-8 için destek vektör sayısı 110-130 arasında, DVM-9 için destek vektör sayısı 55-75 arasında, DVM-10 için destek vektör sayısı 85-115 arasındadır.

### 4.3.3 K-En Yakın Komşu (K-EYK) Algoritması

Gözetimli öğrenme yöntemlerinden biri olan K-EYK algoritması, test örneği ile eğitim örnekleri arasındaki mesafeyi ölçerek test örneğinin ait olduğu sınıfı belirler. Eğitim örnekleri arasındaki en benzer K örneğin çoğu belirli bir sınıfa aitse, test örneği de bu sınıfa aittir. K genellikle tek tam sayıdır. K-EYK algoritması, test örneğinin sınıfını belirlemek için sadece en yakın eğitim örneklerinin sınıflarını değerlendirir [36].



Algoritmayı beş adımda tanımlayabiliriz ;

1. K belirlenir.
2. Eğitim örneklerinden test örneğine olan mesafeler hesaplanır.
3. Mesafeler listelenir ve en yakın komşular minimum mesafeye göre bulunur.
4. En yakın komşularda aynı sınıfta olanların sayısı belirlenir.
5. Test örneği, en fazla sayıda en yakın komşusu olan sınıfa atanır.

#### **4.4 Benzetim Düzenekleri**

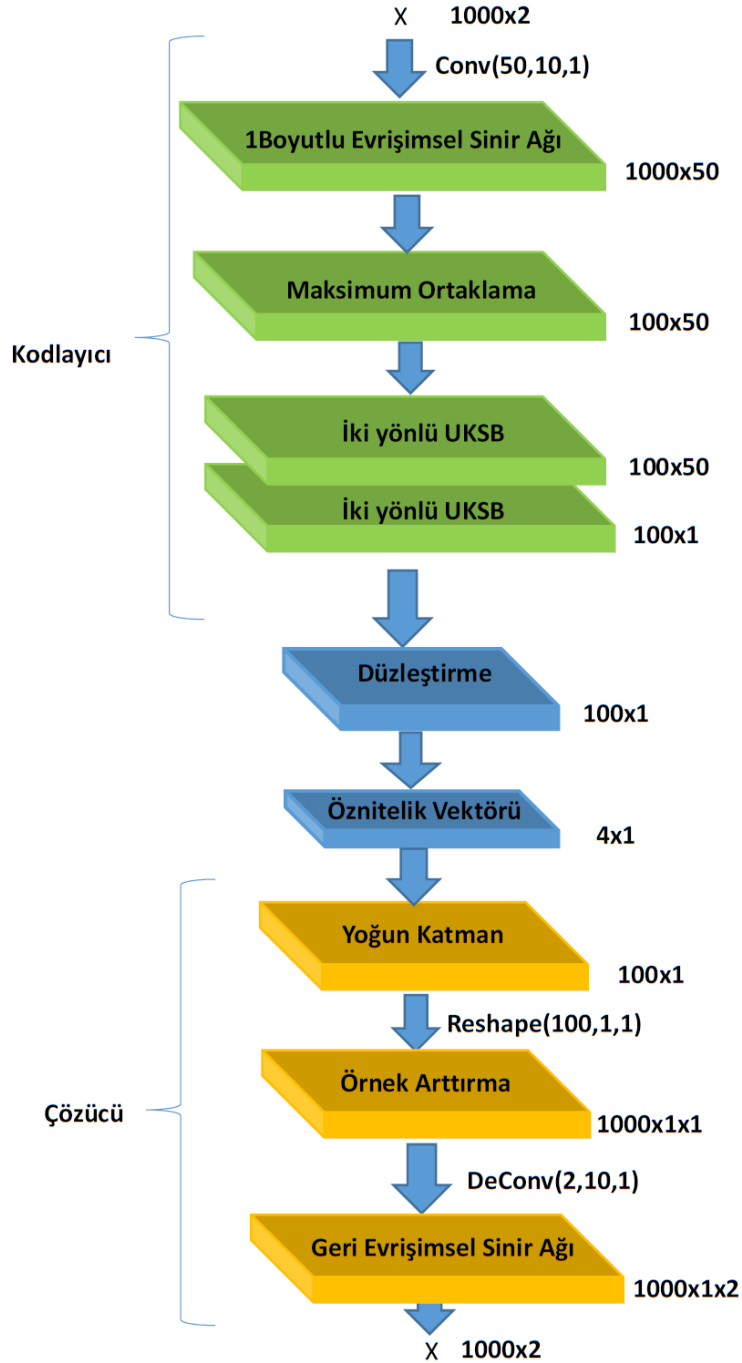
Bu çalışmada, Anten Tarama Tipi (ATT) belirleme yöntemlerini eğitmek ve test etmek için Bölüm 3'te anlatılan anten tarama örüntüsü üretici ile üretilen yapay veriler kullanılmıştır. Bu çalışmada kullanılan her ATT için 10000 tarama örüntüsü örneği üretilmiştir. Her tarama örüntüsü örneği için Anten Tarama Periyodu (ATP) değeri, Konik, Spiral ve Kesik Konik ATT'ler için 10 - 100 ms aralığında; diğer ATT'ler için 0.1-10s aralığında rastgele seçilmiştir. Her tarama örneğinde, farklı anten tarama örüntüleri oluşturmak için radar ve EH sistemlerinin konumları rastgele değiştirilmiştir. Yapay olarak üretilen verilere Bölüm 4'te anlatılan ön işleme adımları uygulanmıştır. Konik ve Spiral ATT verileri sürekli olduğundan, bu tiplerde 1/5 oranında örnekleme yapılmıştır. 90000 anten tarama örüntüsü örneğinin %80'i eğitim, %20'si test için kullanılmıştır.

Bu kısımda eğitim ve test düzenekleriyle yapılan işlemler detaylı olarak anlatılacaktır. Otomatik kodlayıcı yöntemiyle öznitelik çıkarma işlemi ve gözetimli öğrenme yöntemiyle sınıflandırma işleminde modellerin eğitilmesi gerektiği için bu işlemler Eğitim Düzenekleri kısmında anlatılacaktır. İstatiksel öznitelik çıkarma ve kümeleme algoritmaları için modelin eğitilmesi gerekmediğinden Test Düzenekleri kısmında bu modeller ile yapılan testler anlatılacaktır.

##### **4.4.1 Eğitim Düzenekleri**

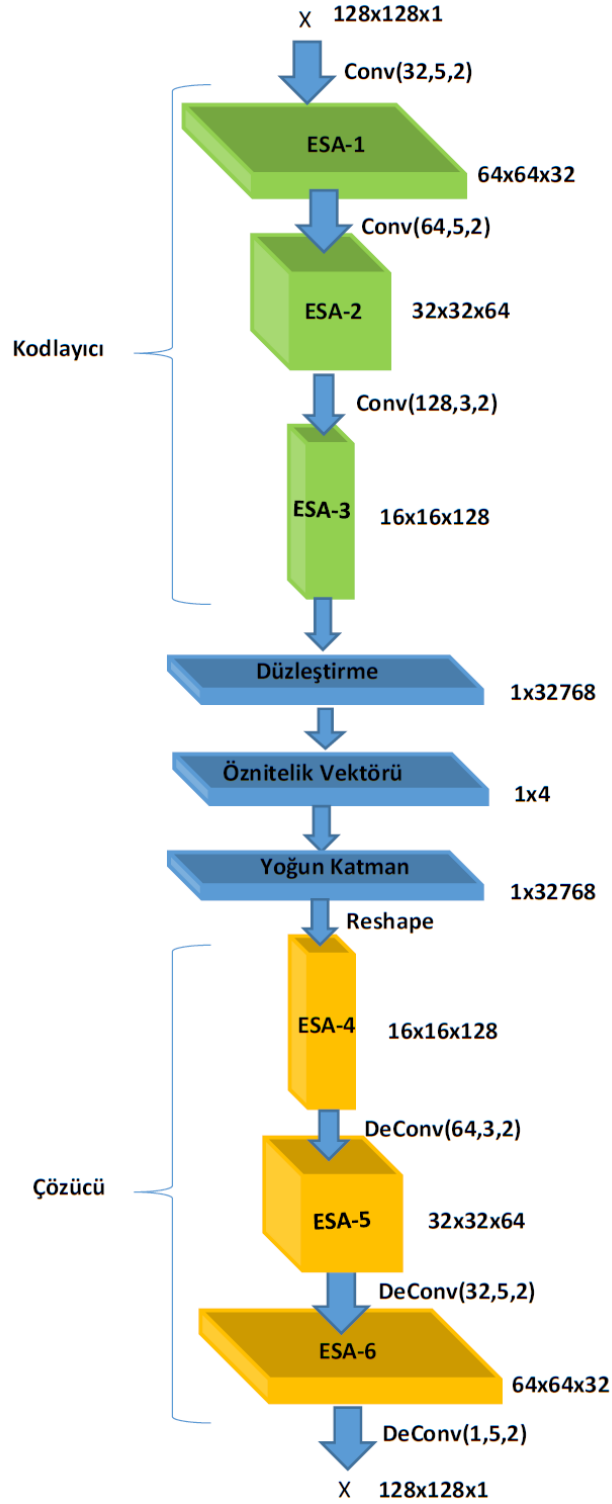
UKSBOK-ÖÇ ve ESAOK-ÖÇ yöntemlerinin öznitelik vektörü katmanının boyutu 4 olarak belirlenmiştir. Modellerde batch boyutu 128 olarak seçilmiştir. Kayıp fonksiyonu olarak MSE (Ortalama Kare Hata), optimizasyon fonksiyonu olarak Adam kullanılmıştır [37]. Modellerin eğitim ve testi için Python TensorFlow kütüphanesi [38] kullanılmıştır. Şekil 4.4.1.1 gösterilen model için hazırlanan eğitim verileri kullanılarak 100 olarak

seçilen tekrarlamaya sayısı (epoch) ile model eğitilmiştir. Eğitilen bu model bölüm 4.4.2’de anlatıldığı gibi test işlemlerinde kullanılacaktır. Şekil 4.4.1.1’de X girdisi 1000x2 boyutunda PA ve TOA değerlerini içeren matrisi; Conv(filtre,kernel,stride), evrişim işlemini; DeConv(filtre,kernel,stride), geri evrişim işlemini ifade eder.



Şekil 4.4.1.1 UKSBOK-ÖÇ modelinin Eğitimi.

Şekil 4.4.1.2’de X girdisi 128x128 boyutunda gri ton zaman-frekans (TFI) görüntüsüdür. Model için hazırlanan eğitim verileri kullanılarak 100 olarak seçilen tekrarlamaya sayısı (epoch) ile eğitim yapılmıştır.

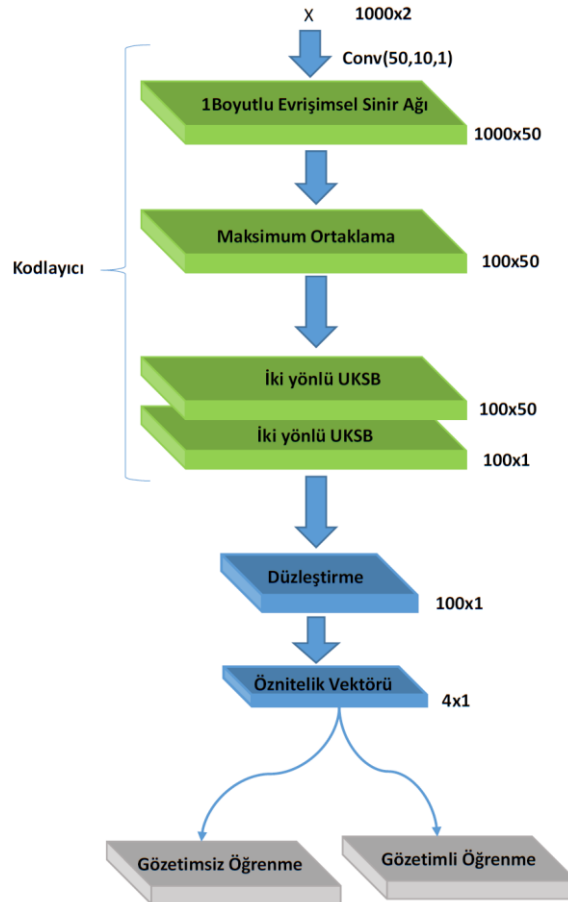


Şekil 4.4.1.2 ESAOK-ÖÇ modelinin Eğitimi.

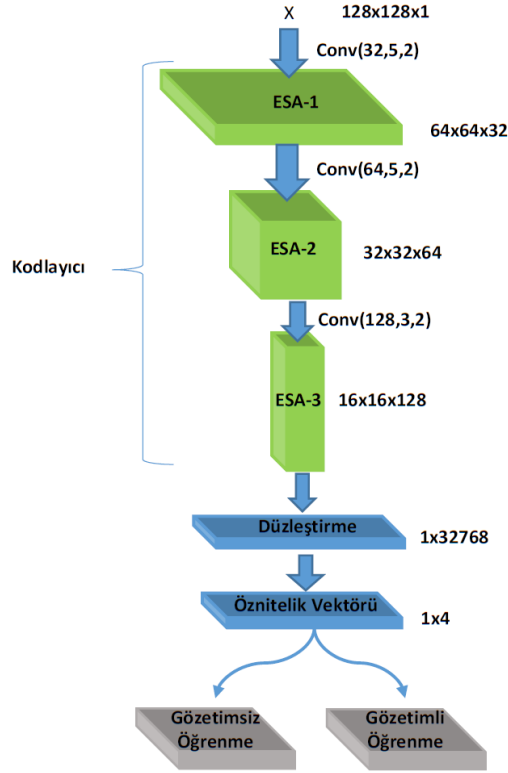
Şekil 4.3.1.1 ve Şekil 4.3.2.1’de gösterilen Sinir Ağı ve DVM modelleri için eğitim verileri, öznitelik çıkarma yöntemleriyle elde edilen verilerdir. Bu eğitim verileri kullanılarak Sinir ağı ve DVM modeli eğitilmiştir. Yapay Sinir Ağı modelinde, batch boyutu olarak 128, optimizasyon fonksiyonu olarak Adam [37] seçilmiştir.

