

ÖĞRENME AKTARIMI İLE SINIFLANDIRMA

CLASSIFICATION WITH TRANSFER LEARNING

MEHMET OTURAK

DR. ÖĞR. ÜYESİ SENİHA ESEN YÜKSEL

Tez Danışmanı

Hacettepe Üniversitesi

Lisansüstü Eğitim – Öğretim ve Sınav Yönetmeliğinin

Elektrik ve Elektronik Mühendisliği Anabilim Dalı İçin Öngördüğü

YÜKSEK LİSANS TEZİ olarak hazırlanmıştır.

2018

Mehmet OTURAK'ın hazırladığı "**Öğrenme Aktarımı ile Sınıflandırma**" adlı bu çalışma aşağıdaki jüri tarafından **ELEKTRİK ve ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI**'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Prof. Dr. Gözde BOZDAĞI AKAR
Başkan

Dr. Öğr. Üyesi S. Esen YÜKSEL
Danışman

Prof. Dr. Birsen SAKA TANATAR
Üye

Doç. Dr. Berkan DÜLEK
Üye

Dr. Öğr. Üyesi Yakup ÖZKAZANÇ
Üye

Bu tez Hacettepe Üniversitesi Fen Bilimleri Enstitüsü tarafından **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Prof. Dr. Menemşe GÜMÜŞDERELİOĞLU
Fen Bilimleri Enstitüsü Müdürü

YAYINLAMA VE FİKRİ MÜLKİYET HAKLARI BEYANI

Enstitü tarafından onaylanan lisansüstü tezimin tamamını veya herhangi bir kısmını, basılı (kağıt) ve elektronik formatta arşivleme ve aşağıda verilen koşullarla kullanıma açma iznini Hacettepe üniversitesine verdiğimi bildiririm. Bu izinle Üniversiteye verilen kullanım hakları dışındaki tüm fikri mülkiyet haklarım bende kalacak, tezimin tamamının ya da bir bölümünün gelecekteki çalışmalarda (makale, kitap, lisans ve patent vb.) kullanım hakları bana ait olacaktır.

Tezin kendi orijinal çalışmam olduğumu, başkalarının haklarını ihlal etmediğimi ve tezimin tek yetkili sahibi olduğumu beyan ve taahhüt ederim. Tezimde yer alan telif hakkı bulunan ve sahiplerinden yazılı izin alınarak kullanması zorunlu metinlerin yazılı izin alarak kullandığımı ve istenildiğinde suretlerini Üniversiteye teslim etmeyi taahhüt ederim.

- Tezimin tamamı dünya çapında erişime açılabilir ve bir kısmı veya tamamının fotokopisi alınabilir.**

(Bu seçenekle teziniz arama motorlarında indekslenebilecek, daha sonra tezinizin erişim statüsünün değiştirilmesini talep etmeniz ve kütüphane bu talebinizi yerine getirirse bile, tezinin arama motorlarının önbelleklerinde kalmaya devam edebilecektir.)

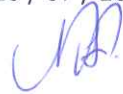
- Tezimin tarihine kadar erişime açılmasını ve fotokopi alınmasını (İç Kapak, Özet, İçindekiler ve Kaynakça hariç) istemiyorum.**

(Bu sürenin sonunda uzatma için başvuruda bulunmadığım takdirde, tezimin/raporumun tamamı her yerden erişime açılabilir, kaynak gösterilmek şartıyla bir kısmı ve ya tamamının fotokopisi alınabilir)

- Tezimin tarihine kadar erişime açılmasını istemiyorum, ancak kaynak gösterilmek şartıyla bir kısmı veya tamamının fotokopisinin alınmasını onaylıyorum.**

- Serbest Seçenek/Yazarın Seçimi**

10 / 07 / 2018



(İmza)

Öğrencinin Adı Soyadı

Mehmet OTURAK

Her zaman yanımda olan canım aileme...

ETİK

Hacettepe Üniversitesi Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada,

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

10 / 07 / 2018

Mehmet OTURAK

ÖZET

ÖĞRENME AKTARIMI İLE SINIFLANDIRMA

Mehmet OTURAK

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Danışmanı: Dr. Öğr. Üyesi Seniha Esen YÜKSEL

Temmuz 2018, 79 sayfa

Sadece birkaç örnekten bir nesne kategorisinin öğrenilmesi genellikle doğru sınıflandırma için yeterli değildir. İyi bir şekilde genelleştiren ve yeterli başarı oranına sahip bir sınıflandırıcı elde etmek için birçok eğitim örneğine ihtiyaç vardır. Bununla birlikte, Yeraltı Radarı (GPR) ile hedef tespiti dahil olmak üzere birçok uygulamada, yeterli sayıda etiketlenmiş veri toplanması her zaman mümkün değildir. GPR verisi toplamada, oluşan görüntüler dielektrik geçirgenlik ve manyetik geçirgenlik gibi toprak özelliklerine doğrusal olmayan bir bağımlılık göstermektedir. Bu nedenle, bir toprak tipi için (kuru kum gibi) iyi bir sınıflandırıcıyı eğitmek için yeterli eğitim verisi mevcut olsa bile, Bu sınıflandırıcının başarısı, toprak tipi değiştiğinde (örn. ıslak kum) yeterince iyi olmayabilir.

Bu çalışmada, GPR kullanarak farklı ortamlarda gömülü piknik tüpü tespit etmek için çok modelli bilgi aktarımı (BA) önerilmiştir. Önerilen yöntemde, ölçekten bağımsız öznitelik dönüşümü (SIFT) yöntemi ile GPR verilerinden öznitelikler hesaplanmıştır. Daha sonra, en küçük kare destek vektör makinesi (LS-SVM) sınıflandırıcıları, bol miktarda verinin mevcut olduğu üç toprak tipi (kuru kum, kuru kumtaşı ve ıslak kumtaşı) için eğitilmiştir. Daha sonra, adaptif LS-SVM, sadece az miktarda eğitim verisinin olduğu ıslak kumdaki hedef piknik tüpünü tespit eden bir sınıflandırıcıyı eğitmek için kullanılmıştır. Sonuç olarak, (i) birden fazla kaynaktan (yani birden çok toprak türünden) bilgi aktarımının, tek kaynaktan aktarımdan daha iyi sonuçlar ürettiği; ve (ii) hedef sınıftan sadece 3 eğitim verisi ile, tespit oranlarının tek kaynaktan BA ile % 25, çok kaynaktan BA ile de % 32 arttığı gösterilmiştir.

Anahtar Kelimeler: Bilgi Aktarımı, Yeraltı Radarı, SIFT, LS-SVM

ABSTRACT

CLASSIFICATION WITH TRANSFER LEARNING

Mehmet OTURAK

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

Supervisor: Assist. Prof. Dr. Seniha Esen YÜKSEL

July 2018, 79 pages

Learning an object category from only a few samples is generally not adequate for correct classification. One needs many training samples to obtain a classifier that generalizes well and that has sufficient success rates. However, in several applications, including target detection from Ground Penetrated Radar (GPR) data, collecting a lot of annotated data is not always possible. In a GPR data collection, the images formed show a nonlinear dependence on the soil properties such as the permeability and permittivity. Therefore, even if enough training data were available to train a good classifier for one soil type (such as dry sand); the success of this classifier does not translate well if the soil type is changed (say, to wet sand). In this work, multi-model knowledge transfer (KT) is proposed to detect a picnic tube buried in different media using GPR. In the proposed method, scale invariant feature transform (SIFT) features are extracted from GPR data. Then, least-square support vector machine (LS-SVM) classifiers are trained for three soil types (i.e. dry sand, dry sandstone, and wet sandstone) where there is ample data available. Then, adaptive LS-SVM is used to train a classifier that detects the target picnic tube in wet sand from where there is only scarcely available training data. We show that (i) knowledge transfer from multiple sources (i.e. multiple types of sand) generates better results than single source transfer; and (ii) as little as 3 training data from the unknown source increases the detection rates by 25% for single source KT, and 32% for multiple source KT

Keywords: Transfer Learning, Knowledge Transfer, GPR, SIFT, LS-SVM

TEŐEKKÜR

Bu tezin oluŐturulmasında sađladıđı katkılardan ötürü tez danıŐmanım Dr. Öđr. Üyesi Seniha Esen YÜKSEL'e teŐekkürlerimi sunuyorum.

Benden sevgilerini esirgemeyen ve bana her zaman destek olan aileme sonsuz teŐekkür ederim.

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET	i
ABSTRACT	ii
TEŞEKKÜR	iii
İÇİNDEKİLER.....	iv
ŞEKİLLER	vi
ÇİZELGELER.....	ix
SİMGELER VE KISALTMALAR	x
1. GİRİŞ	1
1.1. Tezin Amacı.....	3
1.2. Tezin Akışı.....	4
2. GPR VERİLERİNİN OLUŞTURULMASI.....	5
2.1. Yeraltı Radarı Çalışma İlkesi.....	6
2.2. GPR Benzetimleri.....	7
2.2.1. Girdi dosyası.....	8
2.2.2. Çıktı Dosyası.....	16
2.3. Hedef ve Kaynak Verileri	17
3. GPR VERİSİNİN İŞLENMESİ VE ÖZİNİTELİK ÇIKARIMI.....	19
3.1. Yer yansımalarının çıkartılması	19
3.2. Görüntü Beyazlaştırma (Whitening).....	19
3.3. SIFT Öznelikleri.....	21
3.3.1. Kapalı eğrilerin Tespit Edilmesi	21
3.4. GPR Görüntüsünden Öznelik Elde Etme	24
3.5. Pozitif Veriler	26
3.5.1. Birinci Seviye Çeşitlilik.....	26
3.5.2. İkinci Seviye Çeşitlilik.....	27
3.5.3. Üçüncü Seviye Çeşitlilik.....	29
3.6. Negatif Veriler	31
4. LS-SVM SINIFLANDIRICI	34
4.1. Destek Vektör Makinesi (SVM).....	34
4.2. En Küçük Kareler Destek Vektör Makinesi (LS-SVM).....	40
4.2.1. LS-SVM Eğitim.....	42
4.2.2. LS-SVM Sezim.....	45
4.2.3. GPR Verilerinde LS-SVM.....	48

5. BİLGİ AKTARIMI.....	52
5.1. Uyarlamalı LS-SVM	53
5.2. Çok Modelden Bilgi Aktarımı	54
5.3. Ne Zaman ve Ne Kadar Bilgi Aktarılmalı	57
6. BİLGİ AKTARIMI İLE ELDE EDİLEN DENEYSEL SONUÇLAR.....	66
7. SONUÇLAR VE DEĞERLENDİRMELER	71
KAYNAKLAR.....	75
ÖZGEÇMİŞ.....	79

ŞEKİLLER

Şekil 1: Kuru kum içine gömülmüş piknik tüpü.....	1
Şekil 2: Yeraltı Radarı Çalışma Ortamı.....	6
Şekil 3: Yeraltı Radarı ile hedef tespiti. (a) Alıcı-Verici ikilisi toprak üzerinde bir konumdayken, vericiden gönderilen EM dalgaları alıcı tarafından yakalanır ve bu işlem sabit mesafe adımları ile tekrarlanır. (b) Farklı konumlarda alınan sinyaller birleştirilerek GPR görüntüsü oluşturulur.	7
Şekil 4: (a) Simülasyon ortamında oluşturulmuş küresel bir cisim. (b) gaussiandot dalga şekli kullanan verici ile alınan bir A-Scan sinyali.....	8
Şekil 5: Oluşturulan bir simülasyon ortamı ve elde edilen yeraltı radarı görüntüsü. (a) Kuru Toprak İçinde Tüp. (b) Üzerinde herhangi bir işlemenin yapılmadığı B-scan taraması görüntüsü.....	17
Şekil 6: Yer yansımalarının çıkartılmasından sonra B-scan taraması.....	19
Şekil 7: GPR görüntüsünde yer yansımaları çıkartıldıktan sonra yapılan beyazlatma işleminin sonucu. (a) Satır blokları arasında örtüşme yokken (birinci aşama) beyazlatma. (b) Satır blokları arasında %50 örtüşme varken (ikinci aşama) beyazlatma.	20
Şekil 8: Tek Boyutlu Fonksiyonda Kenar Tespit Etme. (üst) Resme ait piksel değerlerinin fonksiyonu. (alt) Fonksiyonun ikinci dereceden türevi, sıfırdan geçiş bilgisini vererek resimde yer alan kenarları bulmaktadır.	21
Şekil 9: Resimde yer alan gürültüleri ortadan kaldırmak için kullanılan (a) Gauss filtresi, (b) birinci türevi ve (c) ikinci türevi.....	23
Şekil 10: LoG filtrenin bir kapalı eğriye (dikdörtgen) uygulanması. (a) Resimde yer alan dikdörtgen ve uygulanan LoG filtre ölçekleri uyumlu değilse sıfırdan geçişler birbirinden uzaktır. (c) Şekil ve filtre ölçekleri uyumlu ise, sıfırdan geçişler üst üste biner ve yerel uç değer oluşturur.	23
Şekil 11: Gauss filtresinin varyansı artırılarak LoG Filtresinin boyutunun artırılması.	24
Şekil 12: Kapalı eğriyi tespit etmek için LoG filtre boyutunun değiştirilmesi. LoG filtrenin boyutu artırılarak resme uygulanır. Uç değer elde edilmişse, kapalı eğri ve boyutu bulunmuş olur.....	24
Şekil 13: (a) GPR Görüntüsünden (b) Anahtar noktalarının ve (c) Tanımlayıcının elde edilmesi.....	25

Şekil 14: Farklı toprak tipleri içinde yer alan tüp ile elde edilmiş tanımlayıcılar. (a) Kuru kum içinde tüp. (b) Kuru kumtaşı içinde tüp. (c) Islak kumtaşı içinde tüp. (d) Islak kum içinde tüp (hedef sınıf).....	26
Şekil 15: Pozitif veri toplamak için yapılan birinci seviye çeşitlilik. Tespit edilecek nesnenin boyutu, şekli, eğikliği vb. değiştirilerek elde edilir.....	27
Şekil 16: Birinci seviye çeşitlilik ile elde edilen tanımlayıcılar. Şekil 15'te yer alan nesne ve ortamlara ait simülasyonla elde edilmiş tanımlayıcılar burada aynı sıra ile yer almaktadır.....	27
Şekil 17: Kuru kumtaşı toprak içerisine gömülü iletken tüpe ait ikinci seviye çeşitlilik ile elde edilen tanımlayıcılar. Simülasyonlardaki tek değişen parametre, toprağa ait ϵ_r ve μ_r değerleridir.....	28
Şekil 18: Üçüncü seviye çeşitlilik için elde edilen pencereler. Merkezdeki resimde yer alan pencere referans olacak şekilde, solundaki ve sağındaki resimlerde pencere y ekseninde; yukarısındaki ve aşağısındaki resimlerde de x ekseninde kaymaktadır.....	30
Şekil 19: Pozitif veri toplama hiyerarşisi. Birinci seviyede, nesneye ait fiziksel özellikler (boyut, şekil vb.); ikinci seviyede toprağa ait parametreler (ϵ_r , μ_r) ve üçüncü seviyede de oluşturulan penceredeki konumsal değişiklikler ile pozitif eğitim verisi çoğaltılmaktadır.....	31
Şekil 20: Sınıflandırıcı eğitiminde kullanılan negatif veriler (Clutter).....	32
Şekil 21: Negatif verilere ait tanımlayıcılar. Şekil 20'de yer alan nesne ve ortamlardan elde edilmiş tanımlayıcılar burada aynı sıra ile yer almaktadır.....	32
Şekil 22: Negatif veri toplama hiyerarşisi. Birinci seviyede, ortamda bulunan nesnenin kendisi (küp, çakıl vb.); ikinci seviyede toprağa ait parametreler (ϵ_r , μ_r) ve üçüncü seviyede de oluşturulan penceredeki konumsal değişiklikler ile negatif eğitim verisi çoğaltılmaktadır.....	33
Şekil 23: Vektör uzayında noktaların hiper düzlem ile ilişkisi.....	34
Şekil 24: İki noktayı ayırabilen hiper düzlemler	35
Şekil 25: w_3 hiper düzlem kümesi	37
Şekil 26: Hiper düzlem formülüne göre noktaların değeri	37
Şekil 27: En iyi hiper düzlem	40
Şekil 28: Girdi verisini daha yüksek boyutlu uzaya taşıma.....	42
Şekil 29: Girdi verisini daha yüksek boyutlu uzaya taşıma.....	42
Şekil 30: LS-SVM Sınıflandırma Örnekleri	48
Şekil 31: “Islak kum içinde tüp” sınıfı için sezim sonuçları.	48

Şekil 32 GPR görüntülerine gürültü eklenmesi	49
Şekil 33: Gürültü seviyelerine göre tanıma başarımların grafikleri.....	50
Şekil 34: Gürültü eklenmiş verilerde sınıflara ait tanıma başarımların grafikleri	51
Şekil 35: Az sayıda eğitim verisi ile tanıma oranları.....	52
Şekil 36: Beta değerlerini en iyileştirirken kullanılan hata değerleri	63
Şekil 37: Giriş verilerinden yola çıkarak Çok Modelli Bilgi Aktarımının gerçekleştirilmesinin akış grafiği.....	65
Şekil 38: Eğitimde kullanılan veri sayısı birden başlayarak 10'a kadar artırılırsa oluşan sezim oranları grafiği. (Siyah) Bilgi Aktarımının olmadığı durum. (Kırmızı) Tek kaynaktan Bilgi Aktarımının olduğu durum. (Mavi) 3 kaynaktan bilgi aktarımının olduğu durum.....	66
Şekil 39: Daha fazla sayıda eğitim verisi ile elde edilmiş Bilgi Aktarımını sonuçları. Şekil, eğitim numunelerinin sayısı daha da artırıldığı durumda, Bilgi Aktarımının başarısının devam ettiğini vurgulamak için Şekil 38'nin uzatılmış versiyonudur.....	68
Şekil 40:Eğitimde kullanılan veri sayısının 200 olması durumunda BA sonuçları.....	70
Şekil 41: Kaynak sınıf verilerinin birleşimi ile oluşturulan sınıflandırıcı başarı grafiği.	72