#### 4.4.2 Test Düzenekleri

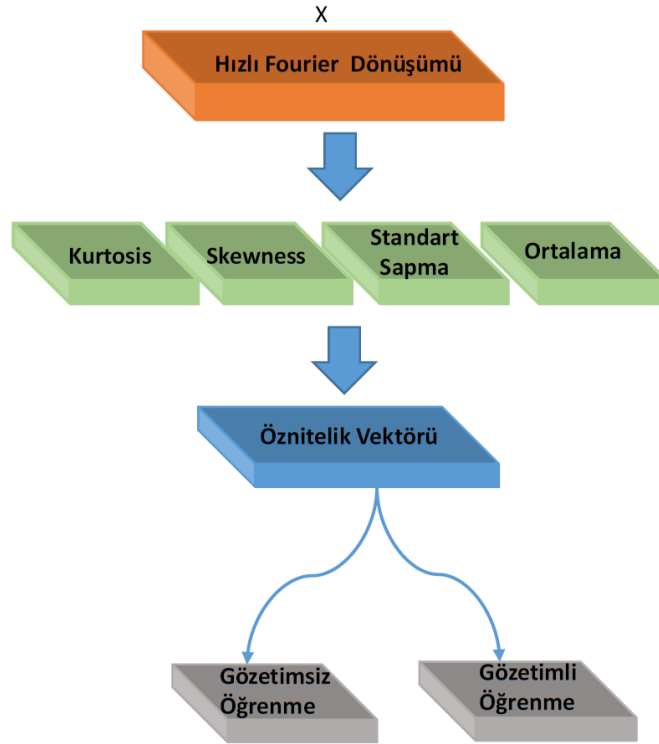
Sentetik olarak üretilen test verileri kullanılarak eğitim yapılan modellerin ve eğitim gerektirmeyen modellerin test düzenekleri kurulmuştur. Şekil 4.4.2.1’de UKSBOK-ÖÇ modelinin test düzeneği, Şekil 4.4.2.2’de ESAOK-ÖÇ modelinin test düzeneği, Şekil 4.4.2.3 İÖÇ modelinin test düzeneği gösterilmiştir. Üç öznitelik çıkarma modelinin de öznitelik vektörü boyutu 4 olarak seçilmiştir. İÖÇ modelinde öznitelik vektörü ortalama, standart sapma, çarpıklık ve basıklık değerlerinden oluşur. UKSBOK-ÖÇ ve ESAOK-ÖÇ modellerinde ise öznitelik vektörü katmanındaki 4 adet sinir hücresinin çıktısıdır.



Şekil 4.4.2.1 UKSBOK-ÖÇ yöntemiyle Anten Tarama Tipinin Belirlenmesi.



Şekil 4.4.2.2 ESAOK-ÖÇ yöntemiyle Anten Tarama Tipinin Belirlenmesi.



Şekil 4.4.2.3 İÖÇ yöntemiyle Anten Tarama Tipinin Belirlenmesi.

### 4.4.3 Öznitelik Vektörlerinin Görselleştirilmesi

Üç farklı öznitelik çıkarma modeliyle elde edilen öznitelikler t-distributed Stochastic Neighbor Embedding (t-SNE) algoritması kullanılarak görselleştirilmiştir. t-SNE Laurens van der Maaten ve Geoffrey Hinton [39] tarafından geliştirilen verilerin görselleştirilmesi için kullanılan gözetimsiz bir makine öğrenimi algoritmasıdır.

Algoritma, yüksek boyutlu bir uzayda yakın noktalar arasındaki ikili benzerlik değerlerinin bulunmasıyla başlar.

$$p_{j|i} = \frac{\exp(-\|x_i - x_j\|^2/2\sigma_i^2)}{\sum_{k \neq l} \exp(-\|x_k - x_l\|^2/2\sigma_i^2)} \quad (4.29)$$

$X = x_1, x_2, \dots, x_n$  olmak üzere;  $x_i$  ve  $x_j$  veri noktaları arasındaki yüksek boyutlu Öklid mesafesini  $p_{j|i}$  koşullu olasılığına dönüştürür.  $x_i$ ,  $x_i$  noktasında merkezlenmiş bir normal dağılım altında olasılık yoğunluğunun oranına bağlı olarak  $x_j$ 'yi komşusu olarak seçecektir.  $\sigma_i$ ,  $x_i$  veri noktasında merkezlenen normal dağılımın varyansıdır. Bir veri çiftinin  $p_{j|i}$  koşullu olasılık değeri, benzerliği ile orantılıdır. Birbirine yakın veri noktaları için  $p_{j|i}$  nispeten yüksek olacak ve uzak olan noktalar için  $p_{j|i}$  küçük olacaktır.

Koşullu olasılıklar, aşağıda gösterildiği gibi iki olasılığın ortalaması alınarak ortak olasılık değeri hesaplanır.

$$p_{ij} = \frac{p_{i|j} + p_{j|i}}{2n} \quad (4.30)$$

Düşük boyutlu uzayda iki nokta arasındaki benzerliği hesaplamak için Student-t dağılımını kullanılır. Bu dağılımı kullanarak,  $q_{ij}$  ortak olasılıkları şu şekilde tanımlanır.

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq l} (1 + \|y_k - y_l\|^2)^{-1}} \quad (4.31)$$

$Y = y_1, y_2, \dots, y_n$  olmak üzere;  $y_i$  ve  $y_j$ ,  $x_i$  ve  $x_j$  yüksek boyutlu veri noktalarının düşük boyutlu karşılıklarıdır.

Kullback-Leibler (KL) Divergence temelli optimizasyon kullanarak  $p_{ij}$  ve  $q_{ij}$  arasındaki uyumsuzluğun en aza indirilmesi amaçlanır.

$$C = KL(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}} \quad (4.32)$$

KL Divergence fonksiyonunun gradyanı (4.33)'de ifade edilmiştir.

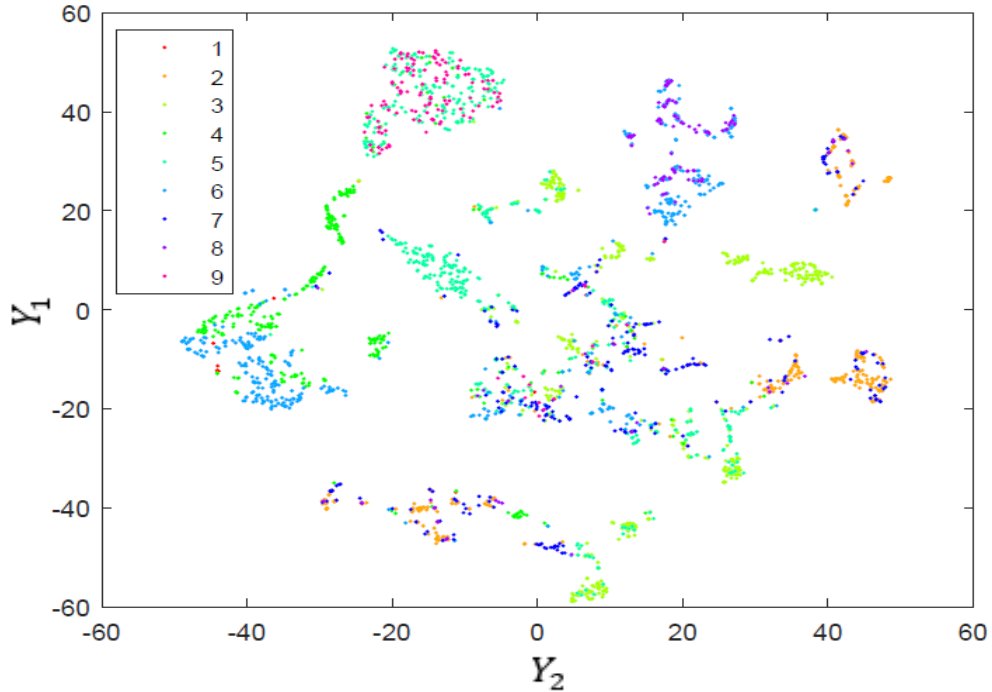
$$\frac{\delta C}{\delta y_i} = 4 \sum_j (p_{ij} - q_{ij})(y_i - y_j) (1 + \|y_i - y_j\|^2)^{-1} \quad (4.33)$$

Düşük boyutlu veri noktaları (4.34) kullanılarak hesaplanır.

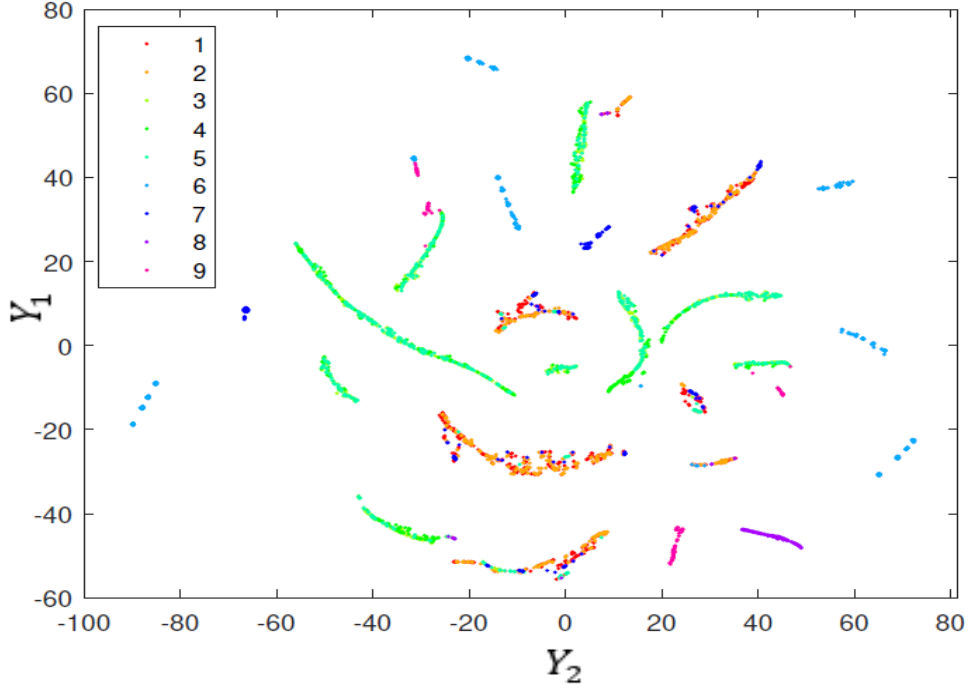
$$Y^{(t)} = Y^{(t-1)} + \alpha \frac{\delta C}{\delta Y} + \beta(t)(Y^{(t-1)} - Y^{(t-2)}) \quad (4.34)$$

$Y^{(t)}$ , t. adımdaki düşük boyutlu verileri ;  $\alpha$ , *learning\_rate* değerini;  $\beta(t)$ , t anındaki momentumu ifade eder. Bu işlemler *n\_iter* değerinde tekrar edilir.

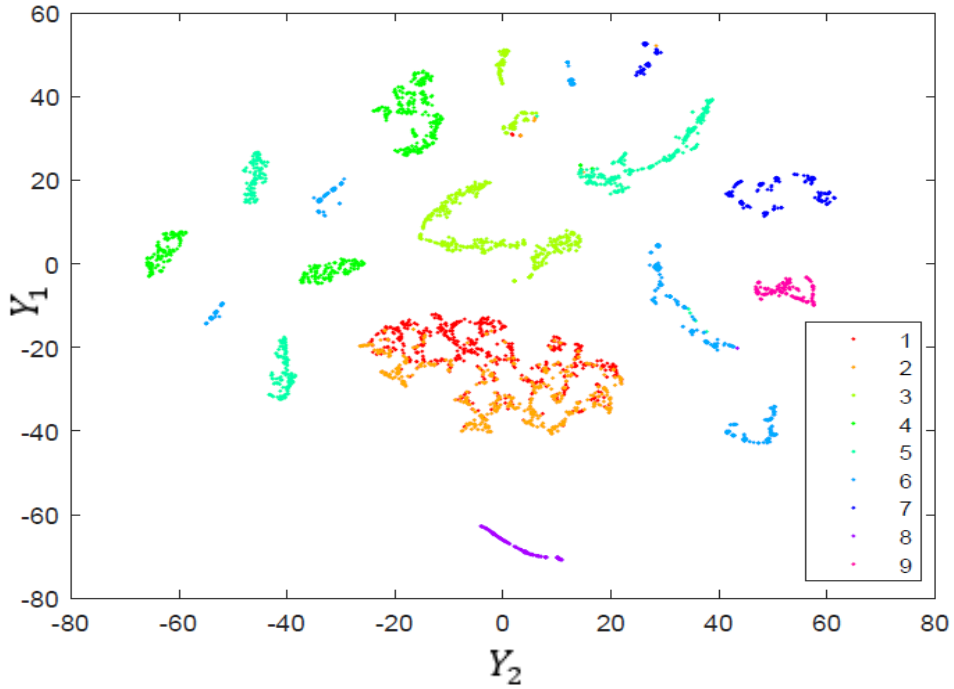
Şekil 4.4.2.4'te ESAOK-ÖÇ yöntemi, Şekil 4.4.2.5'te İÖÇ yöntemi, Şekil 4.4.2.6'da UKSBOK-ÖÇ yöntemi kullanılarak elde edilen öznelik vektörleri t-distributed Stochastic Neighbor Embedding (t-SNE) algoritması kullanılarak görselleştirilmiştir. Görselleştirme için Python sklearn kütüphanesi [35] kullanılmıştır. Algoritmanın kritik parametreleri *n\_components:2*, *perplexity:30*, *n\_iter:500*, *learning\_rate:200* olarak seçilmiştir. *n\_components*, yüksek boyutlu verilerin dönüştürülmesini istediğimiz alt boyut sayısıdır. *Perplexity*, t-SNE algoritmasında kullanılan en yakın komşuların sayısıdır. Daha büyük veri kümeleri genellikle daha büyük perplexity gerektirir. *n\_iter*, optimizasyon için maksimum yineleme sayısıdır.



Şekil 4.4.2.4 ESAOK-ÖÇ yöntemiyle çıkarılan özneliklerin t-SNE ile görselleştirilmesi. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.



Şekil 4.4.2.5 İÖÇ yöntemiyle çıkarılan özneliklerin t-SNE ile görselleştirilmesi. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.



Şekil 4.4.2.6 UKSBOK-ÖÇ yöntemiyle çıkarılan özneliklerin t-SNE ile görselleştirilmesi. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.



## 5. BENZETİM ÇALIŞMALARI

Test verileri kullanılarak her bir öznitelik çıkarma yöntemiyle öznitelik çıkarımı yapılmış, çıkarılan öznitelikler gözetimli ve gözetimsiz öğrenme modelleri girdi olarak verilerek ATT belirlenmiştir. Gözetimsiz öğrenme yönteminde K-Ortalamlar, K-Ortalamlar++ ve Hiyerarşik Kümeleme algoritmaları kullanılarak kümeleme analizi yapılmış ve analiz sonucunda yeni bir sınıflandırma yöntemi önerilmiştir. Gözetimli öğrenme yönteminde ise K-En Yakın Komşu (K-EYK), Yapay Sinir Ağı (YSA) ve Destek Vektör Makinesi (DVM) algoritmaları kullanılarak ATT belirlenmiştir. K-Ortalamlar, K-Ortalamlar++ ve Hiyerarşik Kümeleme algoritmaları farklı mesafe ölçütleriyle çalıştırılarak sonuçlar alınmıştır. Alınan sonuçlar Tablo 5.1, 5.2 ve 5.3'te verilmiştir. Kümele algoritmalarının başarımlar oranları aşağıdaki formül ile hesaplanmıştır.

$$R = \frac{\sum_{j=1}^n \gamma(s_i, \text{map}(r_i))}{n} \times 100 \quad (5.1)$$

Burada  $r_i$  ve  $s_i$  sırasıyla kümeleme algoritmasının çıktısı ve örneğin etiketidir.  $\text{map}$  en iyi sınıf kombinasyonunu temsil eder.  $n$  anten tarama örneklerinin toplam sayısını temsil eder.  $R$  ise algoritmanın başarımlar yüzdesini göstermektedir. Etiket ve sınıf şu formüle göre puanlanır.

$$\gamma(x, y) = \begin{cases} 1 & x = y \\ 0 & x \neq y \end{cases} \quad (5.2)$$

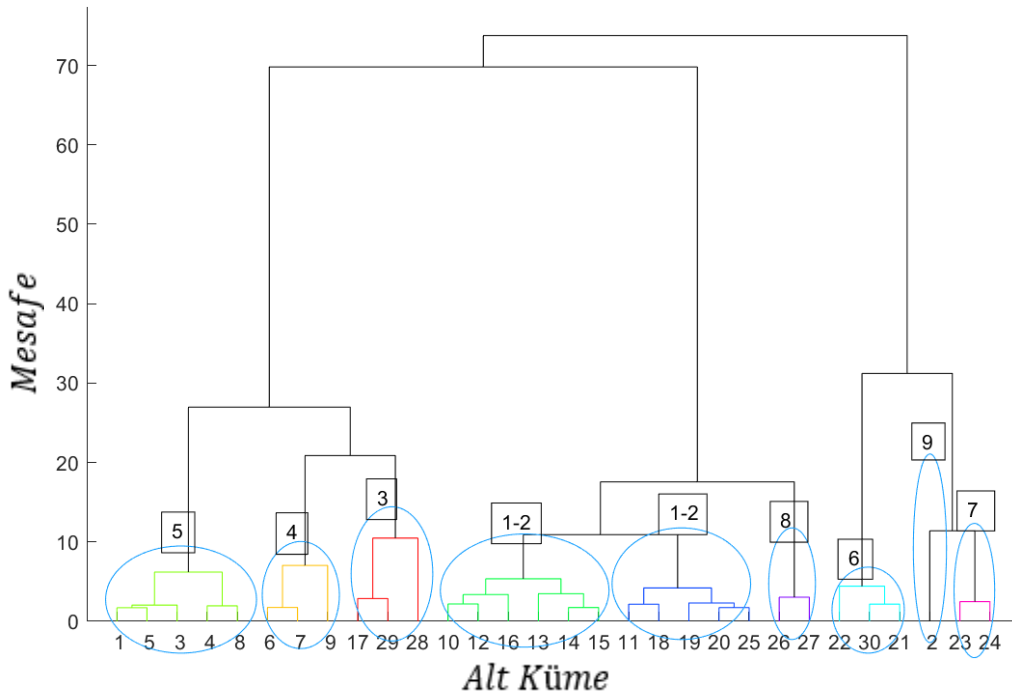
Tablo 5.1 UKSBOK-ÖÇ yöntemi ile elde edilen öznitelik vektörünün farklı kümeleme yöntemlerindeki başarımlar yüzdesi.

Mesafe Ölçütleri	K-Ortalamlar	K-Ortalamlar++	Hiyerarşik Kümeleme
Kosinüs	%50.4	%62.6	%93.9
Öklid	%65.9	%75.4	%94.1
Manhattan	%81.2	%85.3	%93.5
Bray-Curtis	%57.3	%64.5	%93.2
Canberra	%62.6	%76.9	%93.8
Chebyshev	%63.7	%74.2	%94.6
Mahalanobis	%71.0	%83.1	%96.8
Minkowski	%65.0	%71.5	%95.2

Farklı mesafe ölçütleri kullanılarak K-Ortalamlar, K-Ortalamlar++ ve Hiyerarşik kümeleme algoritmaları kullanılarak alınan sonuçlar karşılaştırılmıştır. UKSBOK-ÖÇ

yöntemi ile oluşturulan özneliklerin farklı kümeleme algoritmalarıyla alınan sonuçları Tablo 5.1'de verilmiştir.

Hiyerarşik kümeleme, tüm mesafe ölçütleri için K-Ortalamlar ve K-Ortalamlar++ algoritmalarından daha iyi başarımlarına sahiptir. K-Ortalamlar ve K-Ortalamlar++ algoritmaları, Manhattan mesafe ölçütünü kullanırken en iyi performans oranını vermiştir. Hiyerarşik kümeleme algoritması Mahalanobis mesafe ölçütü kullanıldığında en iyi başarımlarını vermiştir. Bunun nedeni, Mahalanobis mesafe ölçütünün küme dağılımında göz önüne alarak mesafeyi hesaplamasıdır.



Şekil 5.1 UKSBOK-ÖÇ yöntemi ile elde edilen özneliklerin Mahalanobis mesafe ölçütü kullanılarak hiyerarşik kümeleme dendrogramı. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.

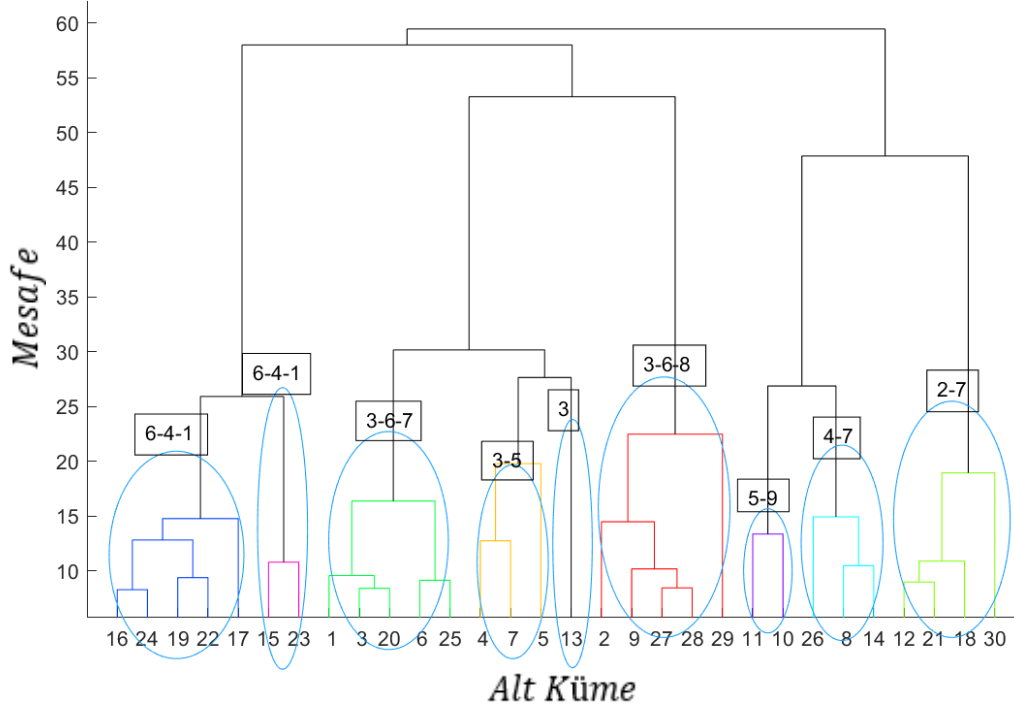
UKSBOK-ÖÇ yöntemiyle elde edilen öznelik vektörünün hiyerarşik kümeleme algoritmasında kullanılmasıyla oluşan kümeleme dendrogramı Şekil 5.1'de gösterilmiştir. Anten tarama tipi olan 9 kümenin alt kümelerini de görebilmek için dendrogram oluşturulurken maksimum alt küme sayısı 30 seçilmiştir. Dendrogramda mesafe değeri 50 üzeri olan 3 ana bağlantı vardır. Bunlar Çift Yönlü Sektörel-Dairesel-Tek Yönlü Sektörel, Helezonik-Raster-Lob anahtarlamalı ve Konik-Kesik Konik-Spiral tarama tipidir. Bölüm 2.2'de anlatılan anten tarama tiplerine baktığımızda Helezonik ve Raster

tipinin benzer bir örüntüye sahip olduğu görülmektedir. Yani Şekil 5.1'de gösterilen dendrogramda bu tiplerin aynı ana bağlantıya sahip olması modelin doğru karar verebildiğinin bir göstergesidir. Dairesel ve Tek Yönlü Sektörel arasındaki fark huzme genişliğidir. Çift Yönlü Sektörel ise bir anten tarama periyodunda iki ana aydınlatmaya sahiptir. Spiral, Konik ve Kesik Konik benzer bir tarama örüntüsüne sahiptir. Sadece örüntü üzerindeki modülasyon değerleri farklıdır. Bölüm 4.4.3'te anlatılan t-SNE algoritması kullanılarak oluşturulan Şekil 4.4.2.6 incelendiğinde Raster ve Helezonik sınıflarının birbirinin içine geçtiği görülmektedir.

Tablo 5.2 ESAOK-ÖÇ yöntemi ile elde edilen öznitelik vektörünün farklı kümeleme yöntemlerindeki başarımlar yüzdesi.

Mesafe Ölçütleri	K-Ortalamlar	K-Ortalamlar++	Hiyerarşik Kümeleme
Kosinüs	%45.0	%56.2	%82.6
Öklid	%56.2	%64.4	%82.5
Manhattan	%65.3	%69.6	%83.6
Bray-Curtis	%45.5	%58.2	%84.6
Canberra	%52.3	%62.4	%75.2
Chebyshev	%53.6	%58.6	%77.4
Mahalanobis	%51.5	%71.8	%83.6
Minkowski	%53.8	%65.4	%81.9

ESAOK-ÖÇ yönteminin karşılaştırma sonuçları Tablo 5.2'de verilmiştir. Hiyerarşik kümeleme algoritması, tüm mesafe ölçütleri için K-Ortalamlar ve K-Ortalamlar++ algoritmasından daha iyi sonuçlar vermiştir. Hiyerarşik kümeleme algoritması, Bray-Curtis mesafe ölçütü kullanıldığında %84.6 ile en iyi performans oranını vermiştir. ESAOK-ÖÇ yöntemiyle elde edilen öznitelik vektörünün hiyerarşik kümeleme algoritmasında kullanılmasıyla oluşan kümeleme dendrogramı Şekil 5.2'de gösterilmiştir. Dendrogramda mesafe değeri 50 üzeri olan 4 ana bağlantı vardır. Bunlar Konik-Dairesel-Helezonik, Tek Yönlü Sektörel-Çift Yönlü Sektörel-Konik-Spiral, Tek Yönlü Sektörel-Konik-Lob anahtarlamalı, Spiral-Çift Yönlü Sektörel-Dairesel-Raster-Kesik Konik tarama tipidir. Bu ana bağlantılardaki sınıfların birbiriyle karışması anten tarama örüntülerinin Şekil 4.3'te gösterilen TFI'ya dönüştürülürken oluşan bozulmanın sonucudur. Bölüm 4.4.3'te anlatılan t-SNE algoritması kullanılarak oluşturulan Şekil 4.4.2.4 incelendiğinde kümelerin öznitelik vektörü çıkarma işleminden dolayı karıştığı daha iyi anlaşılmaktadır.

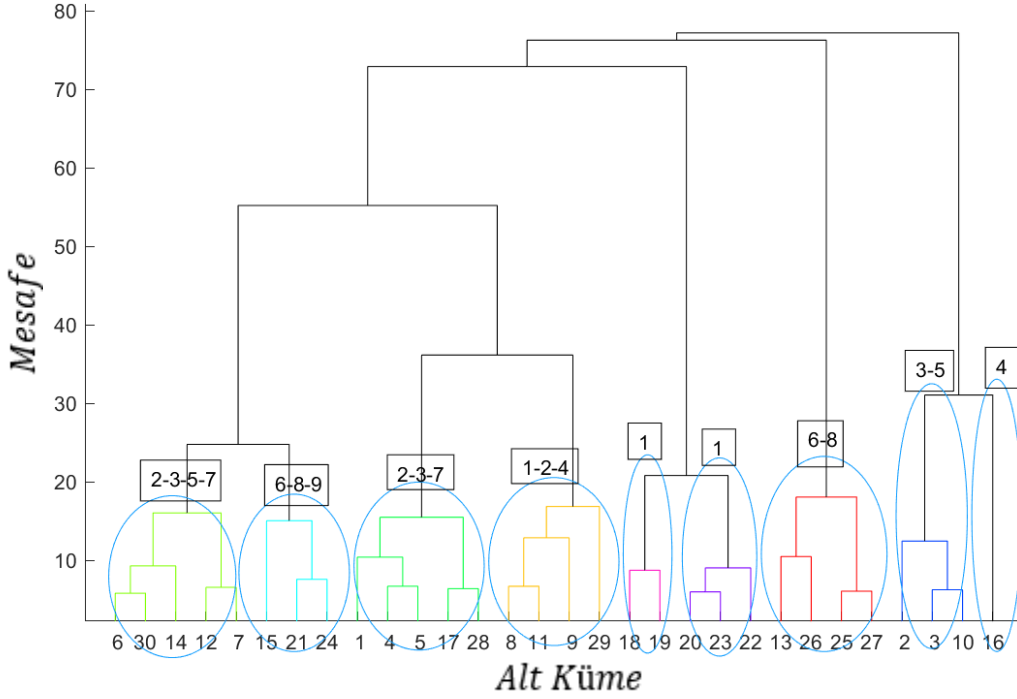


Şekil 5.2 ESAOK-ÖÇ yöntemi ile elde edilen özneliklerin Bray-Curtis mesafe ölçütü kullanılarak hiyerarşik kümeleme dendrogramı. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.