ÇİZELGELER

Tablo 1: Girdi Dosyasında Kullanılan Komutlar (Özet)	8
Tablo 2: Girdi Dosyasında Kullanılan Komutlar (Detay)	9
Tablo 3: Kullanılan sınıflara ait ortamların parametreleri	17
Tablo 4: İçler Çarpımı Çekirdek Tipleri	44
Tablo 5: Kaynak modellerinin, ıslak kum içerisinde tüp hedef verisini tanıma oranları.....	53
Tablo 6: Şekil 38 (a)'daki değerler ve standart sapmaları	67
Tablo 7: Şekil 38 (b)'deki değerler ve standart sapmaları	67
Tablo 8: Şekil 38 (c)'deki değerler ve standart sapmaları	67
Tablo 9: Bilgi Aktarımı sonucunda elde edilen beta değerleri	68
Tablo 10: Şekil 39'daki değerler ve standart sapmaları	69
Tablo 11: Şekil 40'daki değerler ve standart sapmaları	70
Tablo 12: Çok sayıda veri ile sınıflandırma	71
Tablo 13: Az sayıda veri ile sınıflandırma	71
Tablo 14: Kaynak modellerinin, ıslak kum içerisinde tüp hedef verisini tanıma oranları	72
Tablo 15: Şekil 41'deki değerler ve standart sapmaları	73
Tablo 16: Kaynak sınıfların birleşim modeli ile hedef sınıf modelinin karşılaştırılması	73

SİMGELER VE KISALTMALAR

Simgeler

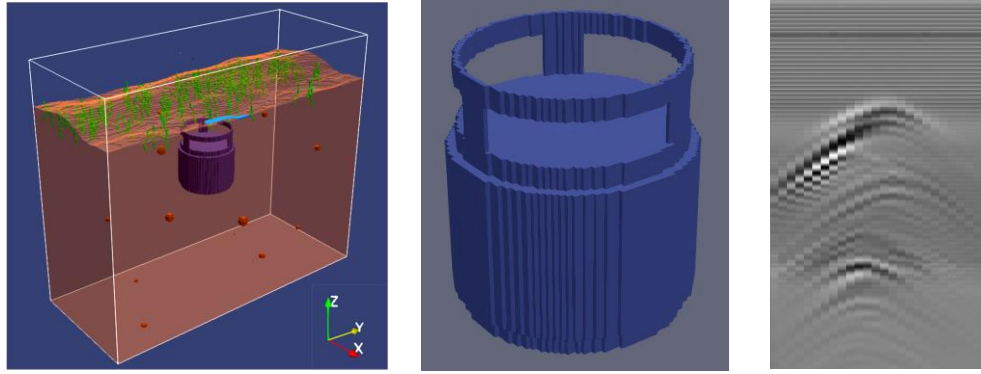
ϵ_r	relative permittivity	bağıl elektrik geçirgenlik
σ	conductivity	iletkenlik
μ_r	relative permeability	bağıl manyetik geçirgenlik
α	magnetic loss	manyetik kayıp

Kısaltmalar

BA	Knowledge Transfer	Bilgi Aktarımı
EM	Electro Magnetic	Elektro Manyetik
EYP	Improvised Explosive Devices	El Yapımı Patlayıcı
GPR	Ground Penetrating Radar	Yere Nüfuz Eden Radar
IED	Improvised Explosive Devices	El Yapımı Patlayıcı
loo	leave-one-out	birini hariç tutma
LS-SVM	Least-Square Support Vector Machine	En Küçük Kareler Destek Vektör Makinesi
PEC	Perfect Electric Conductor	Mükemmel Elektrik İletkeni
SIFT	Scale Invariant Feature Transform	Ölçekten Bağımsız Öznitelik Dönüşümü
SVM	Support Vector Machine	Destek Vektör Makinası

1. GİRİŞ

El yapımı patlayıcılar (EYP); şişe teneke, sargı vb. gibi çok çeşitli malzemelerden alıılmışın dışında inşa edilen bombalardır. Hem Irak savaşında hem de Afganistan savaşında, kayıpların bir numaralı sebebi bunlardır [1]. Bununla birlikte, savaş bölgesinden bir EYP'nin gerçek GPR görüntülerinin elde edilmesi kolay değildir ve yeterli eğitim verisi toplamak için genellikle bir “rüya” olabilir. Öte yandan, GPR sinyali dielektrik geçirgenlik ve manyetik geçirgenlik gibi toprak özelliklerine bağlıdır. Bu nedenle, bir tür toprağa gömülmüş bir EYP, başka bir toprak türüne gömüldüğünde GPR taramasında aynı görüntüyü vermez. Bu durum şu soruyu gündeme getirir: “Elimizdeki tek şey güvenli yerlerden veri ise, güvenliğini sağlamak istediğimiz yerler için yeterince iyi modelleri nasıl eğitebiliriz?”



(a) Ortamın Paraview görüntüsü (b) Tespit edilecek piknik tüpü (c) GPR görüntüsü

Şekil 1: Kuru kum içine gömülmüş piknik tüpü

Bu çalışmada, farklı toprağa gömülmüş piknik tüpünün saptanması, hem gömülü nesne modeli hem de GPR görüntüsünün verildiği Şekil 1'de gösterildiği gibi simülasyonlarla incelenmiştir. Piknik tüpü en yaygın kullanılan EYP'lerden biridir. Bunların elde edilmesi çok kolaydır ve çok fazla hasara yol açarlar. Bu nedenle amacımız, (1) modellerimizi çoklu toprak türlerinde (kaynak sınıfı) eğitsek ve başka bir toprak türünde (hedef sınıfı) test edersek performansımızın ne olacağını ve (2) sınıflandırma oranlarımızı artırmak istersek hedef türünden ne kadar veriye ihtiyaç duyacağımızın anlaşılmasıdır. Bu soruları yanıtlamak için, çok modelden bilgi aktarımı yöntemi [2] kullanılmış ve GPR verilerinden hedef tespiti için uyarlanmıştır.

Bilgi aktarımı (BA), yeni kavramları öğrenmeyi son derece verimli, hızlı ve kolay hale getirir. Bu yöntemle, nesnelere aşamalı olarak öğrenen insanlar gibi, yeni bir durumla karşılaşıldığında, yeni kavramla ilgili öğrenme kendi başına bağımsız olarak yönetilmez; daha ziyade, kavramla ilgili parametrelere dayanarak önceki kavramlarla ilişkiler kurulur

[3], [4]. Örneğin, daha önceden arabaları öğrenmiş olan bir çocuk, ilk defa bir kamyon ile karşılaştığında hemen iki nesne sınıfı arasında bir benzerlik kurar. Her ikisinde de tekerlekler mevcuttur, ön tarafta bir sürücü oturmaktadır, sürücünün dış ortamı görmesi camlar aracılığı ile olmaktadır ve araca giriş çıkışlar için kapı bulunmaktadır. Bu benzerliği kurarak, arabadan öğrenilen bilgiler, yeni karşılaşılan kamyon sınıfının öğrenilmesinde kullanılır.

Bu çalışmada, farklı özelliklere sahip toprakta gömülmüş aynı nesne kullanılarak oluşturulan GPR görüntü kategorileri arasında bilgi aktarımı yapılmıştır. Dört GPR görüntü kategorisi üretilmiş ve kategorilerden birisi hedef sınıfı (ıslak kum) olarak kalan üç sınıf da (kuru kum, kuru kumtaşı ve ıslak kumtaşı) kaynak sınıfı olarak seçilmiştir. Eğitim ve test verisi olarak, oluşturulan bu ortamlara hedef ve çeşitli dağınık nesnelere yerleştirilerek GPR görüntüleri gprMax aracılığıyla oluşturulmuştur [5], [6], [7].

GprMax'te hedef, Şekil 1(a)'da olduğu gibi, farklı türden cisimlerin de eklenebildiği arzu edilen toprak içerisine yerleştirilir. Bu alanın üzerine yerleştirilen bir GPR iletim anteni, seçilen bir frekansa ve dalga şekline sahip olan bir elektromanyetik dalga yayar. Medyada yayılan dalga, medyadaki dielektrik sabitlerinde bir süreksizlik ile karşı karşıya kalması üzerine yayılma yönünün tersine yansır ve toprak üzerinde vericinin yakınına konan bir alıcı anten vasıtasıyla bu yansıyan sinyaller toplanır. Bir elektromanyetik dalga ile yapılan ölçüm A-scan olarak isimlendirilirken, alıcı-vericinin ilerleme yönünde başlangıç noktasından sabit adım uzunluklarında ölçülen A-scan'lerin toplamı B-scan olarak adlandırılır [8]. Toprak içerisinde radar dalgasının ilerleme yönünde farklı dielektrik sabiti gösteren gömülü piknik tüpü, Şekil 1(c)'de görüldüğü gibi, B-taramasında hiperbolik şekil olarak gözlenir.

Bu hiperbolik görünümü tespit etmek için literatürde birkaç GPR hedef tespit tekniği kullanılmıştır. Bu teknikler, hedefi daha net hale getirerek tespitini kolaylaştırmak için genellikle veri ön işleme adımları kullanır. Bu ön işleme, (derinliğe bağlı bilgi kaybını önlemek için) verinin normalleştirilmesi ve gürültü ile arka planın çıkarılması adımlarını içerir [9], [10], [11], [12]. Görüntü ön işlemeden geçirildikten sonra, hedef algılama için model veya öznitelik tabanlı algılama yöntemleri kullanılır. Hough dönüşümü gibi model tabanlı algılama yöntemleri, hiperbol benzeri bir şekil arar [13]. Ancak, yüzeyin pürüzlülüğü, toprağın homojensizliği, nesne ve engellerin şekli veya eğimi bu hiperbolik şekli bozabilir. Öte yandan, ikinci türev [14], [15], yönelimli gradyanların histogramı (Histograms of Oriented Gradients) [16], temel bileşenler (principal components) ve Fourier katsayıları (Fourier coefficients) [17] gibi öznitelik tabanlı tespit tekniklerinin çoğu zaman çevresel değişimlere daha az duyarlı olduğu bulunmuştur.

Literatürde sınıflandırma cephesinde, Gizli Markov Modelleri (Hidden Markov Models) [14], [15], [18], [19], Çekirdek En Az Ortalama Kare Yöntemi (KLMS: Kernel Least Mean Square Method) [20], Yapay Sinir Ağları [21], [22], Katlamalı Sinir Ağları (CNN: Convolutional Neural Networks) [23], [24], Daha Hızlı RCNN [25], AdaBoost [26] ve SVM [22], popüler olarak kullanılan tekniklerden bazılarıdır.

Bu tez çalışması kapsamında, SIFT (Scale Invariant Feature Transform) öznelikleri [27] çıkartılmış ve LS-SVM [28] ile tespit edilmiştir. SIFT öznelikleri, ölçek bağımsız olmasından ve bir görüntüdeki göze çarpan noktaları tespit etmesinden dolayı, nesne tespitinde çokça tercih edilen araçlardır. Algılama için LS-SVM uygulama kolaylığı ve bilgi aktarma yöntemlerine kolayca uyarlanabilen uygun model yapısı nedeniyle tercih edilmiştir. Seçilen üç kaynak sınıfı için LS-SVM yardımıyla sınıflandırıcı (dedektör) modelleri üretilmiştir. Hedef sınıfına ait detektörü üretmek için, oluşturulmuş bu kaynak sınıfı modellerinden bilgi aktarımı yapılmıştır. Bilgi aktarımının mantığı şu şekildedir: (1) BA için sadece tek bir kaynak modeli kullanıldığı durumda, yeni hedef sınıf, kaynak ve hedef sınıfına ait modellerin birbirine yakınlığı koşulu üzerinden, kaynak modelden adaptasyon yoluyla öğrenilmiştir; (2) tüm kaynak sınıfları BA için kullanıldığında ise, hedef sınıf modeli kaynak modellerin lineer birleşimine yakın olacak şekilde tüm kaynak modellerden adaptasyon yoluyla öğrenilmiştir. Kaynak modellere olan yakınlığı değerlendirmek ve hangi kaynak modelinden ne kadar bilgi transferi yapılacağına karar vermek için, birini hariç tutma hatasını (loo error: leave-one-out error) en aza indiren konveks bir en iyileme problemi çözülmüştür.

1.1. Tezin Amacı

Bu tez çalışması kapsamında, kuru kum içinde tüp, kuru kumtaşı içinde tüp ve ıslak kumtaşı içinde tüp sınıfları için oluşturulmuş sınıflandırıcı modellerinden bilgi aktarımı yaparak, az sayıda verisi bulunan ıslak kum içinde tüp sınıfı için oluşturulan sınıflandırıcı modelinin, hedef tespit başarımını artırmak amaçlanmıştır. Bu doğrultuda, T. Tommasi, F. Orabona ve B. Caputo 'nun "Çok Modelden Bilgi Aktarımı İle Birkaç Örnekten Kategorileri Öğrenme" başlıklı çalışmasındaki [2] bilgi aktarımı yöntemi takip edilmiş ve bu yöntem GPR verilerinden hedef tespiti için uyarlanmıştır. Farklı sınıflar için oluşturulmuş sınıflandırıcıların, birbirlerindeki verileri çapraz sezim oranlarının yeterli olmadığı, aynı şekilde, bir sınıf için sadece az sayıda veri ile oluşturulan sınıflandırıcıların da tatmin edici sonuçlar vermediği çalışma kapsamında görülmüştür. Oluşturulan sınıflandırıcılara eğitim

ve test giriş verisi sağlamak amacıyla, GPR görüntülerinden tanımlayıcı üretme yaklaşımı ortaya konmuş ve yöntemin başarılı olduğu görülmüştür.

1.2. Tezin Akışı

Bölüm 1’de, zaman ve mekan olarak imkanların kısıtlı olduğu bir durumda, daha önce öğrenilmiş bilgilerin, yeni verileri öğrenmedeki öneminden ve bu tez çalışmasında, bu özelliğin, üretilecek sınıflandırıcıda hedeflendiğinden bahsedilmiştir. Bu amaç için yapılan çalışmanın GPR veri kümesi üzerinde gerçekleştirildiği ve kullanılan GPR verilerinin nasıl üretildiği anlatılmıştır. Daha sonra, literatürde GPR verilerinden nesne tespiti için tercih edilen model tabanlı ve öznitelik tabanlı yaklaşımlardan ve GPR verileri için tercih edilen sınıflandırıcılardan bahsedilmiştir. Bölüm 1’de ayrıca, bu tez kapsamında tercih edilen yöntemlerden bahsedilmiştir. Bölüm 2’de, bu çalışma kapsamında kullanılan GPR verilerinin nasıl elde edildiği ve bölüm 3’te bu verilerin sınıflandırıcılarda kullanmak üzere ne tür ön işlemlerden geçirildiği ve nasıl anlamlandırıldığı detaylı bir şekilde anlatılmıştır. Bölüm 4’te, kullanılacak olan LS-SVM sınıflandırıcısının çalışma mantığı ve formülizasyonu verilmiş; iki boyutlu düzlemde, rastgele oluşturulmuş veriler üzerinde çalışma başarısı örneklendirilmiş ve bu tez kapsamındaki gerçek veriler üzerindeki başarısı grafikli olarak gösterilmiştir. Bölüm 5’te, bu tez çalışmasının ana hedefi olan bilgi aktarımının nasıl gerçekleştirildiği tüm teorik hesaplamaları ile anlatılmıştır. Bölüm 6’da BA için elde edilen sonuçlar ve bu sonuçların değerlendirilmesi ve bölüm 7’de de tüm tez çalışması boyunca elde edilmiş deneysel sonuçlar ve bunların değerlendirmesi verilmiştir.

2. GPR VERİLERİNİN OLUŞTURULMASI

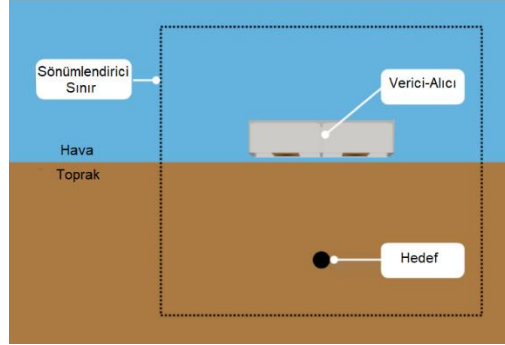
Toprak içerisine gömülü olan Mükemmel Elektrik İletkeni (PEC: Perfect Electric Conductor) malzemedeki bir tüp, vericisi vasıtasıyla yerin altına elektromanyetik sinyal gönderen ve alıcısı vasıtasıyla bu sinyalleri alan bir alıcı verici sistemi ile tespit edilecektir. EM alıcı verici sistemi, cismin içerisinde bulunduğu toprağın üzerinde belirli bir adım uzaklığı ile ilerlerken gönderdiği sinyallerden geri yansıyan kısımları toplayıp bu veriden görüntü elde etmektedir. Yerin altını görüntülemek amacıyla kullanılan bu alıcı verici sistem, yeraltı radarı olarak adlandırılmaktadır.

Yere nüfuz eden radar ya da diğer bir isimle Yeraltı Radarı (GPR: Ground Penetrating Radar) elektromanyetik dalga yardımıyla yerin altındaki nesnelere görüntülenmesinde veya tespit edilmesinde kullanılan bir cihazdır. EM dalgaları, ilerlediği yol boyunca farklı dielektrik özellikli (permittivity) ara yüzlerden geçerken dalganın bir kısmı ilerlemeye devam ederken bir kısmı da geri yansımaktadır. Yeraltı Radarının çalışma prensibi, yüksek frekanslı bir verici ile yerin altına gönderilen elektromanyetik dalgaların ara yüzlerden geri yansıyan kısımlarının bir alıcı vasıtasıyla alınmasına dayanmaktadır. Giga Hertz mertebesinde yüksek frekanslı EM dalgalarının kullanılması sayesinde sığ yeraltı gözlemlerinde kullanılmaktadır. Ayrıca kullanılan yüksek frekans ile yüksek çözünürlüklü ölçümler gerçekleştirilebilir. Gözlem yapılan toprağın iletkenlik durumu görüntü kalitesini etkileyen bir faktördür. Yeraltı ne kadar dirençli (az iletken) ise, yer radarı görüntüleri o kadar kaliteli olmaktadır. Nüfuz derinliğinin önemli olduğu durumlarda, ortamın mümkün olduğunca kuru (rutubetsiz) olması gerekmektedir.

Bu çalışmada, yeraltı radarı görüntüleri gprMax programı ile oluşturulmuştur. Bundan sonraki bölümde, gprMax programının çalışma ilkesi ve GPR görüntüsü oluşturulması için gerekli olan girdi dosyasının oluşturulması anlatılmıştır. Daha sonra, oluşturulan girdi dosyası ile çalıştırılan benzetim sonucunda elde edilen çıktı dosyası ve elde edilen GPR görüntüsü verilmiştir. Sonraki bölümlerde, GPR görüntüleri üzerinde yapılan işlemler; sınıflandırıcıda kullanılacak olan tanımlayıcıların elde edilmesi ve pozitif ve negatif veri kümelerinin oluşturulması anlatılmıştır. Daha sonra, sınıflandırıcı olarak kullanılan yöntem, bu yöntem ile elde edilen sonuçlar ve bilgi aktarımının bu yöntemle uyarlanması detaylı teorik açıklamalar ve formüller ile anlatılmış ve elde edilen sonuçlar ile tez sonlandırılmıştır.

2.1. Yeraltı Radarı Çalışma İlkesi

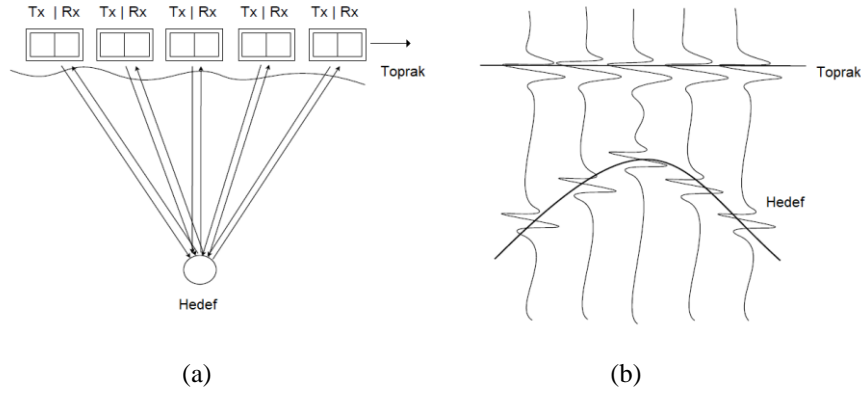
Yeraltı Radarının çalışması, Şekil 2’de gösterildiği üzere toprak içerisinde gömülü olan bir hedef nesneyi, EM verici-alıcı bir sistem ile tarayarak tespit etme prensibine dayanmaktadır. [29]. EM verici-alıcı, Giga Hertz frekanslarında Elektro Manyetik dalga gönderen bir verici ve gönderilen dalganın arayüzlerden geri yansıyan kısmını toplayan bir alıcı sistemden oluşmaktadır.



Şekil 2: Yeraltı Radarı Çalışma Ortamı

Yeraltı Radarı, hedefin gömülü olduğu toprağın üzerinde ilerlerken önceden belirlenmiş bir dalga formuna sahip EM sinyal gönderir. Bu sinyal gaussian, gaussiandot, ricker, sinüs vb. formlarında olabilir. EM dalga, parametrelerde bir değişiklik olmadığı müddetçe gönderildiği ortamda ilerler. Farklı materyaller farklı dielektrik özelliklerine sahip olacağı için, bunların kesişim ara yüzlerinde EM dalgaların bir kısmı geri yansır ve bir kısmı ilerlemeye devam eder. Şekil 2’de görüldüğü gibi, EM dalgalarının vericiden çıktıktan sonraki ilk karşılaştıkları ortam değişikliği, havadan toprağa geçme anıdır. Bu esnada sinyal ilk yansımasına maruz kalır ve yansıyan bu sinyal alıcı sistem tarafından alınır. Sinyalin ilerlemeye devam eden kısmı ise ikinci ortam değişimini yine topraktan farklı dielektrik özelliklere sahip hedef ile karşılaşınca gerçekleştirir ve aynı şekilde bu arayüzden de yansıyan sinyaller alıcı sistem tarafından alınır. Toprak üzerindeki bir noktada tek bir EM dalga ile yapılan bu işleme A-scan denir. Alıcı-verici sistemin toprak üzerinde bir adım uzaklığı kadar ilerlemesi ile bu işlem tekrarlanır ve yeni noktada tekrar bir A-scan işlemi gerçekleştirilir. Bu şekilde, alıcı-verici sistemin toprak üzerinde ilerlerken yaptığı A-scan işlemlerinin toplamı B-scan olarak adlandırılır [8]. Şekil 3’te, anlatılan bu durum resmedilmiştir [30]. Cismin alıcı-verici sistemin konumuna bağlı olarak görüntüsü hiperbol olarak elde edilmiştir. Alıcı-verici sistem, hedefe en yakın olduğunda, hedeften yansıyan sinyaller en kısa sürede geri alınır. Alıcı-verici sistem hedeften her iki yöne doğru

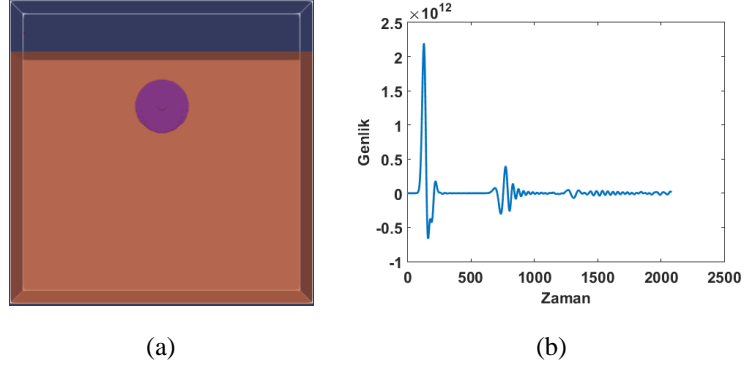
uzaklaştıkça, yansıyan sinyaller o ölçüde daha geç alınır. Böylece Şekil 3(b)'de görülen hiperbol elde edilir. Ayrıca Şekil 3(b)'de gözükten ve her noktada eşit süre ile alınmış olan sinyal, toprak ile havanın kesişim noktasında yansıyan sinyallerdir.



Şekil 3: Yeraltı Radarı ile hedef tespiti. (a) Alıcı-Verici ikilisi toprak üzerinde bir konumdayken, vericiden gönderilen EM dalgaları alıcı tarafından yakalanır ve bu işlem sabit mesafe adımları ile tekrarlanır. (b) Farklı konumlarda alınan sinyaller birleştirilerek GPR görüntüsü oluşturulur.

2.2. GPR Benzetimleri

Bu tez kapsamındaki çalışmada GPR verilerini elde etmek için gprMax programı kullanılmıştır. gprMax, elektromanyetik dalga yayılımını simüle eden ücretsiz bir yazılımdır. Sonlu Fark Zaman Alanı (FDTD: Finite-Difference Time-Domain) [31] [32] [33] yöntemini kullanarak Maxwell denklemlerini 3D olarak çözer. gprMax, GPR modellemesi için tasarlanmıştır, ancak aynı zamanda diğer birçok uygulamalar için elektromanyetik dalga yayılımını modellemek için de kullanılabilir [29]. gprMax programı ortam ve simülasyon parametrelerini içeren bir girdi dosyası üzerinde çalışarak A-scan verileri üretir. Daha sonra bu A-scan verileri birleştirilerek B-scan verisi elde edilir. Örnek olarak, Şekil 4'te toprak içerisinde gömülü olan bir küre ve çalıştırılan bir A-scan ile elde edilen sinyal grafiği verilmiştir.



Şekil 4: (a) Simülasyon ortamında oluşturulmuş küresel bir cisim. (b) gaussiandot dalga şekli kullanan verici ile alınan bir A-Scan sinyali

2.2.1. Girdi dosyası

Girdi dosyası herhangi bir metin belgesi formatındaki içerisinde gprMax programının anlayacağı komut satırlarında oluşan bir dosyadır. Dosya içerisinde, başında # sembolü olan satırlar program için kod satırı olarak değerlendirilmektedir. Başında # sembolü olmayan satırlar ise kullanıcı için yorum satırlarıdır ve program tarafından dikkate alınmaz. Bu tez çalışması kapsamında, girdi dosyasında kullanılan komutlar özet olarak Tablo 1’de, bu komutların anlamları ve örnek kullanımları da detaylı olarak Tablo 2’de verilmiştir.

Tablo 1: Girdi Dosyasında Kullanılan Komutlar (Özet)

Komut	Açıklama
material	Simülasyon ortamında kullanılan tüm malzemelerin elektriksel parametrelerini tanımlar
domain	Simülasyon ortamının x, y ve z yönlerindeki büyüklüklerini tanımlar
pml_cells	Simülasyon ortamının dış yüzlerine eklenip, EM dalgalarının soğurulmasını sağlayan katman kalınlığını tanımlar
fractal_box	Nesnenin gömülü olduğu toprağın sınırlarını belirler
add_surface_roughness	Oluşturulan toprak modeli için engebeli bir yüzey oluşturur
add_grass	Oluşturulan toprak modelinin üst yüzeyine çimen ekler
add_surface_water	Oluşturulan toprak modelinin üst yüzeyine su birikintileri ekler
sphere	Küre şeklinde bir nesne oluşturur
cylinder	Silindir şeklinde bir nesne oluşturur
cylindrical_sector	Silindir dilimi şeklinde bir nesne oluşturur
triangle	Üçgen prizma şeklinde bir nesne oluşturur
dx_dy_dz	Simülasyon ortamının x, y ve z yönlerinde ayrıklaştırılmasını sağlar
time_window	Toplam simülasyon süresini belirler
src_steps	B-scan taramalarında, simülasyonun her bir ardışık çalışmasında, EM vericinin toprak üzerinde belirtilen miktar kadar ilerlemesini sağlar

rx_steps	B-scan taramalarında, simülasyonun her bir ardışık çalışmasında, EM alıcının toprak üzerinde belirtilen miktar kadar ilerlemesini sağlar
waveform	Vericinin yaydığı EM sinyallerinin dalga şeklini belirler
hertzian_dipole	Simülasyondaki vericiyi tanımlamak için kullanılır
rx	Simülasyondaki alıcıyı tanımlamak için kullanılır

Tablo 2: Girdi Dosyasında Kullanılan Komutlar (Detay)

Malzeme Tanımlama Komutu	<p>#material: f1 f2 f3 f4 str1</p> <p>Kullanılacak olan malzemelerin elektriksel parametrelerinin tanımlandığı komuttur.</p> <p>f1: ϵ_r, bağıl elektrik geçirgenlik (relative permittivity)</p> <p>f2: σ, iletkenlik (conductivity) (Siemens/metre)</p> <p>f3: μ_r, bağıl manyetik geçirgenlik (relative permeability)</p> <p>f4: α, manyetik kayıp (magnetic loss) (Ohms/metre)</p> <p>str1: malzeme için kullanıcının verdiği tanımlayıcı isim.</p> <p>Simülasyonda örnek kullanım:</p> <p>#material: 10.0 1e-7 2.0 1e-2 dry_sand</p>
Ortam Tanımlama Komutu	<p>#domain: f1 f2 f3</p> <p>Simülasyon ortamının boyutunu belirtir, f1, f2 ve f3 sırasıyla ortamın x, y ve z yönlerindeki büyüklükleridir. Boyutlar metre cinsindedir.</p> <p>Simülasyonda örnek kullanım:</p> <p>#domain: 0.25 0.60 0.50</p>

<p>Ortam Tanımlama Komutu</p>	<p><u>#pml cells: i1 [i2 i3 i4 i5 i6]</u></p> <p>Simülasyon ortamının dış kenarlarına eklenecek olan, EM dalgalarının soğurulmasını sağlayan katman ve kalınlığını belirten komuttur. 6 kenar için ayrı ayrı belirtilebilir veya i1 ile hepsine aynı değer atanabilir. i1-i6 sırasıyla pozitif yönünde x, y, z ve negatif yönünde x, y, z kenarlarına tekabül eder.</p> <p>Simülasyonda örnek kullanım: #pml_cells: 20 20 20 20 20 20</p>
<p>Ortam Tanımlama Komutu</p>	<p><u>#fractal box: f1 f2 f3 f4 f5 f6 f7 f8 f9 f10 i1 str1 str2 [i2]</u></p> <p>Nesnenin gömülü olduğu toprağın sınırlarını belirlemek için kullanılan komuttur. f1 f2 f3 ile toprağın sol alt köşesi, f4 f5 ve f6 ile de sağ üst köşesi belirtilir. str1 kullanılan madde için, str2 de oluşturulan 3D toprak ortamı için kullanıcı tarafından belirlenen tanımlayıcılardır. i2 rastgele sayı üretici için kullanılan zorunlu olmayan değer parametresidir.</p> <p>Simülasyonda örnek kullanım: #fractal_box: 0 0 0 0.25 0.60 0.40 1.5 1 1 1 1 dry_sand soilBox 14</p>
<p>Ortam Tanımlama Komutu</p>	<p><u>#add surface roughness: f1 f2 f3 f4 f5 f6 f7 f8 f9 f10 f11 str1 [i1]</u></p> <p>Oluşturulan toprak modeli için engebeli yüzey oluşturmak için kullanılır. f1 f2 f3 toprak yüzeyinin sol alt köşesini, f4 f5 f6 da sağ üst köşesini belirtir. f10 ve f11 yüzey engebese için alt ve üst yükseklik limitlerini belirtir. str1 tanımlanan fractal_box için kullanılan tanımlayıcıdır. i1 rastgele sayı üretici için kullanılan zorunlu olmayan değer parametresidir</p> <p>Simülasyonda örnek kullanım: #add_surface_roughness: 0 0 0.4 0.25 0.60 0.40 1 1 1 0.40 0.405 soilBox 14</p>

<p style="text-align: center;">Ortam Tanımlama Komutu</p>	<p><u>#add_grass: f1 f2 f3 f4 f5 f6 f7 f8 f9 i1 str1 [i2]</u></p> <p>Oluşturulan toprak modelinin üst yüzeyine çimen eklemek için kullanılır. f1 f2 f3 oluşturulan çimen katmanı için sol alt köşeyi, f4 f5 f6 da sağ üst köşeyi belirtir. f9 f10 oluşturulacak çimenlerin boylarının alt ve üst limitini belirtir. i1 oluşturulacak çimen sayısını belirtir. str1 tanımlanan fractal_box için kullanılan tanımlayıcıdır. i1 rastgele sayı üretici için kullanılan zorunlu olmayan değer parametresidir.</p> <p>Simülasyonda örnek kullanım: #add_grass: 0 0 0.40 0.25 0.60 0.40 1 0.420 0.422 150 soilBox 14</p>
<p style="text-align: center;">Ortam Tanımlama Komutu</p>	<p><u>#add_surface_water: f1 f2 f3 f4 f5 f6 f7 str1</u></p> <p>Oluşturulan toprak modelinin üst yüzeyine su birikintileri eklemek için kullanılır. f1 f2 f3 oluşturulan su katmanı için sol alt köşeyi, f4 f5 f6 da sağ üst köşeyi belirtir. f7 oluşturulan su tabakası için yükseklik üst limitini belirtir. str1 tanımlanan fractal_box için kullanılan tanımlayıcıdır.</p> <p>Simülasyonda örnek kullanım: #add_surface_water: 0 0 0.4 0.25 0.60 0.4 0.402 soilBox</p>
<p style="text-align: center;">Nesne Oluşturma Komutu</p>	<p><u>#sphere: f1 f2 f3 f4 str1</u></p> <p>Küresel bir nesne oluşturmak için kullanılır. f1 f2 f3 oluşturulan kürenin merkezinin koordinatıdır. f4 kürenin yarıçapını belirtir. str1 kullanıcı tarafından oluşturulmuş ya da programın varsayılanları olarak oluşturulmuş malzemelerden birine ait tanımlayıcıdır.</p> <p>Simülasyonda örnek kullanım: #sphere: 0.125 0.45 0.33 0.005 sandstone_dry</p>