Tablo 5.3 İÖÇ yöntemi ile elde edilen öznelik vektörünün farklı kümeleme yöntemlerindeki başarımlar yüzdesi.

Mesafe Ölçütleri	K-Ortalamalar	K-Ortalamalar++	Hiyerarşik Kümeleme
Kosinüs	%25.0	%46.6	%72.6
Öklid	%36.4	%43.0	%75.6
Manhattan	%48.7	%52.3	%76.2
Bray-Curtis	%25.2	%38.6	%70.5
Canberra	%32.3	%43.6	%68.5
Chebyshev	%33.8	%39.6	%74.2
Mahalanobis	%35.6	%48.6	%78.0
Minkowski	%27.8	%39.4	%75.6

İÖÇ yönteminin karşılaştırma sonuçları Tablo 5.3'te verilmiştir. En iyi performans Mahalanobis mesafe ölçütü ile hiyerarşik kümelemedir. K Ortalamalar++ ve Hiyerarşik kümeleme algoritmaları arasında en az %24 başarımlar oranı farkı vardır.

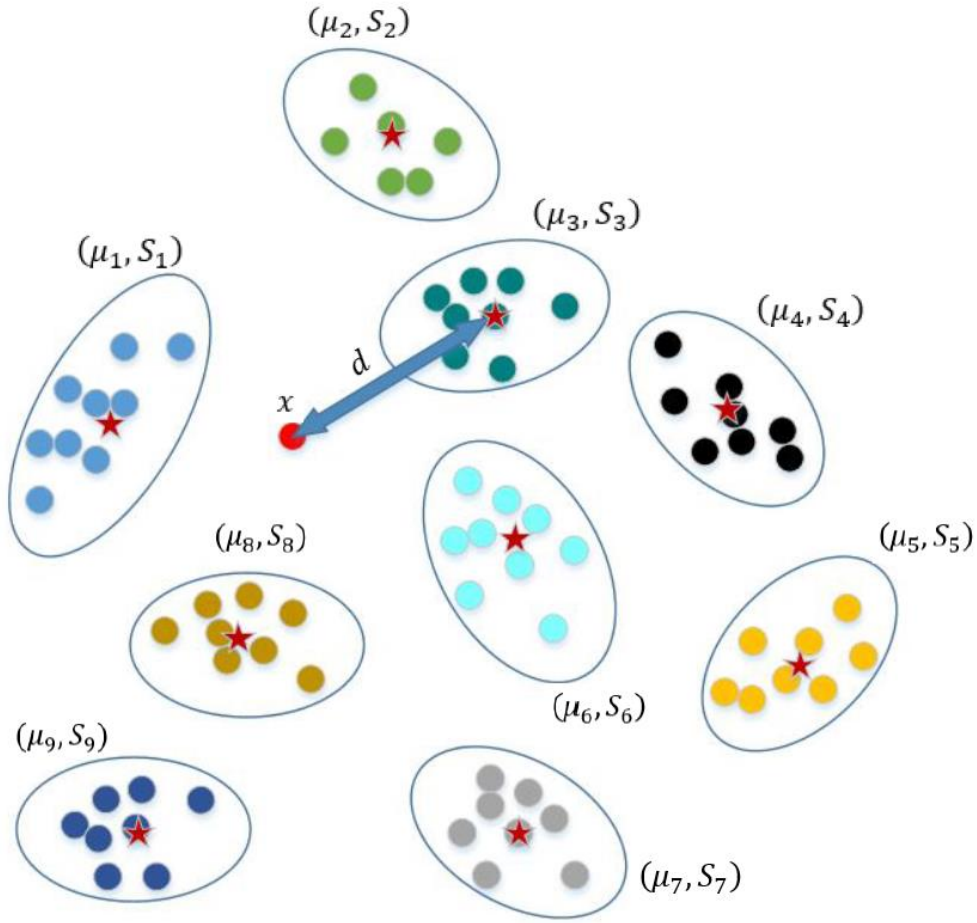


Şekil 5.3 İÖÇ yöntemi ile elde edilen özneliklerin Mahalanobis mesafe ölçütü kullanılarak hiyerarşik kümeleme dendrogramı. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.

Şekil 5.3'te gösterilen dendrogramda mesafe değeri 50 üzeri olan 5 ana bağlantı vardır. Bu 5 ana bağlantı altındaki bütün sınıfların birbirine karıştığı görülmektedir. Bu karışmanın nedeni istatistiksel öznelik çıkarma yönteminin performansının düşük olmasıdır. Şekil 4.4.2.5'te gösterilen 4 boyutlu öznelik vektörünün t-SNE algoritması ile görselleştirilmiş hali incelendiğinde farklı sınıflara ait özneliklerin birbirine karıştığı görülmektedir.

Kümeleme analizi sonucunda en iyi kümeleme performansı Mahalanobis mesafe ölçütü kullanılan hiyerarşik kümeleme algoritması olduğu tespit edilmiştir. Hiyerarşik kümeleme sonucunda elde edilen kümeler kullanılarak yeni bir sınıflandırma yöntemi tasarlanmıştır. Bu yöntem Hiyerarşik Kümeleme algoritması ile oluşturulan küme merkezleri ve kovaryans matrisleri kullanılarak test örneğinin sınıfı belirlenmiştir. Şekil 5.4'te gösterilen küme merkez ( $\mu$ ) ve kovaryans matrisi ( $S$ ) kullanılarak ( $x$ ) test örneği ile küme merkezleri arasında Mahalanobis mesafesi ( $d$ ) hesaplanmıştır. Mesafe hesaplamaları sonucunda  $x$  örneği en yakın kümeye atanır.  $x$  örneğinin atandığı kümenin merkezi ve kovaryans matrisi güncellenir. Hiyerarşik Kümeleme kullanılarak kümelerin

belirlenmesi ve sınıflandırma işleminin bu kümelere göre yapılmasından dolayı bu yönteme Hiyerarşik Kümeleme Tabanlı Sınıflandırma (HKS) denilmiştir.



Şekil 5.4 Hiyerarşik Kümeleme Tabanlı Sınıflandırma (HKS) yöntemi

Öznitelik çıkarma yöntemlerinden elde edilen öznitelik vektörleri Yapay Sinir Ağı (YSA), Destek Vektör Makinesi (DVM), HKS ve K-EYK algoritmaları kullanılarak ATT belirleme işlemi yapılmıştır. Şekil 5.5'te İÖÇ yöntemi ile elde edilen özniteliklerin YSA kullanılarak sınıflandırma sonucu karşılaştırma matrisinde gösterilmiştir. Karşılaştırma matrisinde satırlar tahmin edilen sınıfa (Çıktı Sınıfı) ve sütunlar gerçek sınıfa (Hedef Sınıf) karşılık gelir. Yeşil çapraz hücreler, doğru sınıflandırılan gözlemlere karşılık gelir. Kırmızı çapraz olmayan hücreler, yanlış sınıflandırılmış gözlemlere karşılık gelir. Her hücrede gözlem sayısı gösterilir. Matrisin en sağındaki sütun doğru sınıflandırılan örneklerin, belirtilen sınıfa ait olduğu tahmin edilen tüm örneklerle oranını gösterir. Bu ölçümlere genellikle kesinlik (precision) denir. Matrisin en altındaki satır, her sınıfa ait

doğru sınıflandırılan örneklerin belirtilen sınıfa ait tüm örneklere oranını gösterir. Bu ölçümlere genellikle duyarlılık (recall) denir. Matrisin sağ alt köşesindeki hücre, genel doğruluğu gösterir. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlama{8} ve Kesik Konik{9} ATT'ler için sınıflandırma başarıları sırasıyla % 85.5 % 75.1, % 86.6, % 87.2, % 85.4, % 90.5, % 98.9, % 88.8 ve % 90.8'dir. Raster, Helezonik, Tek yönlü sektörel, Dairesel ve Çift yönlü sektörel tiplerinde hatalı sınıflandırılmış örnekler vardır. Sürekli aydınlatmalı olan ATT'ler ile taramalı aydınlatmalı olan ATT'lerde de hatalı sınıflandırma görülmektedir. Bu hatalı sınıflandırmaların sebebi İÖÇ yönteminin zayıflığından kaynaklanmaktadır. Dairesel sınıfı için kesinlik değeri %67.6 ile en düşük değerdir. Bu durum dairesel sınıflandırılmış örneğin doğru olma olasılığının düşük olduğunu gösterir.

**Karşılaştırma Matrisi**

1	1709	16	0	0	0	11	0	0	0	%98.4
2	59	1503	3	0	0	0	23	0	0	%94.6
3	30	59	1731	188	39	0	0	40	52	%80.9
4	193	211	0	1743	253	18	0	29	133	%67.6
5	7	153	110	0	1708	0	0	0	0	%86.3
6	0	0	156	0	0	1811	0	80	0	%88.5
7	2	58	0	0	0	25	1977	0	0	%95.9
8	0	0	0	0	0	76	0	1777	0	%95.9
9	0	0	0	69	0	59	0	74	1815	%90.0
	%85.5	%75.1	%86.6	%87.2	%85.4	%90.5	%98.9	%88.8	%90.8	%87.6
	1	2	3	4	5	6	7	8	9	
	Hedef Sınıfı									

Şekil 5.5 İÖÇ yöntemi ile elde edilen özneliklerin YSA ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlama{8}, Kesik Konik{9}.

Şekil 5.6'da İÖÇ yöntemi ile elde edilen özniteliklerin DVM kullanılarak sınıflandırma sonucu gösterilmiştir. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel{5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9} ATT'ler için sınıflandırma başarıları sırasıyla % 76.5 % 56.5, % 83.5, % 90.1, % 85, % 87.5, % 85.8, % 97.5 ve % 99.2'dir. DVM yönteminde Raster, Helezonik, Dairesel ve Çift Yönlü Sektörel tipleri arasındaki hatalı sınıflandırma oranı YSA yöntemine göre artmıştır. DVM yönteminde Süreli aydınlatmalı ve taramalı aydınlatmalı tiplerde karışan örnekler vardır. Bunun sebebi taramalı ve sürekli aydınlatma ayrımı yapan DVM modelinin sınıflandırma performansının İÖÇ yöntemi için kötü olmasıdır. Genel başarı oranına baktığımızda YSA kullanılarak yapılan ATT belirleme işleminin başarı oranı DVM yöntemine göre %3 daha iyidir.

Çıktı Sınıfı	1	2	3	4	5	6	7	8	9	
1	1531	248	0	0	0	38	0	0	0	%84.3
2	159	1129	15	5	0	0	43	0	0	%83.6
3	13	55	1671	180	32	0	0	49	6	%83.3
4	293	305	0	1802	268	13	0	0	10	%67.0
5	4	253	306	0	1700	0	0	0	0	%75.1
6	0	0	0	0	0	1750	241	0	0	%87.9
7	0	10	0	0	0	110	1716	0	0	%93.5
8	0	0	0	0	0	0	0	1951	0	%100
9	0	0	8	13	0	89	0	0	1984	%94.7
	%76.5	%56.5	%83.5	%90.1	%85.0	%87.5	%85.8	%97.5	%99.2	%84.6
	1	2	3	4	5	6	7	8	9	
	Hedef Sınıfı									

Şekil 5.6 İÖÇ yöntemi ile elde edilen özniteliklerin DVM ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.



Şekil 5.7'de İÖÇ yöntemi ile elde edilen özneliklerin K-EYK kullanılarak sınıflandırma sonucu gösterilmiştir. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel{5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9} ATT'ler için sınıflandırma başarıları sırasıyla % 81.2 % 73.2, % 48.2, % 46.0, % 56.9, % 94.8, % 96.9, % 92.3 ve % 53.9'dur. K-EYK yönteminde Raster, Tek Yönlü Sektörel, Dairesel ve Çift Yönlü Sektörel tipleri arasındaki hatalı sınıflandırma oranı YSA yöntemine göre artmıştır. Genel başarı oranına baktığımızda K-EYK yönteminin başarı oranı DVM yöntemine göre %12.2, YSA yöntemine göre %16.1 daha kötüdür. Dairesel ve Tek Yönlü Sektörel sınıfı için kesinlik değeri en düşük değerlerdir. Bu durum Dairesel veya Tek Yönlü Sektörel olarak sınıflandırılan örneğin gerçekte bu sınıfa ait olma olasılığının düşük olduğunu gösterir.

**Karşılaştırma Matrisi**

1	1623	28	42	0	0	0	0	0	0	%95.9
2	256	1463	175	317	195	0	0	0	0	%60.8
3	65	138	965	284	322	0	0	0	317	%46.2
4	56	214	261	920	243	0	0	0	451	%42.9
5	0	74	418	405	1137	0	0	0	154	%52.0
6	0	0	0	0	0	1896	63	154	0	%89.7
7	0	83	0	0	36	104	1937	0	0	%89.7
8	0	0	0	0	0	0	0	1846	0	%100
9	0	0	139	74	67	0	0	0	1078	%79.4
	%81.2	%73.2	%48.2	%46.0	%56.9	%94.8	%96.9	%92.3	%53.9	%71.5
	1	2	3	4	5	6	7	8	9	
	Hedef Sınıfı									

Şekil 5.7 İÖÇ yöntemi ile elde edilen özneliklerin K-EYK ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}

Şekil 5.8'de İÖÇ yöntemi ile elde edilen özniteliklerin HKS kullanılarak sınıflandırma sonucu gösterilmiştir. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel{5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9} ATT'ler için sınıflandırma başarıları sırasıyla % 82.0 % 72.7, % 83.7, % 78.7, % 75.4, % 77.6, % 88.8, % 78.5 ve % 77.8 'dir. HKS yönteminde Raster, Tek Yönlü Sektörel, Dairesel ve Çift Yönlü Sektörel tipleri arasında hatalı sınıflandırılmış örnekler vardır. Fakat hatalı sınıflandırma oranı K-EYK yöntemine göre daha azdır. Genel başarı oranına baktığımızda HKS yönteminin başarı oranı K-EYK yöntemine göre %8 daha iyi iken, YSA yöntemine göre %8.1 daha kötüdür.

**Karşılaştırma Matrisi**

Çıktı Sınıfı	1	1653	139	29	0	0	0	0	0	0	%90.8
	2	235	1453	37	27	114	0	0	0	0	%77.9
	3	64	16	1674	228	28	0	0	52	122	%76.6
	4	16	183	65	1574	243	0	0	115	192	%65.9
	5	0	113	168	12	1508	0	0	0	129	%78.1
	6	0	0	0	0	0	1553	225	263	0	%76.1
	7	0	96	0	0	48	205	1775	0	0	%83.6
	8	32	0	0	0	0	163	0	1570	0	%88.2
	9	0	0	27	159	59	79	0	0	1557	%82.8
		%82.0	%72.7	%83.7	%78.7	%75.4	%77.6	%88.8	%78.5	%77.8	%79.5
		1	2	3	4	5	6	7	8	9	
		Hedef Sınıfı									

Şekil 5.8 İÖÇ yöntemi ile elde edilen özniteliklerin HKS ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.