<p>Nesne Oluřturma Komutu</p>	<p><u>#cylinder: f1 f2 f3 f4 f5 f6 f7 str1</u></p> <p>Silindirik bir nesne oluřturmak için kullanılır. f1 f2 f3 dairesel yüzeylerden bir tanesinin merkezini; f4 f5 f6 da diđer yüzün merkezini belirtir. f7 oluřturulan silindirin yarıçapını belirtir. str1 kullanıcı tarafından oluřturulmuř ya da programın varsayılanları olarak oluřturulmuř malzemelerden birine ait tanımlayıcıdır.</p> <p>Simülasyonda örnek kullanım: #cylinder: 0.125 0.30 0.23 0.125 0.30 0.30 0.055 pec</p>
<p>Nesne Oluřturma Komutu</p>	<p><u>#cylindrical_sector: c1 f1 f2 f3 f4 f5 f6 f7 str1</u></p> <p>Silindirik dilim oluřturmak için kullanılır. c1 oluřturulan silindirik dilimin eksenini belirtir. f1 f2 silindirik sektör ekseninin merkezini; f3 f4 seçilen ekseninde oluřturulan nesne için alt ve üst limitlerini; f5 yarıçapı, f6 silindirik dilim için bařlangıç açısını; f7 de nesnenin kapladığı açığı belirtir. str1 kullanıcı tarafından oluřturulmuř ya da programın varsayılanları olarak oluřturulmuř malzemelerden birine ait tanımlayıcıdır.</p> <p>#cylindrical_sector: z 0.125 0.30 0.31 0.33 0.050 20 20 pec</p>
<p>Nesne Oluřturma Komutu</p>	<p><u>#triangle: f1 f2 f3 f4 f5 f6 f7 f8 f9 f10 str1</u></p> <p>Üçgen prizma yapıda bir nesne oluřturmak için kullanılır. f1 f2 f3 üçgen prizmaya ait birinci köşeyi; f4 f5 f6 üçgen prizmaya ait ikinci köşeyi; f7 f8 f9 da üçgen prizmaya ait üçüncü köşeyi belirtir. f10 oluřturulan üçgen prizmanın yüksekliđi için kullanılır. str1 kullanıcı tarafından oluřturulmuř ya da programın varsayılanları olarak oluřturulmuř malzemelerden birine ait tanımlayıcıdır.</p> <p>Simülasyonda örnek kullanım: #triangle: 0.105 0.28 0.30 0.140 0.30 0.30 0.125 0.35 0.30 0.025 sandstone_dry</p>

<p>Simülasyon Parametrelerini Belirleme Komutu</p>	<p>#dx dy dz: f1 f2 f3</p> <p>Simülasyon ortamının x, y ve z yönlerinde ayrıklaştırılmasını sağlayan komuttur. f1 f2 f3 sırasıyla x, y ve z yönlerindeki simülasyon adım mesafesini belirtir. Uzamsal ayrıklaştırma, azami müsaade edilen zaman adımını belirler. Simülasyon, kullanıcı tarafından belirlenen toplam simülasyon zamanına ulaşmak için, Δt zaman adımı kadar ilerleyerek devam eder. Zaman adımı için geçerli formül şu şekildedir.</p> $\Delta t \leq \frac{1}{c \sqrt{\frac{1}{(\Delta x)^2} + \frac{1}{(\Delta y)^2} + \frac{1}{(\Delta z)^2}}}$ <p>Burada, c ışık hızıdır. gprMax programı, Δt 'yi, girilen Δx Δy ve Δz değerlerini kullanarak bulur. Δx, Δy ve Δz değerlerinin küçük seçilmesi, Δt 'nin küçük çıkmasına sebep olur ki bu da belirlenen simülasyon süresine erişmek için daha fazla döngü demektir. Diğer taraftan daha küçük Δx, Δy ve Δz değerleri, simülasyonun daha gerçekçi olması anlamına gelecektir.</p> <p>Simülasyonda örnek kullanım: #dx_dy_dz: 0.002 0.002 0.002</p>
<p>Simülasyon Parametrelerini Belirleme Komutları</p>	<p>#time_window: f1</p> <p>Toplam simülasyon süresini belirlemek için kullanılan komuttur. f1 simülasyon süresini belirtir. Belirtilen süreye ulaşmak için, gprMax gerekli sayıda döngüyü gerçekleştirir. f1 yerine yazılan sayının değerine göre, bu gprMax tarafından döngü sayısı olarak da değerlendirilebilir. Toplam simülasyon zamanı ve döngü sayısı şu şekilde belirlenir.</p> $t = \Delta t N$ <p>Burada, t toplam simülasyon süresi, Δt simülasyon zaman adımı ve N de simülasyon için yapılan döngü sayıdır. Simülasyon süresi için girilen değer, elde edilen görüntünün z eksenindeki büyüklüğünü etkiler.</p> <p>Simülasyonda örnek kullanım: #time_window: 8e-9</p>

<p>Simülasyon Parametrelerini Belirleme Komutları</p>	<p><u>#src_steps: f1 f2 f3</u></p> <p>B-scan taramalarında, simülasyonun her bir ardışık çalışmasında, EM vericinin toprak üzerinde belirtilen miktar kadar ilerlemesini sağlayan komuttur. f1 f2 f3 sırasıyla x, y ve z yönlerindeki vericinin adım mesafesini belirtir.</p> <p>Simülasyonda örnek kullanım: #src_steps: 0.0 0.02 0.0</p>
<p>Simülasyon Parametrelerini Belirleme Komutları</p>	<p><u>#rx_steps: f1 f2 f3</u></p> <p>B-scan taramalarında, simülasyonun her bir ardışık çalışmasında, EM alıcının toprak üzerinde belirtilen miktar kadar ilerlemesini sağlayan komuttur. f1 f2 f3 sırasıyla x, y ve z yönlerindeki alıcının adım mesafesini belirtir.</p> <p>Simülasyonda örnek kullanım: #rx_steps: 0.0 0.02 0.0</p>

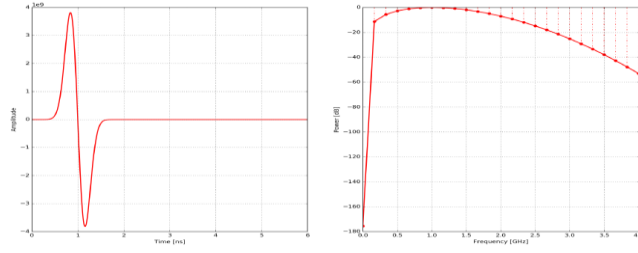
Alıcı-Verici
Tanımlama
Komutu

#waveform: str1 f1 f2 str2

Vericinin yaydığı EM sinyallerinin dalga şeklini belirleyen komuttur. str1 önceden tanımlanmış aşağıdaki dalga şekillerinden bir tanesi olabilir:

- gaussian: Gauss dalga.
- gaussiandot: Gauss dalganın birinci türevi.
- gaussiandotnorm: Normalleştirilmiş Gauss dalganın birinci türevi.
- gaussiandotdot: Gauss dalganın ikinci türevi.
- gaussiandotdotnorm: Normalleştirilmiş Gauss dalganın ikinci türevi
- ricker: Ricker (veya Meksika şapkası) dalga şekli.
- sine: Tek bir sinüs dalgası
- contsine: Devamlı sinüs dalgası

f1 oluşturulan dalga şeklinin genliğini, f2 de dalga şeklinin frekansını belirtir. str2, vericide kullanılmak üzere, oluşturulan dalga şekli için kullanıcı tarafından verilen isimdir. Bu çalışmada kullanılan dalga şekli, gaussiandot, aşağıda gösterilmiştir.



Simülasyonda örnek kullanım:

```
#waveform: gaussiandot 2 2.5e9 my_wave_form
```

#hertzian dipole: c1 f1 f2 f3 str1 [f4 f5]

Simülasyondaki vericiyi tanımlamak için kullanılan komuttur. c1 kaynağın polarizasyonunu belirtir ve üç koordinat ekseninden birisidir. f1 f2 f3 vericinin simülasyon ortamındaki konumunu belirtir. str1 daha önce oluşturulmuş dalga şeklidir.

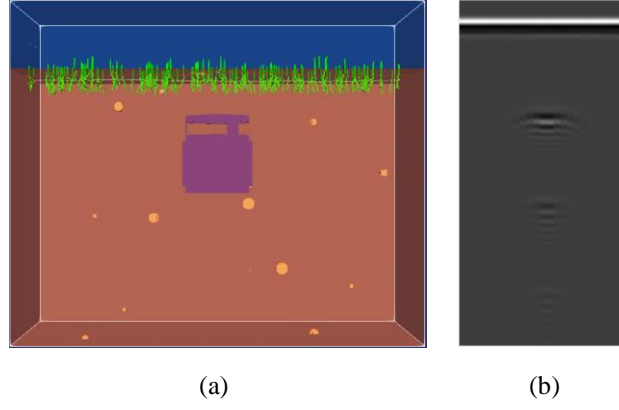
Simülasyonda örnek kullanım:

```
#hertzian_dipole: x 0.105 0.01 0.45 my_wave_form
```

<p><u>#rx: f1 f2 f3 [str1 str2]</u></p> <p>Simülasyondaki alıcıyı tanımlamak için kullanılan komuttur. f1 f2 f3 alıcının simülasyon ortamındaki konumunu belirtir.</p> <p>Simülasyonda örnek kullanım:</p> <p>#rx: 0.145 0.01 0.45</p>

2.2.2. Çıktı Dosyası

Tablo 2’ de tanımlanan komutlar ile oluşturulmuş bir girdi dosyasının tek bir atımlık EM dalgası ile çalıştırılması sonucunda elde edilen A-scan verisi görüntüsü Şekil 4(b)’de gösterilmiştir. Bu durumda, girdi dosyası içerisinde yer alan alıcı ve vericiyi ilerletme komutları bir anlam ifade etmemektedir. Oluşturulan girdi dosyası tek bir komutla birden fazla A-scan yapacak şekilde çalıştırılabilir. Yapılacak tarama sayısı, girilen komutla birlikte kullanıcı tarafından belirlenir. Bu durumda, simülasyonun her bir sinyal atımı için, alıcı ve verici belirtilen adım uzaklığı kadar belirtilen doğrultuda ilerler. Böylece, GPR cismin üzerinde ilerlerken farklı açılarda tarama gerçekleştirilmiş olur. Bu şekilde yapılan taramaya B-scan denilmektedir. Çalıştırılan B-scan taramasında, alıcı-vericinin her yeni konumu için bir A-scan verisi oluşturulur. Üretilmiş olan bu A-scan taramaları, bir komut ile tek bir dosya haline getirilebilmektedir. Üretilen yeni dosya aslında A-scan taramalarının bir matris içerisine atılmış halidir. Satır sayısı çalıştırılan tarama sayısına eşittir; sütun sayısı da belirtilen simülasyon çalıştırılma zamanı ile doğru orantılı bir değerdir. Şekil 5’te kuru toprak içerisinde tüp sınıfına ait bir ortamda çalıştırılmış B-scan taraması verilmiştir.



Şekil 5: Oluşturulan bir simülasyon ortamı ve elde edilen yeraltı radarı görüntüsü. (a) Kuru Toprak İçinde Tüp. (b) Üzerinde herhangi bir işlemin yapılmadığı B-scan taraması görüntüsü

2.3. Hedef ve Kaynak Verileri

Şekil 5’te görüldüğü gibi, toprak içerisine gömülü tüpün tespit edilmesi gerçekleştirilecek sınıflandırma çalışmasıdır. Eğer yeterli sayıda veri toplanabiliyorsa, bu veriler ile sınıflandırma algoritması eğitilir ve elde edilen model üzerinden yeni veriler sınıflandırma işlemine tabi tutulur. Ama eğer yeterli sayıda veri toplanamıyorsa bu durumda benzer sezimlerde bulunan modellerden bilgi aktarımı sayesinde çok az veri ile yeni sınıflandırıcı modeli oluşturulur. Burada yapılan çalışmanın temelini de bu oluşturmaktadır.

Bilgi aktarımının yapılacağı sınıf hedef sınıf (target class), bilgi aktarımının yapıldığı sınıf (lar) da kaynak sınıf (lar) olarak adlandırılmaktadır (source class). Bu çalışmada, aynı nesnenin farklı ortamlarda yer alması durumu farklı sınıfları oluşturmaktadır. Ortam parametreleri (ϵ_r) ve (μ_r) değerlerindeki farklılıklar göz önünde bulundurularak oluşturulan sınıflar şunlardır.

- Kuru kum içerisinde tüp
- Islak kum içerisinde tüp
- Kuru kumtaşı içerisinde tüp
- Islak kumtaşı içerisinde tüp

Yukarıdaki sınıflara tekabül eden ortamlara ait ϵ_r ve μ_r değerleri şu şekildedir.

Tablo 3: Kullanılan sınıflara ait ortamların parametreleri

Sınıf	Bağıl elektrik geçirgenlik (relative permittivity) (ϵ_r)	Bağıl manyetik geçirgenlik (relative permeability) (μ_r)
Kuru kum içerisinde tüp (Tube burried in dry sand)	10-30	2-6

Islak kum içerisinde tüp (Tube burried in wet sand)	2-5	10-30
Kuru kumtaşı içerisinde tüp (Tube burried in dry sandstone)	5-10	2-5
Islak kumtaşı içerisinde tüp (Tube burried in wet sandstone)	5-10	5-10

PEC malzemedan oluşturulan tüp nesnesinin bu ortamlar içerisinde yer aldığı durumlardan elde edilen farklı sınıflar arasında bilgi aktarımı sonuçları gözlemlenecektir.

Her bir sınıf için toplanan veri sayıları aşağıdaki gibidir

- Kuru kum içerisinde tüp : 56 resim (504 tanımlayıcı)
- Islak kum içerisinde tüp : 54 resim (486 tanımlayıcı)
- Kuru kumtaşı içerisinde tüp : 67 resim (603 tanımlayıcı)
- Islak kumtaşı içerisinde tüp : 50 resim (450 tanımlayıcı)

3. GPR VERİSİNİN İŞLENMESİ VE ÖZNETELİK ÇIKARIMI

Sınıflandırıcıda kullanılacak olan tanımlayıcıları elde etmek için, ilk olarak, özneteliklerin elde edilmesi gerekmektedir. Tanımlayıcılar, elde edilecek bu özneteliklerin konum bilgisinden yararlanarak oluşturulmaktadır. Bu bölümde, öznetelikleri verimli bir şekilde elde etmek için, GPR görüntüsü üzerinde yapılacak ön işlemler ve kullanılacak olan öznetelik elde etme yöntemi anlatılacaktır. Daha sonra, tanımlayıcıların ve veri kümelerinin elde edilmesi anlatılacaktır.

3.1. Yer yansımalarının çıkartılması

Şekil 5 (b)'de yer alan B-scan taramasında, en yukarıdaki düz beyaz çizgi vericiden çıkan EM dalgalarının maruz kaldığı ilk yansımalarıdır ve bu yansımaların gerçekleştiği yer toprağın yüzeyidir. Alıcı-verici ve yer arasındaki mesafe kısa olduğu için bu yansımaların genliği oldukça büyüktür. Görüntünün tamamına normalleştirme yapıldığı için diğer yansımalar sönük kalmaktadır. Görüntünün tamamından yerden yansıyan kısımların çıkartılması ile geri kalay toprak altındaki görüntü daha belirgin hale gelmektedir. Şekil 6'dan da görüleceği üzere, en yüksek sinyal genliğine ait yer yansıması çıkartıldığı için ve tüm görüntü verileri üzerinde normalleştirme yapıldığı için arka plan tonunda açılma olmaktadır.

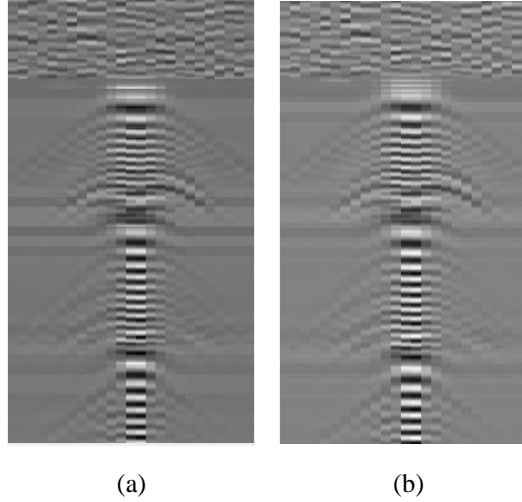


Şekil 6: Yer yansımalarının çıkartılmasından sonra B-scan taraması.

3.2. Görüntü Beyazlaştırma (Whitening)

Şekil 6'da tüpün varlığı ile oluşan hiperbol benzeri şekiller yukarıdan aşağıya doğru sönükleşmektedir. Bunun sebebi, EM dalgalarının mesafenin karesi ile ters orantılı olarak azalmasından dolayı göreceli olarak daha uzakta olan ara yüzlerden yansıyan sinyallerin

genliğinin azalmış olmasıdır. Bu durumu ortadan kaldırmak için yapılacak olan işlem, görüntü normalleştirilmedir. Bu çalışmada iki aşamalı görüntü normalleştirme yapılmaktadır. Birinci aşamada 10 satırdan oluşan görüntü parçaları, örtüşmeyecek şekilde normalleştirilirken, ikinci aşamada, ilk aşamada oluşturulan görüntü parçalarının kesişimi olan 10 satırlık görüntü parçaları ikinci normalleştirme işlemine tabi tutulmaktadır. Bu işlemler sonucunda Şekil 6'da yer alan Yeraltı Radarı görüntüsü Şekil 7'deki gibi olmaktadır.



Şekil 7: GPR görüntüsünde yer yansımaları çıkartıldıktan sonra yapılan beyazlatma işleminin sonucu. (a) Satır blokları arasında örtüşme yokken (birinci aşama) beyazlatma. (b) Satır blokları arasında %50 örtüşme varken (ikinci aşama) beyazlatma.

Yeraltı Radarı görüntü parçası içindeki her bir pikselin değerine x dersek, toplam N adet pikselden oluşan görüntü parçasına ait ortalama piksel değeri (μ), görüntü parçasına ait standart sapma (σ) ve normalleştirilmiş piksel değerleri (z) aşağıdaki formüllerle elde edilir.

$$\mu = \frac{1}{N} \sum_{i=1}^N x_i \quad (2.1)$$

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2 \quad (2.2)$$

$$z = \frac{x - \mu}{\sigma} \quad (2.3)$$

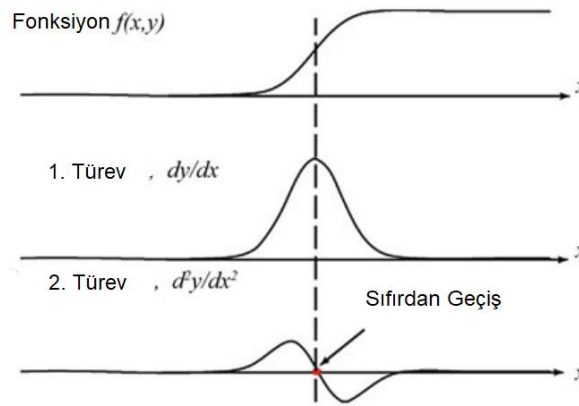
3.3. SIFT Öznitelikleri

SIFT, anahtar nokta (keypoint) olarak adlandırılan, bir görüntüdeki belirgin ve durağan özellikteki noktaları tespit etmek için kullanılan bir tekniktir. SIFT, aynı zamanda tespit edilen bu noktalar etrafındaki görüntü parçasının özelliğini niteleyen bir tanımlayıcı bilgi sağlar. Bu tanımlayıcılar, resmin ölçeğindeki değişimlerden ve resmin dönmesinden etkilenmezler, bu yüzden bu yöntemin ismi ölçekten bağımsız öznitelik dönüşümüdür.

Bir resimdeki SIFT özniteliklerini elde etmenin en önemli adımı, resimdeki bu anahtar noktaların bulunmasıdır. Bu anahtar noktalarının bulunmasında izlenecek yöntem, 2.3.1 alt başlığında detaylı olarak anlatılmaktadır. Anahtar noktaları ve ölçekleri tespit edildikten sonra, her bir kilit noktaya komşu veri noktalarına dayalı bir yön bilgisi ve bir tanımlayıcı atanır. Bu tez kapsamında, SIFT özniteliklerinin, bahsedilen ölçek, yön ve tanımlayıcı bilgileri kullanılmamış, sadece anahtar noktaların elde edilmesinden yararlanılmıştır.

3.3.1. Kapalı Eğrilerin Tespit Edilmesi

Her görüntü az veya çok sayıda kenar veya kapalı eğri içerir. Kapalı eğriler; dikdörtgen, daire gibi simetrik, ya da örneğin, eşyükselti eğrileri gibi asimetrik yapıda olabilir. Bir görüntüdeki bu şekiller, ilk önce onun karşıt kenarlarının bulunup daha sonra bulunan kenarların birbirine olan mesafesi ile orantılı olan ölçeğinin tespit edilmesi ile bulunur. Şekil 8’de görüleceği üzere, bir görüntüdeki kenarlar, onun ikinci dereceden türevini alarak kolaylıkla bulunabilir [34]. Görüntüdeki bir kenarın ikinci dereceden türevi, o noktadaki sıfırdan geçişi verir.



Şekil 8: Tek Boyutlu Fonksiyonda Kenar Tespit Etme. (üst) Resme ait piksel değerlerinin fonksiyonu. (alt) Fonksiyonun ikinci dereceden türevi, sıfırdan geçiş bilgisini vererek resimde yer alan kenarları bulmaktadır.

Bir görüntüde ikinci türev alma, Laplace operatörünü kullanarak yapılır. f görüntüsünün her bir boyutunu x ve y olarak isimlendirirsek, Laplace operatörü şu şekilde yazılır.

$$\nabla^2 = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (2.4)$$

Merkezi fark teoreminden yararlanarak [35], bu formül, uzamsal alanda şu şekilde ayrıklaştırılabilir (discretize).

$$\nabla^2(x_i, y_j) = \left[\frac{f(x_{i+1}, y_j) + f(x_{i-1}, y_j) - 2f(x_i, y_j)}{h^2} \right] + \left[\frac{f(x_i, y_{j+1}) + f(x_i, y_{j-1}) - 2f(x_i, y_j)}{h^2} \right] \quad (2.5)$$

Bir piksel görüntü çözünürlüğü düşünüldüğünde, h değeri olarak 1 alınır. Laplace operatörü, (2.5) eşitliğine bakılarak grafiksel formda, ayrıklaştırılmış fonksiyona uygulanacak uzaysal katsayılar cinsinden aşağıdaki gibi yazılabilir.

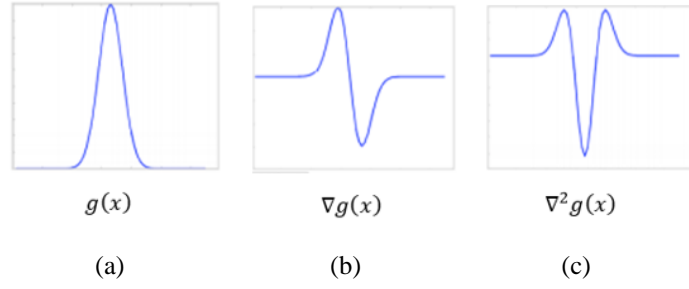
$$\nabla^2 f(x, y) = \begin{bmatrix} & +1 & \\ +1 & -4 & +1 \\ & +1 & \end{bmatrix} \quad (2.6)$$

Türev filtreler gürültüye çok hassas olduğundan, Laplace operatörünü uygulamadan önce, görüntünün (örneğin Gauss filtresini kullanılarak) pürüzsüz hale getirilmesi yaygındır. Konvolüsyon operatörünün birleşme özelliğinden yola çıkarak, Gauss filtrelenmiş fonksiyonun Laplace'ı, Laplace'ı alınmış Gauss (LoG) filtreli fonksiyona eşittir diyebiliriz. Buradan hareketle, görüntülere ilk olarak Gauss filtresini uygulayıp arkasından Görüntünün Laplace'ını almak yerine, Laplace'ı alınmış Gauss filtresini tek operatör olarak görüntülere uygulayabiliriz. Bir boyutlu durumda, Gauss fonksiyonu ve onun birinci ve ikinci türevleri aşağıdaki formüllerde ve Şekil 9'da verilmiştir [36].

$$g(x) = e^{\frac{-x^2}{2\sigma^2}} \quad (2.7)$$

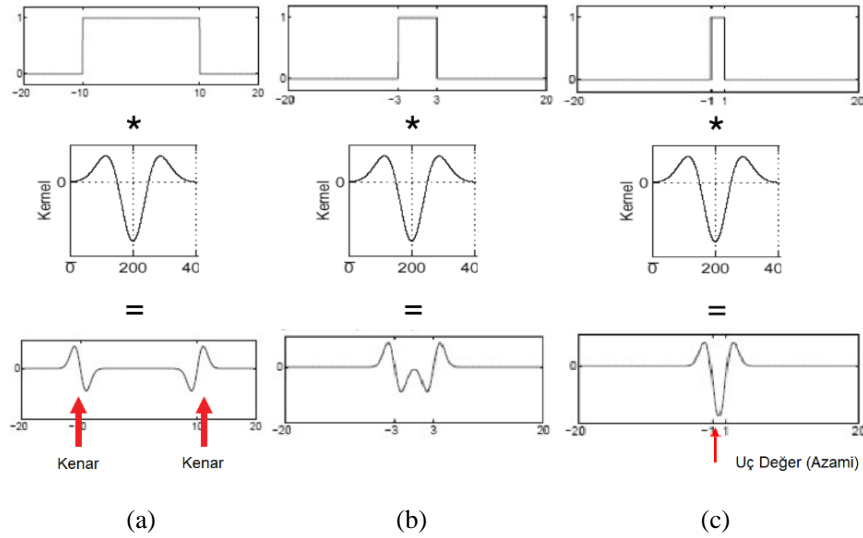
$$\nabla g(x) = \frac{-x}{\sigma^2} e^{\frac{-x^2}{2\sigma^2}} \quad (2.8)$$

$$\nabla^2 g(x) = \left(\frac{x^2}{\sigma^4} - \frac{1}{\sigma^2} \right) e^{\frac{-x^2}{2\sigma^2}} \quad (2.9)$$



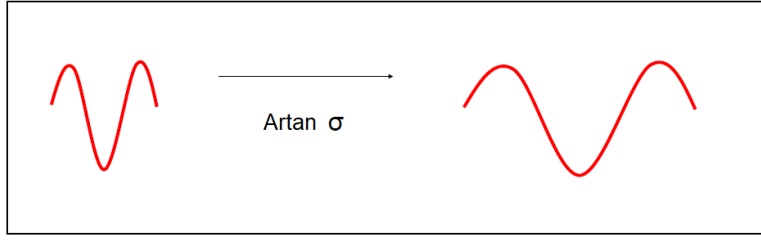
Şekil 9: Resimde yer alan gürültüleri ortadan kaldırmak için kullanılan (a) Gauss filtresi, (b) birinci türevi ve (c) ikinci türevi.

Dikdörtgen bir fonksiyonu, Gauss'un Laplace'ı operatörü ile konvolüsyon işlemine tabi tutarsak, fonksiyonun kenarlarında sıfırdan geçişleri elde ederiz. Eğer dikdörtgenin ölçüsü küçülürse, bulduğumuz sıfırdan geçişler birbirine yaklaşır. Şekil 10'da görüldüğü gibi şeklin büyüklüğünü azaltmaya devam edersek birbirine yaklaşan sıfırdan geçişler bir noktada üst üste binerler ve değer olarak uç nokta oluştururlar [37]. Dikdörtgen şeklin ölçüsü küçülmeye devam ederse, üst üste örtüşmüş olan sıfırdan geçişlerin oluşturduğu uç değer azalmaya başlar.



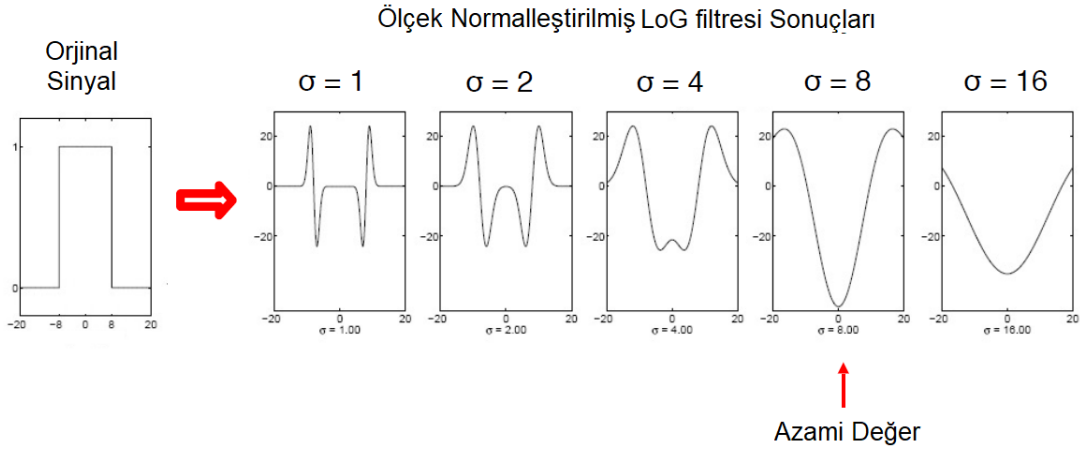
Şekil 10: LoG filtrenin bir kapalı eğriye (dikdörtgen) uygulanması. (a) Resimde yer alan dikdörtgen ve uygulanan LoG filtre ölçükleri uyumlu değilse sıfırdan geçişler birbirinden uzaktır. (c) Şekil ve filtre ölçükleri uyumlu ise, sıfırdan geçişler üst üste biner ve yerel uç değer oluşturur.

Aynı mantıkla, kapalı eğrinin büyüklüğü aynı kalarak LoG filtrenin boyutunu değiştirerek de sıfırdan geçişleri birbirine yaklaştırabilir ve uygun bir büyüklükte bu sıfırdan geçişleri örtüştürebiliriz. Bu şekilde, LoG filtrelenmiş resimde göreceli olarak uç noktaları veren noktalar ile resimdeki kapalı eğriler tespit edilmiş olur. LoG fitreyi elde etmek için kullanılan Gauss kernelinin varyansı (σ) değiştirilerek farklı ölçüklerde LoG filtre elde edilir.



Şekil 11: Gauss filtresinin varyansı artırılarak LoG Filtresinin boyutunun artırılması.

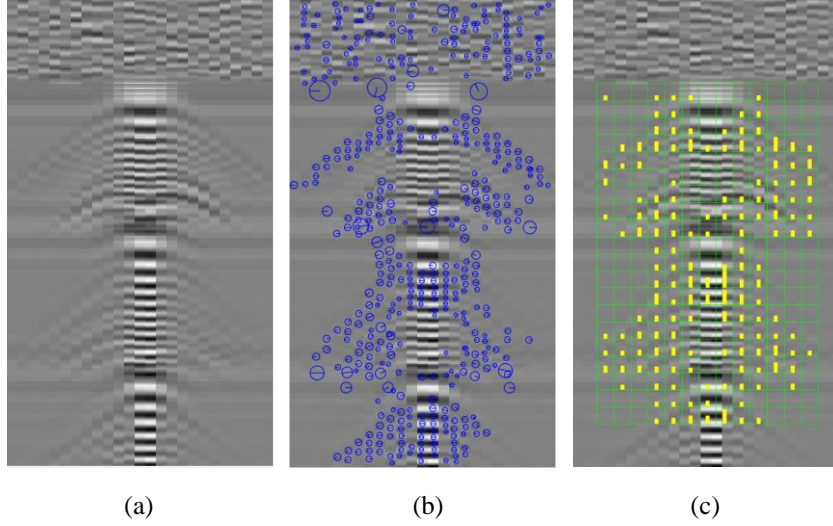
Şekil 12’de görüldüğü üzere [37], bir kapalı eğriyi (dikdörtgen) tespit etmek için, artan büyüklükte LoG filtreler ile fonksiyon filtrelenir ve bir önceki ve bir sonraki filtreler göre daha yüksek uç değerini veren σ değerine sahip LoG filtre uygun operatör olarak belirlenir. Böylece kapalı eğri, σ değeri ile orantılı olan büyüklük bilgisi ile birlikte tespit edilmiş olur.



Şekil 12: Kapalı eğriyi tespit etmek için LoG filtre boyutunun değiştirilmesi. LoG filtrenin boyutu artırılarak resme uygulanır. Uç değer elde edilmişse, kapalı eğri ve boyutu bulunmuş olur.

3.4. GPR Görüntüsünden Öznitelik Elde Etme

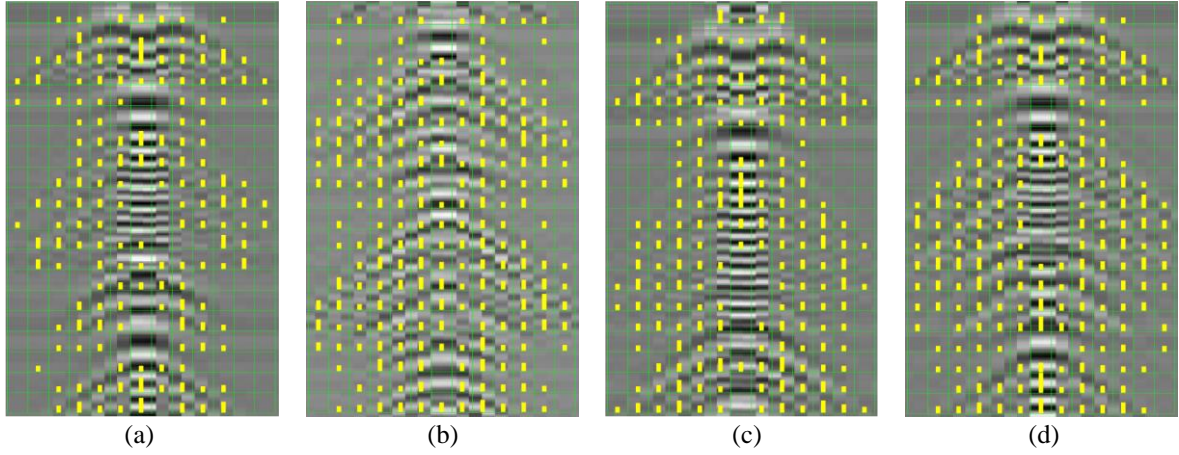
Toprağa gömülmüş bir nesnenin Yeraltı Radar görüntüsü, Şekil 13 (a)'da görülebileceği gibi, temel olarak dikdörtgensel şekillerden oluşmaktadır. Bu kapalı eğriler tespit edilir ve o görüntünün imzasını temsil edecek şekilde kodlanırsa, benzer görüntülerin sınıflandırılması iyi bir sınıflandırıcı ile kolay bir iş olacaktır.