Şekil 5.9'da ESAOK-ÖÇ yöntemi ile elde edilen özniteliklerin Yapay Sinir Ağı (YSA) kullanılarak sınıflandırma sonucu gösterilmiştir. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel{5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9} ATT'ler için sınıflandırma başarıları sırasıyla %94.0 %90.6, %86.2, %100, %86.6, %90.6, %93.8, %92.0 ve %100'dür. Tek Yönlü Sektörel tarama örnekleri Çift Yönlü Sektörel ve Konik olarak sınıflandırılmıştır. Bazı Çift Yönlü Sektörel tarama örnekleri Dairesel olarak sınıflandırılmıştır. Bu hatalı sınıflandırmaların sebebi ESAOK-ÖÇ yönteminin girdisi olan zaman-frekans görüntüsündeki bozulmalardır. Dairesel sınıfının doğruluk değeri %100 olmasına rağmen kesinlik değeri %77.2'dir. Bu durum, sınıflandırma modelinin Dairesel sınıfı için performansının iyi olmadığını göstermektedir.

**Karşılaştırma Matrisi**

1	1889	16	0	0	0	25	0	0	0	%97.9
2	14	1813	8	0	0	0	123	0	0	%92.6
3	25	24	1724	0	0	0	0	159	0	%89.2
4	47	103	0	2000	268	163	0	0	0	%77.2
5	13	44	110	0	1732	0	0	0	0	%91.2
6	0	0	158	0	0	1812	0	0	0	%92.0
7	12	0	0	0	0	0	1877	0	0	%99.4
8	0	0	0	0	0	0	0	1841	0	%100
9	0	0	0	0	0	0	0	0	2000	%100
	%94.0	%90.6	%86.2	%100	%86.6	%90.6	%93.8	%92.0	%100	%92.7
	1	2	3	4	5	6	7	8	9	
	Hedef Sınıfı									

Şekil 5.9 ESAOK-ÖÇ yöntemi ile elde edilen özniteliklerin YSA ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.

Şekil 5.10'da ESAOK-ÖÇ yöntemi ile elde edilen özneliklerin DVM kullanılarak sınıflandırma sonucu gösterilmiştir. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel{5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9} ATT'ler için sınıflandırma başarıları sırasıyla %95.7 %96.2, %92, %100, %87.2, %98.5, %98.8, %97.5 ve %97.4'dür. Çift Yönlü Sektörel ATT için, EH sisteminin almacı tarama sektörünün ortasındaysa, alınan anten modeli Dairesel ATT ile sadece huzme genişliğinden ayrılabilir. Bu nedenle, bazı Çift Yönlü Sektörel ATT'ler, Dairesel ATT olarak hatalı sınıflandırılabilir. Sürekli aydınlatmalı tipler ile taramalı aydınlatmalı tipler arasındaki hatalı sınıflandırma azalmıştır. Bunun sebebi DVM yönteminin ilk olarak sürekli ve taramalı aydınlatma tiplerini ayırmasıdır. Genel başarı oranlarına baktığımızda DVM yönteminin başarı oranı YSA yönteminden %3.2 daha iyidir.

**Karşılaştırma Matrisi**

1	1914	77	0	0	0	0	0	53	%94.1	
2	82	1924	1	0	12	0	0	0	%95.3	
3	0	6	1839	0	0	0	49	0	%97.1	
4	0	0	23	2000	243	0	0	0	%88.3	
5	4	3	102	0	1745	0	0	0	%94.1	
6	0	0	0	0	0	1969	24	0	%98.8	
7	0	0	0	0	0	31	1976	0	%98.5	
8	0	0	35	0	0	0	0	1951	%98.2	
9	0	0	0	0	0	0	0	0	1947	%100
	%95.7	%96.2	%92.0	%100	%87.2	%98.5	%98.8	%97.5	%97.4	%95.9
	1	2	3	4	5	6	7	8	9	
	Hedef Sınıfı									

Şekil 5.10 ESAOK-ÖÇ yöntemi ile elde edilen özneliklerin DVM ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.

Şekil 5.11'de ESAOK-ÖÇ yöntemi ile elde edilen özneliklerin K-EYK kullanılarak sınıflandırma sonucu gösterilmiştir. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel{5}, Konik{6}, Spiral{7}, Lob Anahtarlama{8}, Kesik Konik{9} ATT'ler için sınıflandırma başarıları sırasıyla %68.2 %80.9, %63.8, %79.3, %68.3, %62.8, %71.7, %76.6 ve %85.5'tür. Bazı Çift Yönlü Sektörel ATT örnekleri Tek Yönlü Sektörel ATT olarak sınıflandırılmıştır. Bu nedenle, K-EYK algoritmasının Çift yönlü Sektörel ATT'deki aydınlatmalar arası zaman farkını ayıramamasından dolayı Tek Yönlü Sektörel ATT olarak hatalı sınıflandırılmıştır. Genel başarı oranlarına baktığımızda K-EYK yönteminin başarı oranı DVM yönteminden %23.6 daha kötüdür.

**Karşılaştırma Matrisi**

1	1364	0	0	31	39	64	0	0	0	%91.1
2	59	1618	132	76	22	16	49	304	0	%71.1
3	48	0	1276	34	367	35	22	0	0	%71.6
4	214	0	168	1587	45	264	311	77	0	%59.5
5	0	74	194	191	1366	219	0	0	291	%58.5
6	0	36	0	44	0	1257	0	0	0	%94.0
7	52	28	73	0	74	0	1434	0	0	%86.3
8	134	0	0	0	0	0	0	1533	0	%92.0
9	129	244	157	37	87	145	184	86	1709	%61.5
	%68.2	%80.9	%63.8	%79.3	%68.3	%62.8	%71.7	%76.6	%85.5	%73.0
	1	2	3	4	5	6	7	8	9	
	Hedef Sınıfı									

Şekil 5.11 ESAOK-ÖÇ yöntemi ile elde edilen özneliklerin K-EYK ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlama{8}, Kesik Konik{9}.

Şekil 5.12'de ESAOK-ÖÇ yöntemi ile elde edilen özniteliklerin HKS kullanılarak sınıflandırma sonucu gösterilmiştir. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel{5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9} ATT'ler için sınıflandırma başarıları sırasıyla %82.2 %85.8, %86.7, %79, %84.2, %90.7, %78.1, %83.4 ve %81.7'dir. Bazı Lob Anahtarlamalı ATT örnekleri Raster ATT olarak, Helezonik ATT örnekleri Dairesel ATT olarak hatalı sınıflandırılmıştır. Bu durum HKS yönteminin bazı sınıflar için çok verimli olmadığını göstermektedir. Genel başarı oranına baktığımızda HKS yönteminin başarı oranı K-EYK yöntemine göre %10.5 daha iyi iken, DVM yöntemine göre %13.1 daha kötüdür.

**Karşılaştırma Matrisi**

Çıktı Sınıfı	1	1645	0	0	108	29	0	91	0	0	%87.8
	2	0	1715	27	0	0	0	0	238	0	%86.6
	3	62	0	1733	96	146	38	347	0	0	%71.6
	4	189	53	37	1579	79	0	0	0	69	%78.7
	5	0	0	116	143	1683	47	0	0	297	%73.6
	6	48	0	0	74	25	1814	0	0	0	%92.5
	7	56	94	87	0	38	74	1562	94	0	%77.9
	8	0	138	0	0	0	27	0	1668	0	%91.0
	9	0	0	0	0	0	0	0	0	1634	%100
		%82.2	%85.8	%86.7	%79.0	%84.2	%90.7	%78.1	%83.4	%81.7	%83.5
		1	2	3	4	5	6	7	8	9	
		Hedef Sınıfı									

Şekil 5.12 ESAOK-ÖÇ yöntemi ile elde edilen özniteliklerin HKS ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.

Şekil 5.13'te UKSBOK-ÖÇ yöntemi ile elde edilen özniteliklerin YSA kullanılarak sınıflandırma sonucu gösterilmiştir. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel{5}, Konik{6}, Spiral{7}, Lob Anahtarlama{8}, Kesik Konik{9} ATT'ler için sınıflandırma başarıları sırasıyla %94 %92.2, %99.1, %100, %99.7, %100, %99, %100 ve %100'dür. Radar anteni huzmesi EH sistemini yeteri kadar süre aydınlatmadığı durumlarda Raster ve Helezonik ATT'lerin tüm aydınlatmaları alınamayabilir. Bu durum hatalı sınıflandırmaya yol açılabilir. Örneğin, Raster ATT yalnızca bir aydınlatması alındığında Dairesel olarak sınıflandırılabilir. Benzer şekilde, Helezonik ATT'nin tüm aydınlatmaları alınmadığı durumda Raster veya Dairesel olarak sınıflandırılabilir.

**Karşılaştırma Matrisi**

1	1881	48	0	0	0	0	21	0	0	%96.5
2	58	1843	4	0	0	0	0	0	0	%96.7
3	4	6	1981	0	0	0	0	0	0	%99.5
4	45	95	0	2000	6	0	0	0	0	%93.2
5	6	8	15	0	1994	0	0	0	0	%98.6
6	0	0	0	0	0	2000	0	0	0	%100
7	6	0	0	0	0	0	1979	0	0	%99.7
8	0	0	0	0	0	0	0	2000	0	%100
9	0	0	0	0	0	0	0	0	2000	%100
	%94.0	%92.2	%99.1	%100	%99.7	%100	%99.0	%100	%100	%98.2
	1	2	3	4	5	6	7	8	9	
	Hedef Sınıfı									

Şekil 5.13 UKSBOK-ÖÇ yöntemi ile elde edilen özniteliklerin YSA ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlama{8}, Kesik Konik{9}.

Şekil 5.14'te UKSBOK-ÖÇ yöntemi ile elde edilen özneliklerin DVM kullanılarak sınıflandırma sonucu gösterilmiştir. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel{5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9} ATT'ler için sınıflandırma başarıları sırasıyla %95.0 %95.7, %87.1, %100, %99.7, %99.7, %99.4, %100 ve %100'dür. DVM yönteminde bazı Tek Yönlü Sektörel ATT'lerin, Çift Yönlü Sektörel ATT olarak hatalı sınıflandırıldığı görülmektedir. Bunun sebebi EH sistemi almacının Çift Yönlü Sektörel taramanın tarama bölgesinin ortasına yakın konumda olmasından dolayı Tek Yönlü Sektörel taramaya benzer tarama örüntüsü oluşması ve DVM yönteminin bu iki taramayı ayıracak kadar duyarlı olmamasından dolayı olabilir. Genel başarı oranlarına baktığımızda YSA yönteminin başarı oranı DVM yönteminden % 0.8 daha iyidir.

Çıktı Sınıfı	1	2	3	4	5	6	7	8	9	
1	1900	68	0	0	0	0	0	0	0	%96.5
2	88	1914	1	0	0	0	0	0	0	%95.6
3	0	6	1741	0	0	0	0	0	0	%99.7
4	6	0	0	2000	6	0	0	0	0	%99.4
5	6	12	258	0	1994	0	0	0	0	%87.8
6	0	0	0	0	0	1994	12	0	0	%99.4
7	0	0	0	0	0	6	1988	0	0	%99.7
8	0	0	0	0	0	0	0	2000	0	%100
9	0	0	0	0	0	0	0	0	2000	%100
	%95.0	%95.7	%87.1	%100	%99.7	%99.7	%99.4	%100	%100	%97.4
	1	2	3	4	5	6	7	8	9	
	Hedef Sınıfı									

Şekil 5.14 UKSBOK-ÖÇ yöntemi ile elde edilen özneliklerin DVM ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.



Şekil 5.15'te UKSBOK-ÖÇ yöntemi ile elde edilen özneliklerin K-EYK kullanılarak sınıflandırma sonucu gösterilmiştir. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel{5}, Konik{6}, Spiral{7}, Lob Anahtarlama{8}, Kesik Konik{9} ATT'ler için sınıflandırma başarıları sırasıyla %88.0 %87.6, %95.5, %94.5, %71, %81.1, %100, %91 ve %82.8'dir. K-EYK yönteminden en kötü sınıflandırma performansı Çift Yönlü Sektörel sınıfında görülmüştür. Çift Yönlü Sektörel ATT'ler Dairesel ve Tek Yönlü Sektörel ATT olarak hatalı sınıflandırılmıştır. Genel başarı oranlarına baktığımızda K-EYK yönteminin başarı oranı YSA yönteminden % 10.3 daha kötüdür.

**Karşılaştırma Matrisi**

1	1760	91	0	0	0	0	0	0	0	%95.1
2	0	1753	0	68	0	33	0	0	344	%79.8
3	24	49	1909	0	269	153	0	0	0	%79.4
4	0	0	40	1889	312	0	0	0	0	%84.3
5	0	0	0	0	1419	0	0	0	0	%100
6	0	0	0	0	0	1622	0	181	0	%90.0
7	216	107	0	43	0	0	2000	0	0	%84.5
8	0	0	51	0	0	30	0	1819	0	%95.7
9	0	0	0	0	0	162	0	0	1656	%91.1
	%88.0	%87.6	%95.5	%94.5	%71.0	%81.1	%100	%91.0	%82.8	%87.9
	1	2	3	4	5	6	7	8	9	
	Hedef Sınıfı									

Şekil 5.15 UKSBOK-ÖÇ yöntemi ile elde edilen özneliklerin K-EYK ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlama{8}, Kesik Konik{9}.

Şekil 5.16'da UKSBOK-ÖÇ yöntemi ile elde edilen özneliklerin HKS kullanılarak sınıflandırma sonucu gösterilmiştir. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel{5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9} ATT'ler için sınıflandırma başarıları sırasıyla %88.3 %84.4, %97.2, %100, %90.8, %95.6, %100, %93.8 ve %100'dür. K-EYK yönteminden en kötü sınıflandırma performansı Raster ve Helezonik sınıfında görülmüştür. Bazı Raster ATT ve Helezonik ATT'ler birbirleri arasında hatalı sınıflandırılmıştır. Genel başarı oranlarına baktığımızda HKS yönteminin başarı oranı K-EYK yönteminden % 6.6 daha iyi, YSA yönteminden % 3.7 daha kötüdür.