Şekil 13: (a) GPR Görüntüsünden (b) Anahtar noktalarının ve (c) Tanımlayıcının elde edilmesi

Yer yansıması kaldırılmış, normalleştirilmiş ve beyazlatılmış GPR görüntüsü üzerinde anahtar noktalarının elde edilmesi Şekil 13(b)'de gösterilmiştir. Şekilde bulunan noktalar, vlfeat [38] aracı kullanılarak elde edilmiş SIFT anahtar noktalarıdır. Anahtar noktalar elde edildikten sonraki hedef, LS-SVM sınıflandırıcıda kullanılacak olan tanımlayıcının elde edilmesidir. Bu amaçla, görüntü üzerinde nesnenin bulunduğu bölgeye bir pencere yerleştirilir ve bu pencere hücrelere bölünür. Her bir pencereye, içerisinde yer alan anahtar nokta sayısı değeri atanır. Pencere 20x13 hücreden oluşmaktadır. Böylece, her bir GPR görüntüsü, 20x13 hücreden oluşan bir tanımlayıcı verir ve her hücre o hücreye ait anahtar noktaların sayısını belirtir ve bir tanımlayıcı, 260 elemanlı bir vektörle temsil edilir. Sınıflandırıcıyı eğitmek için tüm tanımlayıcılar, kaynak ve hedef görüntülerden toplanır. Anahtar noktalarının boyutları, sayıları ve konumları, farklı nesnelere ya da farklı ortamlar ile alınmış resimlerde farklılık gösterir. Bu nedenle, her bir resim kendine has belirli bir tanımlayıcı koduna sahip olmuş olur. 3 tane kaynak sınıf ve bir tane hedef sınıf olmak üzere 4 tane sınıf bulunmaktadır. Her bir sınıf, kendine has bir anahtar nokta dağılım imzasına sahiptir ve sınıflandırma işlemi bu temel üzerine dayanmaktadır. Şekil 14'te ilk üç tanımlayıcı görüntüsü kaynak sınıflarına ait verilerden, dördüncü görüntü de hedef sınıfına ait veriden elde edilmiştir. Tanımlayıcılar, aynı ortam üzerinde sadece toprak tipi değiştirilerek elde edilmiş simülasyon çıktılarından elde edilmiştir. Yukarıda bahsedildiği gibi, her bir görüntü ön işlemden geçirildikten sonra SIFT öznitelikleri elde edilmiş ve bunların resim üzerine oturtulan hücreli yapıdaki penceredeki konum ve sayılarına göre

tanımlayıcılar oluşturulmuştur. Şekilden de görüleceği üzere, farklı toprak tipleri için birbirinden ayırt edilebilir tanımlayıcılar oluşmuştur.



Şekil 14: Farklı toprak tipleri içinde yer alan tüp ile elde edilmiş tanımlayıcılar. (a) Kuru kum içinde tüp. (b) Kuru kumtaşı içinde tüp. (c) Islak kumtaşı içinde tüp. (d) Islak kum içinde tüp (hedef sınıf).

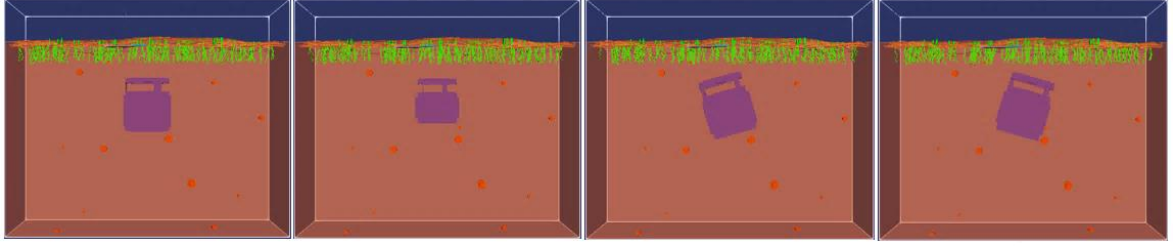
3.5. Pozitif Veriler

Nesnenin sezimini gerçekleştirmek için kullanılan algoritma, LS-SVM'dir. Algoritmanın eğitiminde kullanılmak üzere pozitif veri elde etmek için, 3 seviye çeşitlilik uygulanmıştır.

1. Ortam parametreleri değiştirilmiştir: Bunlar, ortama ait ϵ_r (permittivity) ve μ_r (permeability) değerleridir.
2. Nesnenin ortam içerisindeki özellikleri (boyutu, şekli, duruş açısı vb.) çeşitlendirilmiştir.
3. Tanımlayıcı oluşturulurken GPR görüntüsü üzerine yerleştirilen pencere, x ve y eksenlerinde kaydırılmıştır.

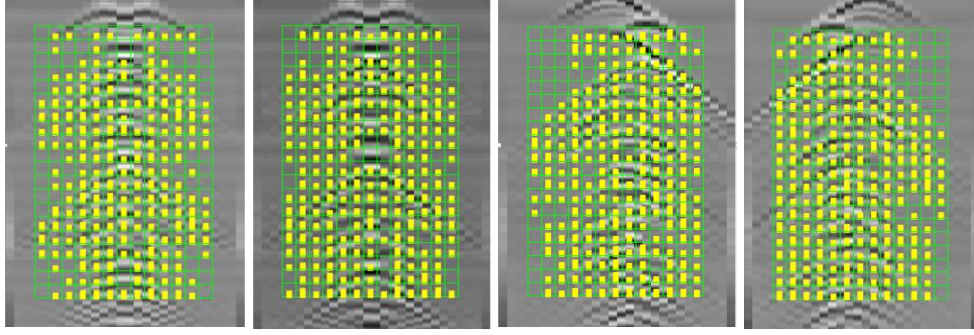
3.5.1. Birinci Seviye Çeşitlilik

Ortam içerisinde gömülü olan cismin fiziksel özelliklerinin değiştirilmesi ile elde edilen çeşitliliktir. Cismin boyutu, şekli, ortam içerisindeki eğikliği vb. gibi özelliklerin farklılaştırılması ile elde edilir. Bu şekilde nesnenin fiziksel özelliklerinde yapılan değişiklik ile görüntü sayısı artırılmıştır. Şekil 15'te soldan sağa cismin referans alınan biçimi, boyutunun artırılmış biçimi, sağa yatmış ve sola yatmış biçimleri yer almaktadır. Görüntüler, kuru kumtaşı içerisinde gömülü PEC malzemeden piknik tüplerine aittir.



Şekil 15: Pozitif veri toplamak için yapılan birinci seviye çeşitlilik. Tespit edilecek nesnenin boyutu, şekli, eğikliği vb. değiştirilerek elde edilir.

Bu ortamlar için çalıştırılan GPR benzetim sonucuna göre elde edilen SIFT tanımlayıcıları Şekil 16’da verilmiştir. Benzetimler arasındaki tek fark nesneye ait fiziksel özelliklerin Şekil 15’te görüldüğü gibi değiştirilmiş olmasıdır, bunun dışındaki tüm girdiler aynıdır. Soldan sağa her bir tanımlayıcı, Şekil 15’teki resimlerde soldan sağa karşılık gelen cisme aittir.



Şekil 16: Birinci seviye çeşitlilik ile elde edilen tanımlayıcılar. Şekil 15’te yer alan nesne ve ortamlara ait simülasyonla elde edilmiş tanımlayıcılar burada aynı sıra ile yer almaktadır.

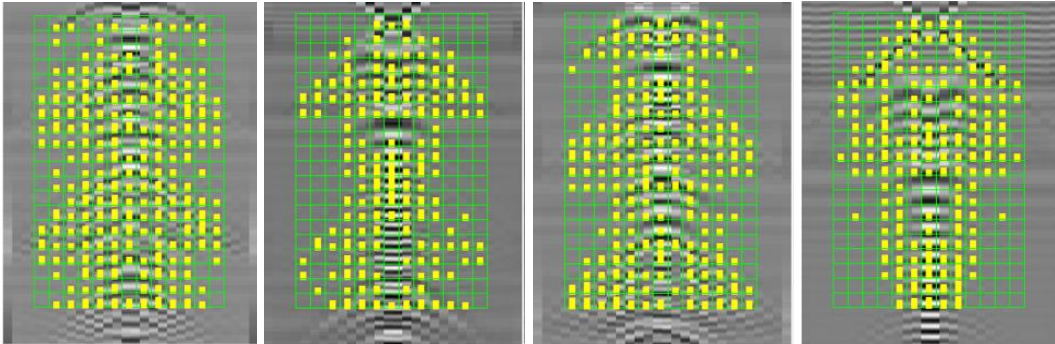
Şekilden görüleceği üzere, ortamın parametrelerinin değişmemiş olmasına rağmen cisme ait fiziksel özelliklerin değişmiş olması oldukça farklı tanımlayıcıların oluşmasına neden olmuştur. Bu şekilde yapılan çeşitlilik ile birlikte, elde edilecek olan sınıflandırıcının, cismin boyutunda ve toprak içerisine yerleştirilmesinde oluşabilecek (eğiklik gibi) değişikliklere adapte olabilmesi sağlanmış olacaktır. Bu sayede, cismin boyutuna ve toprağa gömülmesindeki açısal değişimlerine bakmaksızın tespit edilmesi gerçekleştirilmiş olacaktır.

3.5.2. İkinci Seviye Çeşitlilik

Cismin içinde bulunduğu ortamın parametrelerinin değiştirilmesi ile elde edilen çeşitliliştir. GPR verisi oluşturulurken kullanılan giriş dosyası içerisinde, ortama ait 4 parametre bulunmaktadır.

- ϵ_r : bağıl elektrik geçirgenlik (relative permittivity)
 σ : iletkenlik (conductivity)
 μ_r : bağıl manyetik geçirgenlik (relative permeability)
 α : manyetik kayıp (magnetic loss)

Bu parametrelerden σ ve α 'nın değişimi oluşturulan GPR verisi üzerinde belirgin bir etki oluşturmamaktadır. Bu yüzden sadece ϵ_r ve μ_r değerleri değiştirilerek cismin bir ortam içindeki GPR benzetimleri elde edilmektedir. Örnek olarak kuru kumtaşı, ϵ_r değeri olarak 5-10 arası değerleri, μ_r değeri olarak da 2-5 arasındaki değerleri alabilmektedir. Parametreler için belirtilen bu aralıklardan rastgele değer çiftleri seçilerek oluşturulan kombinasyonlar ile benzetimler yapılarak pozitif görüntü sayısı artırılmıştır. Şekil 17'de kuru kumtaşı içerisine gömülü PEC malzemeden piknik tüpüyle elde edilmiş GPR verisinden üretilmiş tanımlayıcı görüntüleri verilmiştir. Benzetimler arasındaki tek fark ϵ_r ve μ_r değerlerinin değiştirilmiş olmasıdır, bunun dışındaki tüm girdiler aynıdır. Şekilde soldan sağa ϵ_r ve μ_r değerleri: 5-2, 5-6, 8-2, 8-6 çiftleridir.



(a) $\epsilon_r = 5, \mu_r = 2$

(b) $\epsilon_r = 5, \mu_r = 6$

(c) $\epsilon_r = 8, \mu_r = 2$

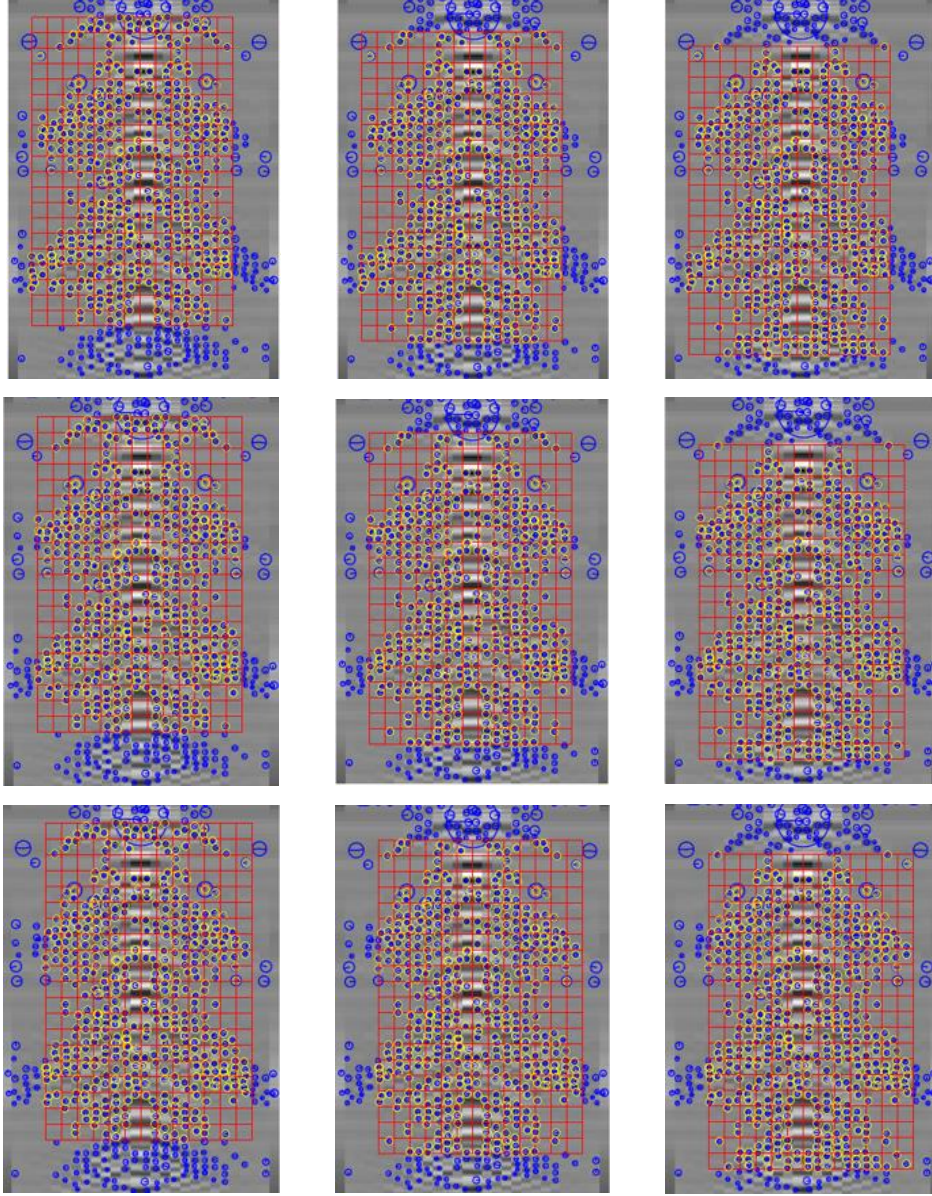
(d) $\epsilon_r = 8, \mu_r = 6$

Şekil 17: Kuru kumtaşı toprak içerisine gömülü iletken tüpe ait ikinci seviye çeşitlilik ile elde edilen tanımlayıcılar. Simülasyonlardaki tek değişen parametre, toprağa ait ϵ_r ve μ_r değerleridir.

Şekilden görüleceği üzere, cisme ait özelliklerin değişmemiş olmasına rağmen ortamın parametrelerindeki değişimler oldukça farklı tanımlayıcıların oluşmasına neden olmuştur. Bu şekilde yapılan çeşitlilik ile birlikte, elde edilecek olan sınıflandırıcının, cismin gömülü olduğu ortamın parametrelerinin bölgesel değişimlerinden kaynaklanabilecek farklılıklara adapte olabilmesi sağlanmış olacaktır. Bu sayede, cismin gömülü olduğu toprağın özelliklerinin bölgesel değişimine bakmaksızın tespit edilmesi gerçekleşmiş olacaktır.

3.5.3. Üçüncü Seviye Çeşitlilik

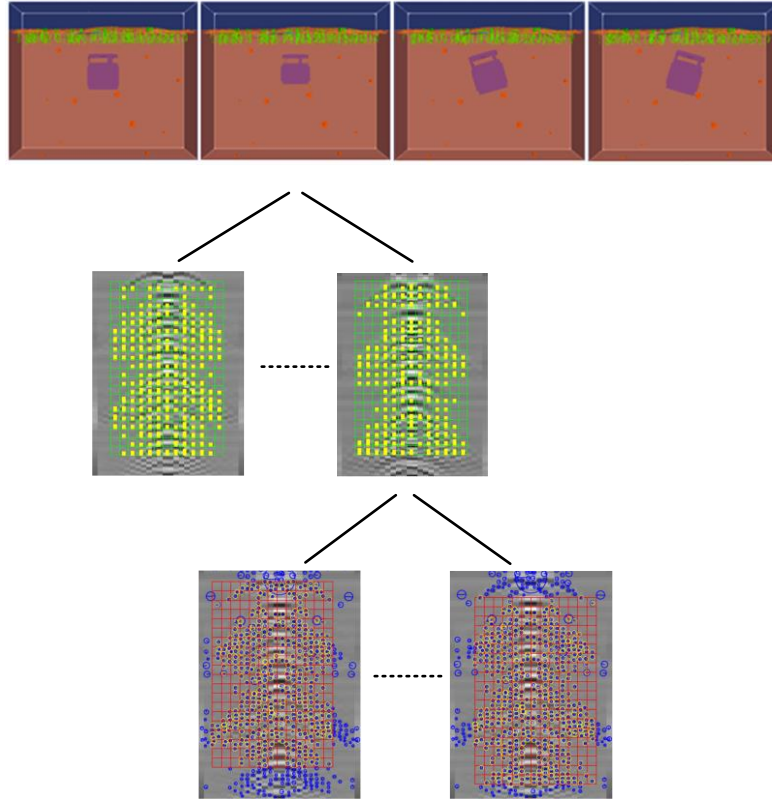
GPR verisi üzerinde oluşturulan SIFT özneteliklerini seçmek için kullanılan pencerenin, x ve y eksenlerinde kademeli olarak kaydırılması ile elde edilen çeşitliliğdir. Bu yöntem, pencereyi 3 adım x-ekseninde 3 adım da y-ekseninde kaydırarak, tek resimden 9 farklı tanımlayıcı elde edilmesini sağlamaktadır ve böylece pozitif görüntü sayısı artırılmıştır. Şekil 18'de merkezdeki resim referans olacak şekilde x eksenindeki kaydırmalar 10 sütundan y eksenindeki kaydırmalar da 15 satırdan oluşmaktadır. Pencereyi oluşturan hücreler 16 satır ve 16 sütundan oluşmaktadır. Dolayısı ile x ekseninde ilk kaydırma yaklaşık yarım hücre genişliğinde, ikinci kaydırma da bir hücre genişliğinden fazla kaydırmaya karşılık gelir. Diğer taraftan y ekseninde kaydırma birinci kaydırmada yaklaşık bir hücre yüksekliğine, ikinci kaydırmada da iki hücre yüksekliğine kaydırmaya karşılık gelir.