**Karşılaştırma Matrisi**

1	1767	270	0	0	0	0	0	0	0	%86.7
2	204	1688	57	0	0	0	0	0	0	%86.6
3	29	42	1943	0	184	0	0	0	0	%88.4
4	0	0	0	2000	0	0	0	0	0	%100
5	0	0	0	0	1816	0	0	0	0	%100
6	0	0	0	0	0	1912	0	123	0	%94.0
7	0	0	0	0	0	0	2000	0	0	%100
8	0	0	0	0	0	88	0	1877	0	%95.5
9	0	0	0	0	0	0	0	0	2000	%100
	%88.3	%84.4	%97.2	%100	%90.8	%95.6	%100	%93.8	%100	%94.5
	1	2	3	4	5	6	7	8	9	
	Hedef Sınıfı									

Şekil 5.16 UKSBOK-ÖÇ yöntemi ile elde edilen özneliklerin HKS ile sınıflandırma sonucu. Helezonik{1}, Raster{2}, Tek Yönlü Sektörel{3}, Dairesel{4}, Çift Yönlü Sektörel {5}, Konik{6}, Spiral{7}, Lob Anahtarlamalı{8}, Kesik Konik{9}.

Tablo 5.4. Kullanılan ATT Belirleme yöntemlerinin zamansal performansları.

Yöntem	Başarı Yüzdesi	F Skor	Öznitelik Çıkarma Süresi (ms)	Sınıflandırma Süresi (ms)	Toplam Süre (ms)
İÖÇ+K-EYK	%71.5	%71.5	74.13	35.68	109.81
ESAOK-ÖÇ+K-EYK	%73.0	%73.4	504.33	34.25	538.58
İÖÇ+HKS	%79.5	%79.5	74.13	5.38	79.51
ESAOK-ÖÇ+ HKS	%83.5	%83.7	504.33	5.44	509.77
İÖÇ+DVM	%84.6	%80.6	74.13	61.18	135.31
İÖÇ+YSA	%87.6	%87.8	74.13	0.73	74.86
UKSBOK-ÖÇ+ K-EYK	%87.9	%87.8	12.53	36.05	48.58
ESAOK-ÖÇ+ YSA	%92.7	%92.8	504.33	0.76	505.09
UKSBOK-ÖÇ+ HKS	%94.5	%94.5	12.53	5.65	18.18
ESAOK-ÖÇ+DVM	%95.9	%95.9	504.33	62.35	566.68
UKSBOK-ÖÇ+DVM	%97.4	%97.4	12.53	61.84	74.37
UKSBOK-ÖÇ+ YSA	<b>%98.2</b>	%98.2	12.53	0.78	<b>13.31</b>

Tablo 5.4'te ATT belirleme yöntemlerinin zamansal performansları verilmiştir. Zamansal performanslar 128 örnek üzerinden ortalama süre alınarak hesaplanmıştır. Öznitelik çıkarma süresi en düşük olan 12.53 ms ile UKSBOK-ÖÇ modelidir. Sınıflandırma süresi en düşük olan 0.73 ms ile YSA modelidir. Toplam süre ve başarı yüzdesi incelendiğinde en iyi ATT belirleme yöntemi UKSBOK-ÖÇ modeli ile YSA sınıflandırma yönteminin kullanılmasıdır. Sınıflandırıcı olarak en kötü zamansal performansı 62.35 ms ile DVM yöntemi vermiştir. ESAOK-ÖÇ modeli ile öznitelik çıkarma süresi 504.33 ms ile diğer modellere göre en fazla zamanı almıştır. Bunun nedeni modelin girdisi olan zaman-frekans görüntüsünün oluşturulmasının zaman alan bir işlem olmasıdır. Karşılaştırma matrislerinde verilen kesinlik ve duyarlılık değerlerinin harmonik ortalaması F skoru verir. Bütün yöntemler için ortalama F skor değerleri ile başarı yüzdeleri benzerdir.

## 6. SONUÇLAR

Bu çalışmada, EH sistemlerinde kullanılmak üzere Anten Tarama Tipi (ATT) belirleme işlemini otomatik hale getiren algoritma tasarımları yapılmış ve bunlar karşılaştırılmıştır. Öznitelik çıkarma yöntemi olarak otomatik kodlayıcılar ve istatistiksel yöntemler kullanılarak öznitelik vektörleri çıkarılmıştır. Bu öznitelik vektörleri, ATT belirleme işlemi için gözetimli ve gözetimsiz öğrenme yöntemlerinde kullanılmıştır. Farklı mesafe ölçütleriyle yapılan testlerde en iyi sonucu %96,8 ile Uzun-Kısa Süreli Bellek Tabanlı Otomatik Kodlayıcılar ile Öznitelik Çıkarma (UKSBOK-ÖÇ) yöntemi kullanılarak çıkarılan öznitelik vektörünün Mahalanobis uzaklık ölçütü kullanılan hiyerarşik kümeleme algoritmasıyla kümelenebilmesi vermiştir. K-Ortalamlar, K-Ortalamlar++ ve Hiyerarşik Kümeleme algoritmaları farklı mesafe ölçütlerinde karşılaştırılmıştır. Hiyerarşik kümeleme algoritmasının performansı, tüm mesafe ölçümleri için en iyi sonucu vermiştir. Kümeleme analizi sonucu Hiyerarşik Kümeleme tabanlı Sınıflandırma (HKS) yöntemi geliştirilmiştir.

Yapay Sinir Ağı (YSA), Destek Vektör Makinesi (DVM), Hiyerarşik Kümeleme Tabanlı Sınıflandırma (HKS) ve K-En Yakın Komşu (K-EYK) algoritmaları ile yapılan testlerde YSA yönteminin başarımlarının K-EYK ve HKS yöntemlerine göre bütün öznitelik çıkarma yöntemleri için daha iyi olduğu görülmüştür. Bunun sebebi olarak YSA yönteminin özniteliklerdeki küçük değişiklikleri fark etmesi ve değişime karşı duyarlılığının daha yüksek olmasıdır. HKS yönteminin avantajı gerçek sahadan alınan veriler üstünde uygulanabilir olmasıdır. Çünkü EH alanında etiketlenmiş gerçek veri bulmak çok zor ve zahmetli bir işlemdir. HKS yöntemi kümeleme tabanlı bir yöntem olduğundan gerçek veri üzerinde kümeleme işlemi yapılarak elde edilen küme elemanlarının bazıları incelenerek küme etiketlenebilir. DVM modelinde ise sürekli aydınlatmalı ve taramalı aydınlatmalı anten tarama tipleri çok sınıfta sınıflandırma yapılmadan daha üst bir sınıfta ayrıldığı için bu tiplerin birbirleriyle karışması daha zor olmuştur. Bu nedenle Evrimsel Sinir Ağı Tabanlı Otomatik Kodlayıcılar ile Öznitelik Çıkarma (ESAOK-ÖÇ) yöntemi kullanılarak çıkarılan öznitelik vektörlerinin DVM yöntemi ile sınıflandırılması sonucu elde edilen başarımların, YSA yöntemi kullanılarak elde edilen başarımlarından daha iyidir. Diğer öznitelik çıkarma modellerinde YSA yönteminin başarımları daha iyi olmasına rağmen fark çok azdır. Başarımların farkının az olmasına rağmen, YSA yöntemi DVM yönteminden 80 kat daha hızlıdır. İÖÇ ve ESAOK-ÖÇ yöntemleriyle çıkarılan özniteliklerin YSA, DVM, K-EYK ve HKS yöntemleriyle sınıflandırılması sonucunda

Dairesel sınıfın kesinlik değeri en düşük değerdir. Bunun sebebi, İÖÇ ve ESAOK-ÖÇ modellerinin öznitelik çıkarma performanslarının iyi olmamasından kaynaklanmaktadır. Sınıflandırıcılar karar veremedikleri örnekleri en genel tarama örüntüsü olan Dairesel tarama tipine atamışlardır. Başarım yüzdesi ve ortalama F skor değerleri en yüksek hesaplanan model UKSBOK-ÖÇ yöntemiyle öznitelik çıkarılarak YSA yöntemiyle ATT'nin sınıflandırılması modelidir. Bu çalışma sonucunda UKSBOK-ÖÇ modeliyle öznitelik çıkarılarak YSA yöntemiyle ATT'nin sınıflandırılmasının en iyi yöntem olduğu anlaşılmıştır. Gerçek zamanlı veriler ile çalışılması durumunda HKS yönteminin diğer yöntemlere göre daha başarılı olacağı düşünülmektedir.

## 7. KAYNAKLAR

- [1] S. Robertson, Practical ESM Analysis. Artech House, 2019.
- [2] F. Altıparmak, F. Ç. Akyön, E. Özmen, F. Çoğun and A. Bayri, "Towards Cognitive Sensing: Radar Function Classification using Multitask Learning," 2019 27th Signal Processing and Communications Applications Conference (SIU), 2019, pp. 1-4.
- [3] B. Barshan and B. Eravci, "Automatic Radar Antenna Scan Type Recognition in Electronic Warfare," in IEEE Transactions on Aerospace and Electronic Systems, vol. 48, no. 4, pp. 2908-2931, October 2012.
- [4] Y.-H. Kim, K.-H. Song, J.-W. Han, and H.-N. Kim, "Radar scan pattern analysis for reduction of false identification in electronic warfare support systems," IET Radar, Sonar & Navigation, vol. 8, no. 7, pp. 719–728, 2014.
- [5] S. Ayazgök, C. Erdem, M. T. Öztürk, A. Orduyılmaz, and M. Serin, "Automatic antenna scan type classification for next-generation electronic warfare receivers," IET Radar, Sonar & Navigation, vol. 12, no. 4, pp. 466–474, 2018.
- [6] E. Özmen, A. G. Aydın, H. S. Balaban and A. K. Kamçı, "Antenna Scanning Type Classification using Machine Learning," 2020 28th Signal Processing and Communications Applications Conference (SIU), 2020, pp. 1-4.
- [7] D. Adamy, EW 101: A first course in electronic warfare. Artech House, 2001.
- [8] R. G. Wiley, ELINT: The interception and analysis of radar signals. Artech House, 2006.
- [9] D. Adamy, Introduction to Electronic Warfare Modeling and Simulation, Scitech Pub Inc, 2006.
- [10] F. Neri, Introduction to electronic defense systems. SciTech Publishing, 2006.
- [11] D. Adamy, EW 102: A second course in electronic warfare. Artech House, 2004.
- [12] Sejdić, Ervin, Igor Djurović, and Jin Jiang. "Time–frequency feature representation using energy concentration: An overview of recent advances." Digital signal processing 19.1 (2009): 153-183.
- [13] S. G. Johnson and M. Frigo, "A modified split-radix fft with fewer arithmetic operations," IEEE Transactions on Signal Processing, vol. 55, no. 1, pp. 111–119, 2006.

- [14] B. Esmael, A. Arnaout, R. K. Fruhwirth, and G. Thonhauser, "A statistical feature-based approach for operations recognition in drilling time series," vol. 5, pp. 454–61.
- [15] T. Liu, S. Yu, B. Xu, and H. Yin, "Recurrent networks with attention and convolutional networks for sentence representation and classification," *Applied Intelligence*, vol. 48, no. 10, pp. 3797–3806, 2018.
- [16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [17] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.
- [18] Renton, Guillaume, et al. "Fully convolutional network with dilated convolutions for handwritten text line segmentation." *International Journal on Document Analysis and Recognition (IJ DAR)* 21.3 (2018): 177-186.
- [19] Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: *CVPR*, pp. 3431–3440 (2015).
- [20] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016, vol. 1.
- [21] Chen, Min, et al. "Deep features learning for medical image analysis with convolutional autoencoder neural network." *IEEE Transactions on Big Data* (2017).
- [22] Gondara, Lovedeep. "Medical image denoising using convolutional denoising autoencoders." *2016 IEEE 16th international conference on data mining workshops (ICDMW)*. IEEE, 2016.
- [23] Lloyd, Stuart. "Least squares quantization in PCM." *IEEE transactions on information theory* 28.2 (1982): 129-137.
- [24] R. Ostrovsky, Y. Rabani, L. J. Schulman, and C. Swamy. The effectiveness of Lloyd-type methods for the k-means problem. In *FOCS*, pages 165-176, 2006.
- [25] Vassilvitskii, Sergei, and David Arthur. "k-means++: The advantages of careful seeding." *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. 2006.
- [26] Bahmani, Bahman, et al. "Scalable k-means++." *arXiv preprint arXiv:1203.6402* (2012).

- [27] Lee, Sang Su, Dongwoo Won, and Dennis McLeod. "Discovering Relationships among Tags and Geotags." ICWSM. 2008.
- [28] Forina, M., C. Armanino, and V. Raggio. "Clustering with dendrograms on interpretation variables." *Analytica Chimica Acta* 454.1 (2002): 13-19.
- [29] Zhao, Haifeng, and Zijie Qi. "Hierarchical agglomerative clustering with ordering constraints." 2010 Third International Conference on Knowledge Discovery and Data Mining. IEEE, 2010.
- [30] Chomboon, Kittipong, et al. "An empirical study of distance metrics for k-nearest neighbor algorithm." Proceedings of the 3rd international conference on industrial application engineering. 2015.
- [31] Somerfield, Paul J. "Identification of the Bray-Curtis similarity index: Comment on Yoshioka (2008)." *Marine Ecology Progress Series* 372 (2008): 303-306.
- [32] Singh, Archana, Avantika Yadav, and Ajay Rana. "K-means with Three different Distance Metrics." *International Journal of Computer Applications* 67.10 (2013).
- [33] Mayoraz, Eddy, and Ethem Alpaydın. "Support vector machines for multi-class classification." *International Work-Conference on Artificial Neural Networks*. Springer, Berlin, Heidelberg, 1999.
- [34] Franc, Vojtech, and Václav Hlavác. "Multi-class support vector machine." Object recognition supported by user interaction for service robots. Vol. 2. IEEE, 2002.
- [35] Pedregosa, Fabian, et al. "Scikit-learn: Machine learning in Python." *the Journal of machine Learning research* 12 (2011): 2825-2830.
- [36] Ali, Munwar, et al. "Semantic-k-NN algorithm: an enhanced version of traditional k-NN algorithm." *Expert Systems with Applications* 151 (2020): 113374.
- [37] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).
- [38] Abadi, Martín, et al. "Tensorflow: Large-scale machine learning on heterogeneous distributed systems." arXiv preprint arXiv:1603.04467 (2016).
- [39] Van der Maaten, Laurens, and Geoffrey Hinton. "Visualizing data using t-SNE." *Journal of machine learning research* 9.11 (2008).



## EKLER

### EK 1 - Tezden Türetilmiş Yayınlar

- Tezden türetilmiş olan '**Determination of Antenna Scan Type in Electronic Warfare Support Systems via Machine Learning**' başlıklı yayın, *IEEE Transactions on Aerospace and Electronic Systems* dergisine başvurulmak üzere gönderilecektir.

### EK 2 - Tezden Türetilmiş Bildiriler

- **Özmen, E., & Özkazanç, Y. (2021, June). Antenna Scanning Type Classification with Autoencoder Based Deep Clustering.** In 2021 29th Signal Processing and Communications Applications Conference (SIU) (pp. 1-4). IEEE.
- Tezden türetilmiş olan '**DeepASTC:Antenna Scan Type Classification using Deep Learning**' başlıklı yayın, *IEEE 2022 Radar Konferansına* gönderildi.

# Otomatik Kodlayıcı Tabanlı Derin Kümeleme ile Anten Tarama Tipi Sınıflandırma Antenna Scanning Type Classification with Autoencoder Based Deep Clustering

Emirhan Özmen<sup>[1][2]</sup>, Yakup Özkazanç<sup>[2]</sup>

[1]Radar, Elektronik Harp ve İstihbarat Sistemleri, ASELSAN A.Ş., Ankara, Türkiye

[2]Elektrik ve Elektronik Mühendisliği Bölümü, Hacettepe Üniversitesi, Ankara, Türkiye

eozen@aselsan.com.tr, yakup@ee.hacettepe.edu.tr

**Özetçe** —Bu çalışmada Elektronik Harp (EH) sistemlerinde kullanılan anten tarama tiplerinin otomatik bulunması için derin öğrenme tabanlı bir algoritma önerilmiştir. Önerilen algoritma literatürdeki diğer algoritmalara göre daha iyi sonuçlar göstermiştir.

**Anahtar Kelimeler**—Anten Tarama Tipi Sınıflandırma, Derin öğrenme, Derin Kümeleme, Elektronik Harp.

**Abstract**—In this study, a deep learning-based algorithm is proposed for automatic detection of antenna scanning types used in EW systems. The proposed algorithm has shown better results than other algorithms in the literature.

**Keywords**—Antenna Scan Type Classification, Deep Learning, Deep Clustering, Electronic Warfare.

## I. GİRİŞ

EH sistemlerinde bir radarın anten tarama örüntüsünü sınıflandırmış olmak radarın doğru kimliklendirilebilmesi ve radar karşısında karıştırma işlevi yapılabilmesi açısından çok önemlidir. Radarın fonksiyonu (arama, aydınlatma, takip, vb.) ve tehdit seviyesi anten tarama tipinden (ATT) çıkarılabilmektedir [1]. Aynı zamanda ATT, EH sistemlerinde radarların kimliklendirilebilmesinde kullanılan en önemli parametrelerden bir tanesidir. Bu özelliklere sahip olması ATT analizini Elektronik Harp (EH) sistemlerinin en önemli parçalarından biri haline getirmektedir. Buna karşın literatürde ATT analizi ile ilgili yayınlar oldukça kısıtlıdır. Son zamanlarda bu alanda yapılmış kapsamlı çalışmalardan biri de [2]'dir. Bu çalışmada ön veri işleme yöntemleri uygulandıktan sonra radar sinyalinin bazı özellikleri çıkarılarak örüntü tanıma tabanlı algoritmalar yardımı ile ATT sınıflandırması yapılmıştır. Beş temel ATT (daireysel, raster, helezonik ve konik) için dört geleneksel sınıflandırıcının (Naive Bayes Sınıflandırıcı, Karar Ağacı Sınıflandırıcı, Yapay Sinir Ağı Tabanlı Sınıflandırıcı, Destek Vektör Makinesi) sınıflandırma performansları karşılaştırılmıştır. Literatürdeki daha sonra yapılan çalışmalar [2]'yi temel almıştır. [3]'te ek olarak radarların hatalı kimliklendirilmesini önlemek amacıyla 'Tanımlanamayan' tipinde yeni bir

anten tarama sınıfı önerilmiştir. [4]'de [2]'deki yönteme ek olarak Spiral ve Elektronik tarama tipleri sınıflandırmaya dahil edilmiştir. Ek olarak elektronik taramaları ayırt etmek amacıyla korelasyon temelli yeni bir ana huzme belirleme yöntemi önerilmiştir. Bu konuda yapılan en güncel çalışma [5]'tir. Bu çalışmada sinyal tepe noktasını bulmak için yeni bir algoritma önerilmiş ve oluşturulan öznelikler çok katmanlı algılayıcı tabanlı bir sınıflandırıcı kullanılarak sınıflandırılmıştır.

Radar sistemlerinde genellikle dar huzmeye sahip antenler kullanılır. Bunun sebebi, uzaysal çözünürlük ve algılama mesafesini arttırmaktır. Gözetleme alanındaki hedefleri tespit edip konumlandırabilmek için radarın ana huzmesinin hareketi gereklidir. Bu ana huzme hareketine tarama, oluşan örüntüye de anten tarama örüntüsü denir. Radar anteninin hareketi mekanik ya da elektronik olabilir. Bu bildiride kullanılacak olan verinin içerdiği tarama tipleri şöyledir;

**Dairesel Tarama** Arama radarlarında kullanılan en yaygın Anten Tarama Tipidir. Radar anteni sabit hızla yarıda 360 dereceyi tarar. Yükselişte bakılan açı sabittir, herhangi bir tarama yapılmaz.

**Bisektörel Tarama** İlgili alanı belirli bir yarı açı aralığı olan arama radarları tarafından kullanılan anten tarama tipidir. Radar anteni sabit hızla yarıda ilgili sektör açı aralığını tarar. Tarama sektörünün sonuna geldiğinde radar anteni tarama yönünü tersine çevirerek tekrar tarama yapar. Yükselişte bakılan açı sabittir, herhangi bir tarama yapılmaz.

**Unisektörel Tarama** Bisektör tarama tipinden farklı olarak tarama sektörünün sonuna geldiğinde radar anteni tarama yönünü tersine çevirerek tekrar tarama yapmaz. Aynı sektöre geldiğinde yine tarama yapar. Yükselişte bakılan açı sabittir, herhangi bir tarama yapılmaz.

**Raster Tarama** İlgili alanı belirli bir yarı açı ve yükseliş açı aralığı olan arama radarları tarafından kullanılan anten tarama tipidir. Radar anteni sabit hızla yarıda ilgili sektör açı aralığını tarar. Tarama sektörünün sonuna geldiğinde radar anteni yükseliş açısını değiştirerek tekrar tarama yapar. İlgili yarı açı ve yükseliş açı aralıkları boşluk bırakmadan satır satır taranır.

*Helezonik Tarama* Arama radarları tarafından kullanılan bir anten tarama tipidir. Radar anteni sabit hızla yarıda 360 dereceyi tararken yükselişte de açısını sürekli değiştirerek belirli bir yükseliş açısı aralığını tarar.

*Konik Tarama* Hedef takip radarlarında kullanılan en yaygın anten tarama tipidir. Radar anteni huzmesini hedef etrafında dairesel hareketler ile gezdirerek hedef takibi yapar.

*Spiral Tarama* Hedef takip radarlarında kullanılan anten tarama tipidir. Radar anteni huzmesini hedef etrafında spiral hareketler ile gezdirerek hedef takibi yapar.

*Lobing Tarama* Lobing taraması, bir izleme radarının izleme doğruluğunu iyileştirmek için anten huzmelerini farklı açılar ile konumlandırır. Konik taramaya benzer, ancak sürekli dönen bir huzme yerine farklı pozisyonlar vardır. Tipik konfigürasyonlar iki veya dört lobdan oluşur.

*Kesik Konik Tarama* Elektronik Harp sistemine gelen bazı darbelerin EH sisteminde duyarlılıktan dolayı konik tarama tipi örüntüsünde bozulmalar oluşur. Oluşan yeni örüntü tipine Kesik Konik anten tarama tipi denilmiştir.

Bu çalışmada otomatik kodlayıcılar kullanılarak anten tarama örüntüsünden öznelikler çıkarılacaktır. Bu öznelikler farklı kümeleme yöntemleriyle sınıflandırılarak sınıflandırma sonuçları karşılaştırılmıştır. Bölüm II de veri üzerinde yapılan ön işleme adımları, Bölüm III'te önerilen yöntemin detaylı açıklaması, Bölüm IV'te benzetim sonuçlarının değerlendirilmesi, Bölüm V'te ise çalışmanın sonuçları hakkında bilgi verilmiştir.

## II. ÖN İŞLEME VE VERİ SETİ HAZIRLAMA

Her bir anten tarama örneğindeki darbeler genlik değeri ile darbe geliş zamanı bilgisini içermektedir. Her bir örnek, eşit aydınlatma sayısını içerecek kadar darbe biriktirilmesi gerekmektedir. Bu çalışma için seçilen aydınlatma sayısı on olarak belirlenmiş her bir tarama örneği için on aydınlatma darbe dizileri sentetik olarak üretilmiştir. Üretilen her bir tarama örneğinin darbe geliş zamanları arasındaki fark, farklı olabilmektedir. Bu yüzden aydınlatma sayısı sabit tutulduğunda her bir tarama örneğindeki darbe sayısı farklı olabilir. Öznelik çıkartmak için kullanacağımız yöntem için her bir taramada eşit sayıda darbe gerektiğinden dolayı yeniden örnekleme yapılarak her bir tarama örneği için darbe sayıları eşitlenmiştir. Anten tarama örneğinde bazı yabancı darbeler veya aydınlatmalar bulunabilir. Bu yabancı darbe ve aydınlatmaların ilk on aydınlatma seçilmeden önce elenmesi gerekmektedir. Bu eleme işlemi üç aşamadan oluşmaktadır;

- Belirli bir darbe sayısından daha az darbe bulunan aydınlatmaları silinmesi.
- Her bir aydınlatma içinde yayının maksimum genlik değerinden yüksek gelen darbeleri silinmesi.
- 1. adımın tekrarlanması.

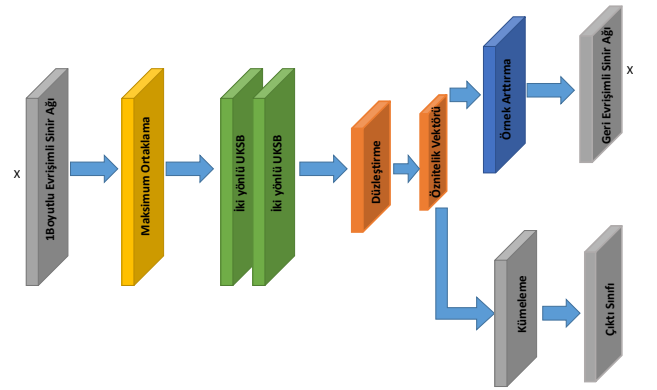
Eleme işlemi yapıldıktan sonra her bir tarama örneği için on aydınlatma seçilmiştir. Her bir tarama örüntüsü örneğinde 1000 darbe örneği vardır ve bu darbe örneklerinin her biri genlik ve darbe geliş zamanı bilgilerinden oluşmaktadır.

## III. ÖNERİLEN OTOMATİK KODLAYICI TABANLI ANTEN TARAMA TİPİ SINIFLANDIRMA YÖNTEMİ

Bu kısımda önerilen anten tarama tipi sınıflandırma yöntemi detaylı olarak anlatılacaktır. Önerilen anten tarama tipi sınıflandırma yöntemi Şekil 1'de gösterilmiştir. İlk olarak otomatik kodlayıcılar kullanılarak tarama örüntüsünün öznelikleri çıkartılmıştır. Daha sonra K-Ortalama ve K-En Yakın Komşu algoritmaları kullanılarak anten tarama tipleri bulunmuştur.

### A. Otomatik Kodlayıcılar ile Öznelik Çıkartma

Öznelik çıkarmak için önerilen otomatik kodlayıcı yapısı bir boyutlu evrişimli sinir ağları ve iki yönlü uzun kısa süreli bellek yapılarından oluşmaktadır. Orta katmanda üç boyutlu bir öznelik vektörü oluşturulmaktadır. Bu öznelik vektörü kullanılarak girdi verisi tekrar yapılandırılmıştır. Hata fonksiyonu olarak ortalama kareler hata fonksiyonu kullanılmıştır.



Şekil 1: Önerilen Anten Tarama Tipi Sınıflandırma Yöntemi

*1-Boyutlu Evrişimli Sinir Ağı Katmanı* Evrişimli Sinir Ağları, verideki basit örüntüleri tanımlayarak katmanlar arttıkça daha karmaşık örüntüleri tanımlamak için kullanılır [6]. 1 Boyutlu Evrişimli sinir ağı, veri kümesinin daha kısa bölümlerinden ilginç öznelikler elde etmeyi beklediğinizde ve bulunan özneliğin bölüm içindeki konumunun yüksek alaka düzeyine sahip olmadığı durumlarda çok etkilidir.

*Maksimum ortalama Katmanı* Maksimum ortalama, örnek tabanlı bir ayrıklaştırma işlemidir [7]. Amaç, bir girdi temsilini aşağı örnekleme, boyutsallığını azaltmak ve binlenmiş alt bölgelerdeki özellikler hakkında varsayımların yapılmasına izin vermektir.