Şekil 18: Üçüncü seviye çeşitlilik için elde edilen pencereler. Merkezdeki resimde yer alan pencere referans olacak şekilde, solundaki ve sağındaki resimlerde pencere y ekseninde; yukarıdaki ve aşağıdaki resimlerde de x ekseninde kaymaktadır.

Şekilden görüleceği üzere, ne cismin kendisinde ne de cismin içerisinde bulunduğu ortamın özelliklerinde değişim olmamasına rağmen, pencere kaydırma, aynı cisme ait farklı tanımlayıcıların oluşmasına neden olmuştur. Bu şekilde yapılan çeşitlilik ile birlikte, elde edilen sınıflandırıcının, detektör veya ölçümleme kaynaklı ofsetlere adapte olabilmesi sağlanmış olacaktır.

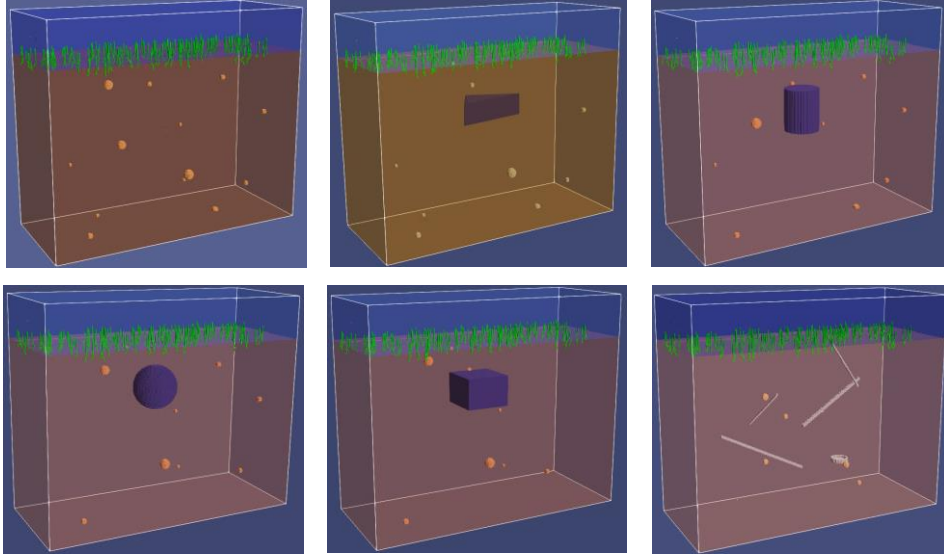
Yukarıda anlatılan üç seviye pozitif veri toplama yöntemi ile sınıflandırıcının eğitimi için yeterli sayıda tanımlayıcı elde edilmektedir. Aşağıda Şekil 19’da üç seviyenin birbirleri ile olan ilişkisi hiyerarşik bir gösterim ile çizilmiştir.



Şekil 19: Pozitif veri toplama hiyerarşisi. Birinci seviyede, nesneye ait fiziksel özellikler (boyut, şekil vb.); ikinci seviyede toprağa ait parametreler (ϵ_r , μ_r) ve üçüncü seviyede de oluşturulan penceredeki konumsal değişiklikler ile pozitif eğitim verisi çoğaltılmaktadır.

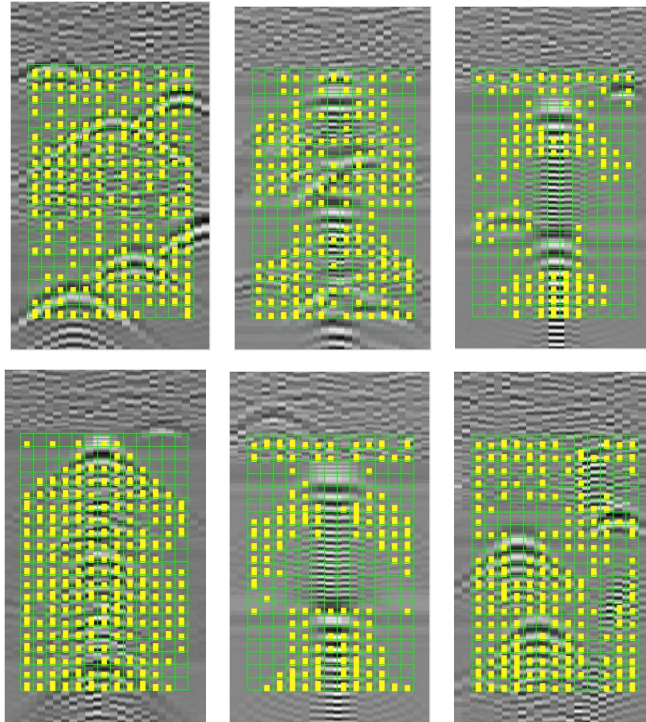
3.6. Negatif Veriler

Sınıflandırıcı eğitimi için pozitif verilerin nasıl elde edildiği yukarıda anlatılmıştı. Negatif verilerin elde edilmesinde yine aynı yöntem kullanılmaktadır. Negatif veri olarak, pozitif veriler ile alakasız nesnelerin gömülü olduğu ortamların kullanılması durumunda, örneğin boyut olarak çok küçük veya çok büyük nesneler, özellik olarak mükemmel elektrik iletkeninden farklı olan nesneler vb. sezim oranları oldukça yüksek olabilmektedir. Pozitif verilere benzer tanımlayıcılar verebilen nesneler seçildiğinde ise sezim oranları azalmaktadır. Negatif verilerin toplanması için kullanılan ortamların bazıları Şekil 20’de verilmektedir



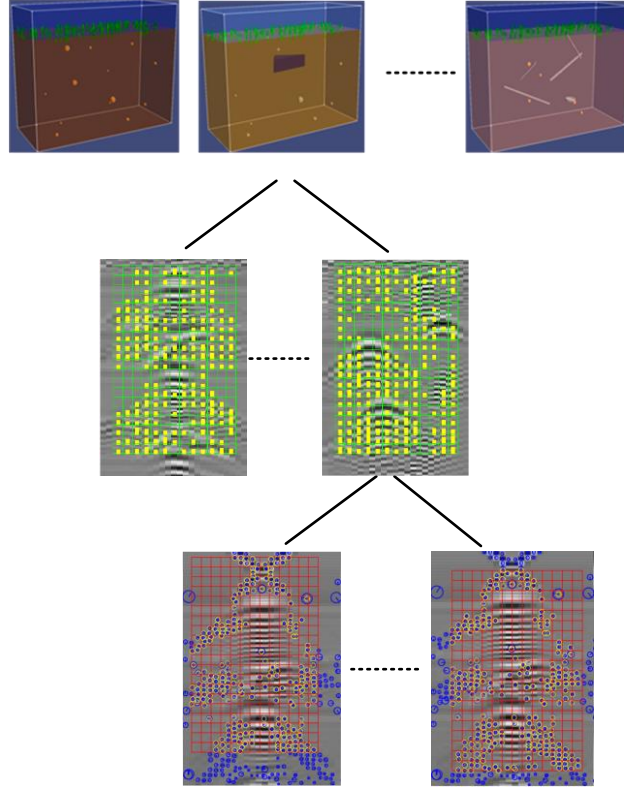
Şekil 20: Sınıflandırıcı eğitiminde kullanılan negatif veriler (Clutter)

Negatif veri çeşitliliği için, pozitif veride olduğu gibi 3 seviye çeşitlilik kullanılmıştır. İkinci ve üçüncü seviye çeşitlilik birebir aynıdır fakat birinci seviye çeşitlilik için her defasında farklı bir nesne ve farklı ortam kullanılmıştır. Şekil 21’de negatif veriler için elde edilen tanımlayıcılar gösterilmiştir. Her bir tanımlayıcı Şekil 20’deki ortama tekabül etmektedir.



Şekil 21: Negatif verilere ait tanımlayıcılar. Şekil 20’de yer alan nesne ve ortamlardan elde edilmiş tanımlayıcılar burada aynı sıra ile yer almaktadır.

Yukarıda anlatılan üç seviye negatif veri toplama yöntemi ile sınıflandırıcının eğitimi için yeterli sayıda tanımlayıcı elde edilmektedir. Aşağıda Şekil 22’de üç seviyenin birbirleri ile olan ilişkisi hiyerarşik bir gösterim ile çizilmiştir.



Şekil 22: Negatif veri toplama hiyerarşisi. Birinci seviyede, ortamda bulunan nesnenin kendisi (küp, çakıl vb.); ikinci seviyede toprağa ait parametreler (ϵ_r , μ_r) ve üçüncü seviyede de oluşturulan penceredeki konumsal değişiklikler ile negatif eğitim verisi çoğaltılmaktadır.

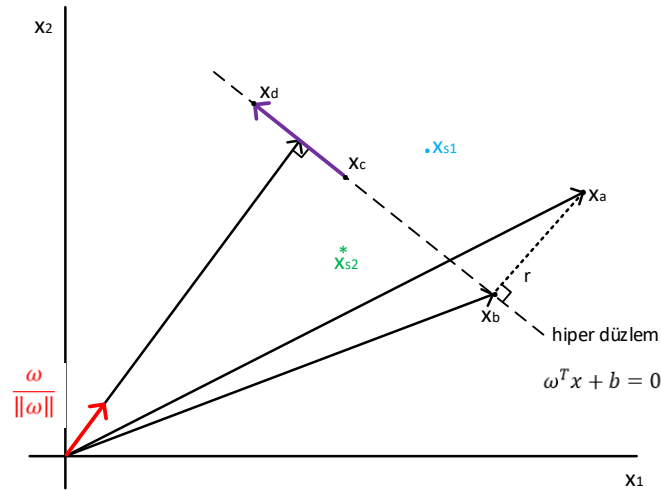
Negatif veri için oluşturulan GPR görüntü sayısı 80 tanedir. Bu sayı ilk iki seviye ile elde edilen sayıdır. Üçüncü seviyeyle birlikte bu değer 720’ye ulaşmaktadır.

4. LS-SVM SINIFLANDIRICI

En küçük kareler destek vektör makinesi (Least Squares Support Vector Machines: LS-SVM) verilerin analiz edilip örüntülerin tanınmasında kullanılan bir güdümlü öğrenme (supervised learning) yöntemidir. LS-SVM, klasik SVM yönteminde dışbükey ikinci derece problem (convex quadratic QP problem) çözme zorluğuna alternatif olarak geliştirilmiş ve sadece bir dizi lineer problemin çözümünden ibaret olan bir tekniktir. Çekirdek (kernel) tabanlı öğrenme yöntemi olan LS-SVM sınıflandırıcılar Suykens ve Vandewalle tarafından geliştirilmiştir [39].

LS-SVM sınıflandırıcı yöntemine geçmeden önce klasik SVM yöntemine bakmak faydalı olacaktır.

4.1. Destek Vektör Makinesi (SVM)



Şekil 23: Vektör uzayında noktaların hiper düzlem ile ilişkisi

Vektör uzayı, bir hiper düzlem ile iki farklı bölgeye ayrılır. Şekil 23'te görüldüğü gibi, lineer $g(\omega, \mathbf{x}) = \omega^T \mathbf{x} + b$ fonksiyonu bir hiper düzlemi ifade eder ve hiper düzlem üzerindeki tüm noktalar aşağıdaki eşitliği sağlar.

$$\omega^T \mathbf{x} + b = 0 \quad (4.1)$$

ω vektörü $g(\omega, \mathbf{x})$ hiper düzlemine diktir. Hiper düzlem üzerinde iki noktayı ele alalım \mathbf{x}_c ve \mathbf{x}_d . Her iki nokta hiper düzlem üzerinde olduğundan ikisi için de geçerli olan denklem $\omega^T \mathbf{x}_c + b = 0$ ve $\omega^T \mathbf{x}_d + b = 0$ dir. Bu iki denklemi birbirinden çıkarırsak, $\omega^T (\mathbf{x}_c - \mathbf{x}_d) = 0$ elde edilir. Bu sonuç ω vektörünün $(\mathbf{x}_c - \mathbf{x}_d)$ vektörüne dik (ortogonal) olduğunu

belirtir. $(\mathbf{x}_c - \mathbf{x}_d)$ vektörü hiper düzlem üzerindeki tüm vektörleri temsil etmektedir, bu durum $\boldsymbol{\omega}$ vektörünün hiper düzleme dik olduğunu kanıtlar.

Hiper düzlem üzerinde olmayan herhangi bir noktanın $g(\boldsymbol{\omega}, \mathbf{x})$ fonksiyonundaki değeri, o noktanın hiper düzleme uzaklığı ile orantılı bir sonuç verir. Örneğin, \mathbf{x}_a noktasına bakalım.

$$\mathbf{x}_a = \mathbf{x}_b + r \frac{\boldsymbol{\omega}}{\|\boldsymbol{\omega}\|}$$

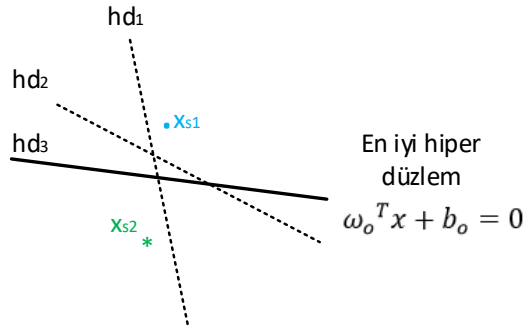
Bu noktanın g fonksiyonundaki değeri,

$$g(\boldsymbol{\omega}, \mathbf{x}_a) = \boldsymbol{\omega}^T \mathbf{x}_a + b \Rightarrow \boldsymbol{\omega}^T \left(\mathbf{x}_b + r \frac{\boldsymbol{\omega}}{\|\boldsymbol{\omega}\|} \right) + b = \boldsymbol{\omega}^T \mathbf{x}_b + b + r \frac{\boldsymbol{\omega}^T \boldsymbol{\omega}}{\|\boldsymbol{\omega}\|}$$

Denklemden $\boldsymbol{\omega}^T \mathbf{x}_b + b = 0$ dır, çünkü \mathbf{x}_b hiper düzlem üzerindedir. Sadeleştirilirse

$$g(\boldsymbol{\omega}, \mathbf{x}_a) = r \frac{\boldsymbol{\omega}^T \boldsymbol{\omega}}{\|\boldsymbol{\omega}\|} = r \|\boldsymbol{\omega}\| \quad (4.2)$$

elde edilir. Bu değer, ilgili noktanın hiper düzleme uzaklığının (r), hiper düzlem ağırlık vektörünün büyüklüğü $\|\boldsymbol{\omega}\|$ ile çarpımıdır.



Şekil 24: İki noktayı ayırabilen hiper düzlemler

Şekil 24'te iki farklı sınıfa ait \mathbf{x}_{s1} ve \mathbf{x}_{s2} noktalarını ayıran hiper düzlemlere bakalım, muhtemel hiper düzlemlerden çok sayıda olabileceği görülür. Bu hiper düzlemlerden en uygun olanı, sınıfların ayırt edilebilirliğini ve sonraki noktaların doğru sınıflandırılma ihtimalini en yüksek yapanıdır. SMV sınıflandırıcılarında bu amaçla izlenen yöntem, her iki noktaya eşit mesafede ve en uzak olan hiper düzlemin seçilmesidir. Şekle bakılacak olursa, hd_1 hiper düzlemi, her iki noktaya da eşit uzaklıktadır fakat uzaklığın en fazla olduğu hiper düzlem değildir. Diğer taraftan, hd_2 'de mesafeler artmıştır fakat noktalar hiper düzleme eşit uzaklıkta değildir, fakat hd_3 'te hem iki nokta hiper düzleme eşit uzaklıktadır hem de bu uzaklık en fazladır.