*İki yönlü Uzun Kısa Süreli Bellek (UKSB) Katmanı* UKSB ilk olarak Hochreiter ve Schmidhuber [8] tarafından geliştirilmiştir. Bir girdi vektörü  $x = (x_1, x_2, \dots, x_n)$  olarak verilir, burada  $n$  girdi vektörünün uzunluğudur. UKSB, hafıza hücrelerini kontrol etmek için (1)'de de gösterildiği gibi üç kontrol kapısına sahiptir. Bu hafıza hücresi  $c$  ile gösterilmiştir.  $W_{ix}$ ,  $W_{ih}$  ve  $b_i$  sırasıyla girdi ağırlığı, yineleme ağırlığı, eşik değeridir. Unutma kapısı için  $W_{fx}$ ,  $W_{fh}$  ve  $b_f$  sırasıyla girdi ağırlığı, yineleme ağırlığı, eşik değeridir. Çıktı kapısı için  $W_{ox}$ ,  $W_{oh}$  ve  $b_o$  sırasıyla girdi ağırlığı, yineleme ağırlığı, eşik değeridir.  $t - 1$  durumu  $t$  zamanına göre bir önceki durumu gösterir. Hücre Durmu için  $W_{cx}$ ,  $W_{ch}$  ve  $b_c$  sırasıyla girdi ağırlığı, yineleme ağırlığı, eşik değeridir.

$$\begin{aligned}
\text{Girdi Kapısı : } i_t &= \sigma(W_{ix}x_t + W_{ih}h_{t-1} + b_i) \\
\text{Unutma Kapısı : } f_t &= \sigma(W_{fx}x_t + W_{fh}h_{t-1} + b_f) \\
\text{Çıktı Kapısı : } o_t &= \sigma(W_{ox}x_t + W_{oh}h_{t-1} + b_o) \\
\text{Hücre Durumu : } c_t &= f_t * c_{t-1} + i_t * \tanh \\
(W_{cx}x_t + W_{ch}h_{t-1} + b_c) \\
\text{Hücre Çıktısı : } h_t &= o_t * \tanh(c_t)
\end{aligned} \tag{1}$$

İki yönlü UKSB, (2)'de verilen formüle göre hesaplanır.  $x = (x_1, x_2, \dots, x_n)$  girdi vektörünü temsil eder. ileri ve geri gizli vektörler sırasıyla  $h_t = (h_1, h_2, \dots, h_n)$  ve  $h'_t = (h'_1, h'_2, \dots, h'_n)$  olarak gösterilmiştir. Son geri  $h'_t$  ve ileri  $h_t$  çıktı vektörünün birleştirilmesiyle  $y_t = [h_t, h'_t]$  oluşturulur.

$$\begin{aligned}
h_t &= \sigma(W_{hx}x_t + W_{hh}h_{t-1} + b_h) \\
h'_t &= \sigma(W_{hx}x_t + W_{hh}h_{t+1} + b_h) \\
y_t &= W_{yh}h_t + W_{yh'}h'_t + b_y
\end{aligned} \tag{2}$$

**Düzleştirme Katmanı** Evrişimli katman ile tamamen bağlantılı katman arasında bir "Düzleştirme" katmanı vardır. Düzleştirme, iki boyutlu bir öznitelik matrisini, tamamen bağlı bir sinir ağı sınıflandırıcısına beslenebilen bir vektöre dönüştürür.

**Örnek Arttırma ve Geri Evrişimli Sinir Ağı Katmanı** Bu katman türü, bir girdi vektörünün boyutunu arttırmanın öğrenilebilir bir yolunu sunar. Bir öznitelik vektörünü daha yüksek boyutlu bir uzayda görmemiz gerektiğinde bu yöntem her zaman kullanılabilir.

### B. K-Ortalama Algoritması

K-Ortalama algoritması bir Gözetimsiz Öğrenme ve kümeleme algoritmasıdır. K-Ortalama algoritmasındaki K değeri küme sayısını belirler ve bu değeri parametre olarak alması gerekir. K değeri belirlendikten sonra algoritmada rastgele K tane merkez noktası seçer. Her veri ile rastgele belirlenen merkez noktaları arasındaki uzaklığı hesaplayarak veriyi en yakın merkez noktasına göre bir kümeye atar. Daha sonra her küme için yeniden bir merkez noktası seçilir ve yeni merkez noktalarına göre kümeleme işlemi yapılır. Bu durum sistem kararlı hale gelene kadar devam eder. Algoritmanın adımları şöyledir;

- Küme merkezleri rasgele olarak belirlenir.
- Her bir veri örneğinin küme merkezlerine uzaklığını bulunur.
- Örneği merkezine yakın olan kümeye ait olduğu belirlenir.
- Her bir küme için örneklerin ortalamasını o kümenin merkezi olarak belirlenir.
- Merkezlerde değişim oranı kritik bir sınırın altına düşene kadar 2,3 ve 4. adımlar tekrarlanır.

### C. K-En Yakın Komşu Algoritması

Gözetimli öğrenme metodlarından sınıflandırma işleminde kullanılan K-En Yakın Komşu algoritmasında K değeri bakılacak eleman sayısını belirler [9]. K-En Yakın Komşu (K-EYK)

algoritması K-Ortalama algoritmasıyla benzer özellikler taşısa da bazı farklılıklar da içermektedir. K-EYK algoritması bir eğitim verisi içerirken K-Ortalama algoritması bir eğitim verisi içermez. Yeni bir örnek geldiğinde K tane örneğe mesafeler hesaplanır ve yeni örnek bir kümeye ilave edilir.

Bu algoritma beş adımdan oluşur;

- Öncelikle K değeri belirlenir.
- Diğer nesnelere hedef nesneye olan uzaklıklar hesaplanır.
- Uzaklıklar sıralanır ve minimum uzaklığa bağlı olarak en yakın komşular bulunur.
- En yakın komşu kategorileri toplanır.
- En uygun komşu kategorisi seçilir.

### D. Mesafe Ölçütleri

Mesafe ölçütleri kümeleme algoritmalarının performansını etkileyen en önemli etkidir [10]. Farklı mesafe ölçütleri bu kısımda açıklanmıştır.

**Cosine** N boyutlu uzayda iki noktanın nokta vektörleri arasındaki açının kosinüsünü belirler.

$$d(a, b) = \frac{\sum_{j=1}^k a_j b_j}{\sqrt{\sum_{j=1}^k a_j^2 \sum_{j=1}^k b_j^2}} \tag{3}$$

**Öklid** Öklid uzaklığı, sınıflandırma ve kümeleme algoritmalarında en sık kullanılan mesafe ölçütüdür. Öklid uzaklığı, iki nokta arasındaki doğrusal uzaklık olup herhangi iki nokta, a ve b arasındaki Öklid uzaklığı eşitlik 4'de göre hesaplanır.

$$d(a, b) = \sqrt{\sum_{j=1}^k (a_j - b_j)^2} \tag{4}$$

**Manhattan** İki veri noktasının koordinatları arasındaki mutlak farkların toplamını hesaplar.

$$d(a, b) = \sum_{j=1}^k |a_j - b_j| \tag{5}$$

**Braycurtis** Farklı örnekleme lokasyonlarında toplanan veriler söz konusu olduğunda, Bray-Curtis ölçütü örnekler arasındaki farkı ölçmenin en bilinen yollarından biridir.

$$d(a, b) = \frac{\sum_{j=1}^k |a_j - b_j|}{\sum_{j=1}^k a_j + b_j} \tag{6}$$

**Canberra** Canberra mesafe ölçütü, sıralı listeleri karşılaştırmak için kullanılan bir ölçüttür.

$$d(a, b) = \sum_{j=1}^k \frac{|a_j - b_j|}{|a_j + b_j|} \tag{7}$$

**Chebyshev** Chebyshev uzaklığı, Minkowski mesafe ölçütündeki girdi vektör uzunluğunun sonsuz olduğu özel durum olup, iki nokta arasındaki farkların mutlak değerlerinin maksimumu olarak tanımlanmaktadır.

$$d(a, b) = \max(|a_j - b_j|) \quad (8)$$

*Mahalanobis* Mahalanobis mesafesi, bir  $a$  noktası ile bir  $b$  dağılımı arasındaki mesafenin bir ölçüsüdür,  $a$ 'nin  $b$  ortalamasından kaç standart sapma uzakta olduğunu ölçme fikrinin çok boyutlu bir genellemesidir. Burada  $S$ ,  $b$  dağılımının kovaryans matrisidir.

$$d(a, b) = \sqrt{(a_k - b_k)^T S (a_k - b_k)} \quad (9)$$

*Minkowski* Aynı zamanda genelleştirilmiş mesafe metriği olarak da bilinir. Hem sıralı hem de nicel değişkenler için kullanılabilir. Aşağıdaki eşitlikte  $p$  pozitif bir reel sayıdır.

$$d(a, b) = \left( \sum_{j=1}^k |a_j - b_j|^p \right)^{1/p} \quad (10)$$

#### IV. BENZETİM SONUÇLARI

Bu çalışmada önerilen sınıflandırıcıyı eğitmek ve performansını görebilmek amacıyla yapay olarak üretilen veriler kullanılmıştır. Her ATT için 1000 adet anten tarama örüntüsü üretilmiştir. Veriler üretilirken Konik, Kesik Konik ve Spiral ATT'ler için anten tarama periyodu 10 - 100 ms aralığında; diğer ATT'ler için 0.1-100s aralığında seçilmiştir. Her veri için Radar ve EH sisteminin konumları rastgele değiştirilerek farklı anten tarama örüntüleri oluşması sağlanmıştır. Daha sonra üretilen anten örüntülerine bozulma etkileri de eklenmiştir. Öznitelikler Bölüm III.A'da anlatıldığı gibi çıkarılmıştır. Çıkarılan öznitelik vektörünün boyutu üç olarak seçilmiştir. Bu işlem sırasında anten tarama örüntüleri aydınlatma aydınlatma ayrılarak öznitelikler aydınlatmalar üzerinden hesaplanmıştır. Aydınlatma kararı ardışık darbelerin geliş zamanları arasındaki farka ( $toa(i) - toa(i - 1) \geq E_{sik}$ ) bakılarak verilmiştir. Konik, Kesik Konik ve Spiral ATT'ler için veri sürekli olduğundan ilk 5000 örnekten 1000 örnek seçilerek alınmıştır. Toplam 9000 adet anten tarama örüntüsünden %80'i sınıflandırıcının eğitimi için, %20'ü ise test için kullanılmıştır. K-Ortalama algoritması ile K-EYK algoritması farklı mesafe ölçütleri için karşılaştırılmıştır. Karşılaştırma sonuçları Tablo I'de verilmiştir. Başarım oranı (11) kullanılarak hesaplanmıştır. K-EYK algoritması bütün mesafe ölçütleri için K-Ortalama algoritmasından daha iyi sonuçlar vermiştir.

$$d(a, b) = \frac{\sum_{j=1}^n \gamma(s_i, \text{map}(r_i))}{n} \quad (11)$$

$r_i$  ve  $s_i$  sırasıyla kümeleme algoritmasının çıktısı ve örneğin gerçek etiketidir.  $\text{map}$  optimum sınıf kombinasyonun temsil etmektedir.  $n$  toplam anten tarama örneği sayısını temsil etmektedir.  $\gamma$  fonksiyonu (12)'de verilmiştir. K-Ortalama algoritması en iyi başarımları Manhattan mesafe ölçütü kullanıldığında vermiştir. K-En Yakın Komşu algoritması ise mahalanobis mesafe ölçütü kullanıldığında en iyi başarımları vermiştir. K-En Yakın Komşu algoritmasının bütün ölçütleri için başarımları yüzdesi K-Ortamala algoritmasına göre daha iyi başarımları göstermiştir. Bunun sebebi K-Ortamala algoritmasının küme merkezi ve örnek arasındaki uzaklığa bakmasıdır. Otomatik kodlayıcı yapısından çıkan özniteliklerin dağılımı küme içinde homojen olmadığından K-Ortamala algoritması hatalı sınıflandırmaya yol açmaktadır.

$$\gamma(x, y) = \begin{cases} 1 & x = y \\ 0 & x \neq y \end{cases} \quad (12)$$

TABLO I: K-ORTALAMA VE K-EYK ALGORİTMALARI İÇİN FARKLI MESAFE ÖLÇÜTÜ SONUÇLARI

Ölçütler	Kümeleme Algoritmaları	
	K-Ortamala	K-En Yakın Komşu
Cosine	%50	%92.7
Öklid	%65	%93.8
Manhattan	%81	%92.7
Braycurtis	%57	%93.9
Canberra	%62	%93.2
Chebyshev	%63	%93.7
Mahalanobis	%71	%95.4
Minkowski	%65	% 94.8

#### V. SONUÇ

Bu çalışmada EH sistemleri için ATT sınıflandırma işleme otomatik hale getirilmiştir. İlk olarak otomatik kodlayıcılar kullanılmış ve öznitelik vektörü çıkartılmıştır. Bu öznitelik vektörü farklı kümeleme algoritmaları kullanılarak kümeleme işlemi yapılır. Kümeleme işleminde en iyi sonucu mahalanobis mesafe ölçütü kullanılan K-En Yakın Komşu algoritması % 95.4 olarak bulunmuştur. Mahalanobis mesafe ölçütünün en iyi başarımları yüzdesi vermesinin sebebi olarak veri dağılımına da bakmasıdır. K-Ortamala ve K-En Yakın Komşu algoritmaları farklı mesafe ölçütlerine karşılaştırılmıştır. K-En Yakın Komşu algoritmasının performansı mesafe ölçütlerinin K-ortalama tabanlı kümelemeye kıyasla daha iyi sonuçlar vermiştir.

#### KAYNAKLAR

- [1] S. Robertson, Practical ESM Analysis. Artech House, 2019.
- [2] B. Barshan and B. Eravci, "Automatic radar antenna scan type recognition in electronic warfare," IEEE Transactions on Aerospace and Electronic Systems, vol. 48, no. 4, pp. 2908–2931, 2012.
- [3] Y.-H. Kim, K.-H. Song, J.-W. Han, and H.-N. Kim, "Radar scan pattern analysis for reduction of false identification in electronic warfare support systems," IET Radar, Sonar Navigation, vol. 8, no. 7, pp. 719–728, 2014.
- [4] S. Ayazgok, C. Erdem, M. T. Ozturk, A. Orduyilmaz, and M. Serin, "Automatic antenna scan type classification for next-generation electronic warfare receivers," IET Radar, Sonar Navigation, vol. 12, no. 4, pp. 466–474, 2018.
- [5] E. Özmen, A. G. Aydın, H. S. Balaban and A. K. Kamçı, "Antenna Scanning Type Classification using Machine Learning," 2020 28th Signal Processing and Communications Applications Conference (SIU), Gaziantep, Turkey, 2020, pp. 1-4.
- [6] L. Cai, S. Zhou, X. Yan, and R. Yuan, "A stacked bilstm neural network based on coattention mechanism for question answering," Computational intelligence and neuroscience, vol. 2019, 2019.
- [7] T. Liu, S. Yu, B. Xu, and H. Yin, "Recurrent networks with attention and convolutional networks for sentence representation and classification," Applied Intelligence, vol. 48, no. 10, pp. 3797–3806, 2018.
- [8] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.
- [9] Wang, Zhenggang, et al. "Unsupervised Clustering of Neighborhood Associations and Image Segmentation Applications." Algorithms 13.12 (2020): 309.
- [10] A. E. TAŞCI And A. Onan, "K En Yakın Komşu Algoritması Parametrelerinin Sınıflandırma Performansı Üzerine Etkisinin İncelenmesi," Akademik Bilişim , Aydın, Turkey, pp.1-8, 2016