Matematiksel olarak hd_3 hiper düzlemini elde etmek için eşitlik (4.2) de $g(\boldsymbol{\omega}, \boldsymbol{x})$ fonksiyonunun noktanın hiper düzleme olan uzaklık bilgisini vermesinden yararlanılabilir. Noktaların hiper düzleme olan uzaklıklarına r_2 , yeni hiper düzlem kümesi ağırlık vektörüne $\boldsymbol{\omega}_2$ dersek,

$$\begin{aligned} g(\boldsymbol{\omega}_2, \boldsymbol{x}_{s1}) &= \boldsymbol{\omega}_2^T \boldsymbol{x}_{s1} + b_2 = r_2 \|\boldsymbol{\omega}_2\| = d \\ g(\boldsymbol{\omega}_2, \boldsymbol{x}_{s2}) &= \boldsymbol{\omega}_2^T \boldsymbol{x}_{s2} + b_2 = -r_2 \|\boldsymbol{\omega}_2\| = -d \end{aligned}$$

eşitlikleri elde edilir. Her iki eşitlik de d ile bölünürse,

$$\begin{aligned} \left(\frac{\boldsymbol{\omega}_2}{d}\right)^T \boldsymbol{x}_{s1} + \frac{b_2}{d} &= 1 \\ \left(\frac{\boldsymbol{\omega}_2}{d}\right)^T \boldsymbol{x}_{s1} + \frac{b_2}{d} &= -1 \end{aligned}$$

ve $\boldsymbol{\omega}_2/d$ yerine $\boldsymbol{\omega}_3$, b_2/d yerine de b_3 yazılırsa,

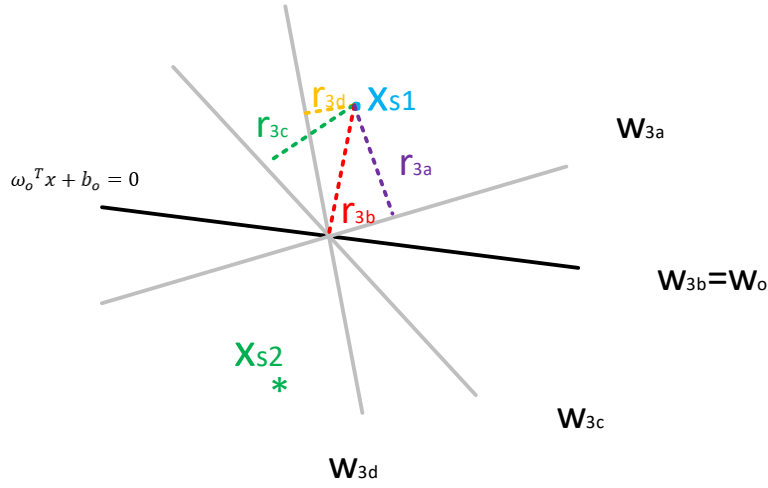
$$\begin{aligned} g(\boldsymbol{\omega}_3, \boldsymbol{x}_{s1}) &= \boldsymbol{\omega}_3^T \boldsymbol{x}_{s1} + b_3 = 1 = r_3 \|\boldsymbol{\omega}_3\| \\ g(\boldsymbol{\omega}_3, \boldsymbol{x}_{s2}) &= \boldsymbol{\omega}_3^T \boldsymbol{x}_{s2} + b_3 = -1 = -r_3 \|\boldsymbol{\omega}_3\| \end{aligned}$$

eşitlikleri elde edilir. Elde edilen bu yeni hiper düzlem kümesi $g(\boldsymbol{\omega}_3, \boldsymbol{x})$, \boldsymbol{x}_{s1} ve \boldsymbol{x}_{s2} noktalarına uzaklıklarının eşit olduğu tüm hiper düzlemleri belirtir. Şekle göre hd_1 ve hd_3 bu hiper düzlem kümesinin iki elemanıdır. Şimdi yapılması gereken bu hiper düzlemlerden uzaklığı azami yapanın bulunmasıdır. Noktaların her birinin hiper düzleme uzaklığı r_3 olduğundan, iki noktanın hiper düzleme uzaklığı $2r_3$ tür.

$$r_3 = \frac{1}{\|\boldsymbol{\omega}_3\|} \Rightarrow 2r_3 = \frac{2}{\|\boldsymbol{\omega}_3\|}$$

$g(\boldsymbol{\omega}_3, \boldsymbol{x})$ hiper düzlem kümesi içinde, $2r_3$ mesafesini en büyük yapan hiper düzlem, ağırlık vektörünün büyüklüğünü en az yapan hiper düzlemdir. Bu koşulu sağlayan ağırlık vektörüne $\boldsymbol{\omega}_o$ dersek, en iyi hiper düzlem denklemi şu şekilde elde edilmiş olur.

$$\boldsymbol{\omega}_o^T \boldsymbol{x} + b_o = 0, \quad \boldsymbol{\omega}_o = \min_{\boldsymbol{\omega}_3} \|\boldsymbol{\omega}_3\|, \quad \text{öyleki} \begin{cases} \boldsymbol{\omega}_3^T \boldsymbol{x}_{s1} + b_3 = 1 \\ \boldsymbol{\omega}_3^T \boldsymbol{x}_{s2} + b_3 = -1 \end{cases}$$



Şekil 25: w_3 hiper düzlem kümesi

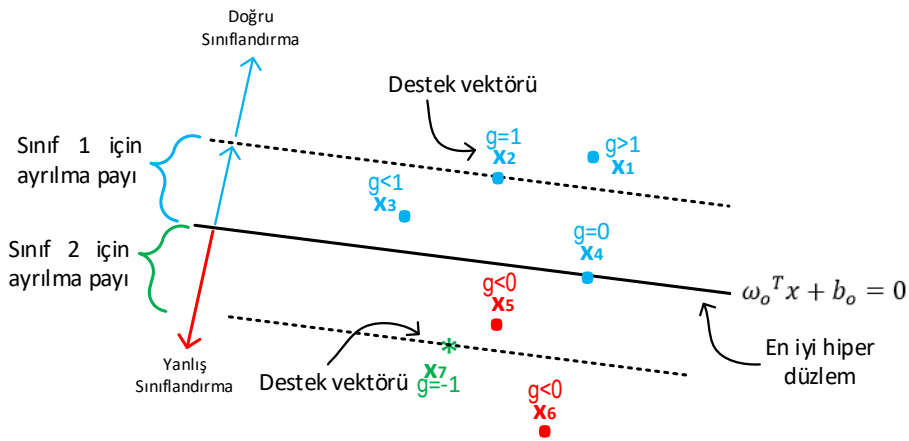
Şekil 25'te w_3 hiper düzlem kümesi görülmektedir. Yani, şekilde gösterilmiş olan dört hiper düzlem de aşağıda verilen eşitlikleri sağlar.

$$\omega_3^T x_{s1} + b_3 = 1 = r_3 \|\omega_3\|$$

$$\omega_3^T x_{s2} + b_3 = -1 = -r_3 \|\omega_3\|$$

Ancak, bunlardan sadece w_{3b} en küçük vektör uzunluğuna (norma) sahiptir ve x_{s1} ve x_{s2} noktalarına en büyük mesafe değerini r_{3b} verir. En iyi sınıflandırıcı için aranan hiper düzlem budur.

Her bir sınıfta birden fazla noktanın bulunması durumunda, bu noktaların en iyi hiper düzlem denklemine $(g(\omega_3, x)|_{\omega_3=\omega_o})$ göre değeri, o noktaların sınıflandırılmaları ile ilgili bilgi verecektir.



Şekil 26: Hiper düzlem formülüne göre noktaların değeri

Şekil 26'ya bakıldığında, Sınıf 1'de yer alan x_1 - x_6 noktalarının aldığı değerler şu şekildedir.

$$g(\omega_3, x)|_{\omega_3=\omega_0} = r_0 \|\omega_0\|, \left\{ \begin{array}{l} r_0 \|\omega_0\| > 1, \text{ örnek: } x_1 \\ r_0 \|\omega_0\| = 1, \text{ örnek: } x_2, x_7 \\ r_0 \|\omega_0\| < 1, \text{ örnek: } x_3 \\ r_0 \|\omega_0\| = 0, \text{ örnek: } x_4 \\ r_0 \|\omega_0\| < 0, \text{ örnek: } x_5 \\ r_0 \|\omega_0\| < 0, \text{ örnek: } x_6 \end{array} \right.$$

Bu noktalardan, g fonksiyonundaki değeri 0'dan büyük olanlar sınıf-1 verisi, g fonksiyonundaki değeri 0'dan küçük olanlar ise sınıf-2 verisi olarak değerlendirilirler. Dolayısıyla x_5 ve x_6 yanlış sınıflandırılmış olmaktadır. Ayrılma payı hiper düzleminin üzerindeki noktalar (x_2 ve x_7), yani g fonksiyonundaki değeri 1 olanlar, destek vektörü (support vector) olarak adlandırılır.

Çok sayıda veriden oluşan iki farklı sınıfını ayıran SVM sınıflandırıcısını bulmak için yukarıda anlatılan matematiksel temellerden yararlanır. İlk olarak, sınıflar için ayrılan ayrılma payları arasında verinin olmadığı duruma bakacak olursak, aşağıdaki problemde yer alan ω_0 'ın bulunması gerekmektedir.

$$\min \|\omega_0\| = \min(\omega_0^T \omega_0), \text{ öyle ki } y_i(\omega_0^T x_i + b_0) \geq 1, i = 1, 2, \dots, N$$

$$y_i = i'inci \text{ veri için etiket(sınıfına göre } +1 \text{ veya } -1)$$

Bu şekilde tanımlanan bir problem, koşullu en iyileştirme problemidir ve Lagrange katsayıları kullanılarak çözülür. Bunun için ilk olarak Lagrangian fonksiyonu yazılır ve daha sonra bu fonksiyon üzerinden türev yardımı ile en iyileştirme gerçekleştirilir [40].

$$\mathcal{L}(\omega, b, \alpha) = \frac{1}{2} \omega^T \omega - \sum_{i=1}^N \alpha_i (y_i(\omega^T x_i + b) - 1)$$

Lagrangian en iyileştirme sonucuna göre elde edilen Lagrange katsayıları sadece destek vektörleri için sıfırdan farklıdır ve pozitifdir. Diğer tüm noktalar için bu katsayılar sıfıra eşittir. Bu özellik, en iyileştirme teorisinin Kuhn-Tucker şartları olarak adlandırılır [41].

Tüm noktaların ayrılma payı üzerinde ve ötesinde olma kuralı genişletilip, noktaların tüm bölgelerde bulunma durumu ele alınırsa, koşulun bu durumu sağlayacak şekilde genişletilmesi gerekir.

$$y_i(\omega_0^T x_i + b_0) \geq 1 - \varepsilon_i, i = 1, 2, \dots, N$$

$$y_i = i'inci \text{ veri için etiket(sınıfına göre } +1 \text{ veya } -1)$$

Yani koşula hata terimi (ε_i) dahil edilir. Bu durumda, ayrılma payı içerisinde ve hatalı konumda olan verilerin olduğu durum için SVM sınıflandırıcı elde edilmiş olur. Eğer $0 < \varepsilon_i \leq 1$ durumu söz konusu ise, veri noktası doğru sınıflandırma bölgesinde ama ayrılma payı içerisinde olmuş olur. Eğer $1 < \varepsilon_i$ durumu söz konusu ise, veri yanlış sınıflandırma bölgesinde olmuş olur. Daha önceki analiz ettiğimiz durum ise $\varepsilon_i \leq 0$ durumudur.

Bu yeni durum için bulunması gereken en iyi hiper düzlem, daha önceki durumda olduğu gibi sadece ağırlık vektörünün büyüklüğünü en küçük yapan değil, aynı zamanda toplam hatayı en az yapan hiper düzlem olmalıdır. Burada bahsedilen iki terim arasındaki önem derecesi (trade off) bir katsayı ile belirlenir.

Yeni durumda problem

$$\frac{1}{2} \boldsymbol{\omega}^T \boldsymbol{\omega} + C \sum_{i=1}^N \varepsilon_i$$

değerini en küçük yapan, aynı zamanda

$$y_i(\boldsymbol{\omega}_o^T \mathbf{x}_i + b_o) \geq 1 - \varepsilon_i, \quad i = 1, 2, \dots, N$$

$$\varepsilon_i \geq 0$$

şartlarını sağlayan hiper düzlemin bulunmasıdır.

Dikkat edilirse, $\varepsilon_i = 0$ durumu, daha önceki durumdur. Yeni durum, daha önceki durumu içermektedir. C katsayısı, sınıflandırıcının geliştirilmesi ve noktaları ezberlemesi (overfit) arasında bir dengeleme katsayısıdır. C değeri düşük seçilirse, sınıflandırıcının geliştirme özelliğinin ağırlığı artarken, C değerinin büyük seçilmesi durumunda hatayı en az yapacak şekilde verileri ezberlemesi sağlanmış olacaktır.

Daha önceki durumda olduğu gibi, burada da Lagrange katsayıları kullanılarak koşullu en iyileştirme problemi çözülür. Yeni Lagrangian şu şekilde elde edilir.

$$\mathcal{L}(\boldsymbol{\omega}, b, \boldsymbol{\alpha}, \boldsymbol{\varepsilon}) = \frac{1}{2} \boldsymbol{\omega}^T \boldsymbol{\omega} + C \sum_{i=1}^N \varepsilon_i - \sum_{i=1}^N \alpha_i (y_i(\boldsymbol{\omega}^T \mathbf{x}_i + b) - 1 + \varepsilon_i) - \sum_{i=1}^N \mu_i \varepsilon_i$$

Bu en iyileştirme problemi çözüldüğünde, elde edilen yeni problem:

N adet eğitim verisi verildiğinde $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$

$$Q(\boldsymbol{\alpha}) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i=1}^N \sum_{k=1}^N \alpha_i \alpha_k y_i y_k \mathbf{x}_i^T \mathbf{x}_k$$

fonksiyonunun deęerini en byk yapan, aynı zamanda

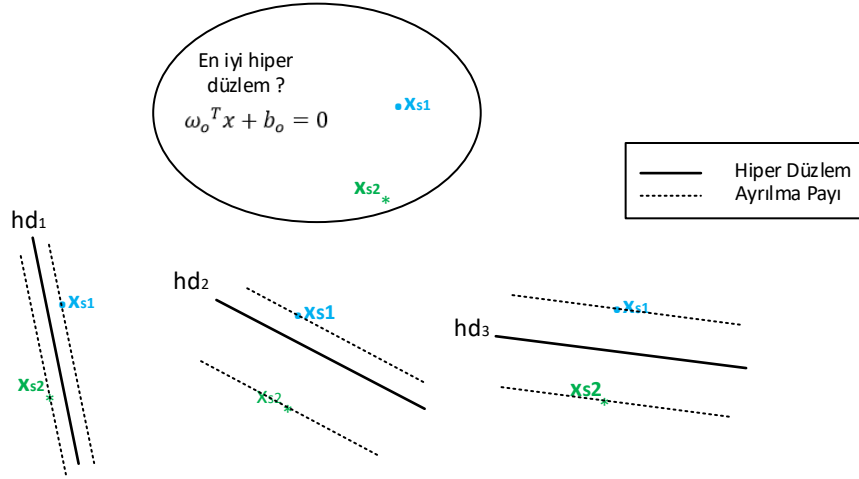
$$\sum_{i=1}^N \alpha_i d_i = 0$$

$$0 \leq \alpha_i \leq C \quad i = 1, \dots, N$$

koşullarını saęlayan Lagrange katsayılarının α bulunması olarak ifade edilir [42].

4.2. En Kçük Kareler Destek Vektr Makinesi (LS-SVM)

LS-SVM’de sınıflandırıcı, standart SVM de olduęu gibi farklı sınıftaki veriler arasındaki ayrılma payını en fazla yapan hiper dzlemin bulunması ile elde edilir. Hedef aynı fakat problemin zm yntemleri farklıdır.



Şekil 27: En iyi hiper dzlem

Daha nceki kısımda, hiper dzlem aęrılık vektrnn byklęnn en kçük yapılması durumunda, ayrılma payı mesafesinin en byk yapıldıęı grlmşt. Bu durum bu yntemde de aynen kullanılmaktadır. Dięer taraftan, gereki bir problem ele alındıęı iin, ayrılma payı ierisinde ve hatta yanlış tarafta yer alan veriler olacaktır. Ayrılma payı en byk yapılırken, aynı zamanda elde edilen hiper dzleme gre oluřan hataların da en kçük olması saęlanmalıdır. Bu iki en iyileřtirme iřlemi, ncelik ayarlama katsayısı da dahil edilerek řu řekilde ifade edilebilir:

$$\min Q(\omega, \varepsilon) = \frac{1}{2} \omega^T \omega + \frac{C}{2} \sum_{i=1}^N \zeta_i \varepsilon_i^2$$

Standart SVM'e göre ilk fark girdi verilerinden kaynaklı hatanın probleme dahil edilmesinden kaynaklanır ki burada hatayı doğrudan almak yerine karesi alınır. Yöntemin ismindeki en az kareler (Least Squares) ifadesi tam olarak buradan kaynaklanır. Denklemdaki C katsayısı, iki terim arasındaki önem faktörünü temsil eder. C büyük seçilirse, hata değerinin küçültülmesi önem kazanır ve veriler ezberlenir, diğer taraftan, C küçük seçilirse, hiper düzlemin ayrılma payı önem kazanır ve sınıflandırıcının genelleştirme özelliği artar. Denklemdaki diğer bir katsayı, her bir hata değeri için atanan zita (ζ)'dır. Bu katsayı iki farklı sınıfa ait verilerin miktarları ile ters orantılı olup, eşit veri olmaması durumunda dengeleme faktörü olarak görev yapar.

Yukarıda ortaya konan Q probleminde en iyileştirme gerçekleştirilirken, ortaya çıkan hiper düzlemin sağlanması gereken hata denklemi şu şekildedir [43]:

$$y_i(\boldsymbol{\omega}^T \mathbf{x}_i + b) = 1 - \varepsilon_i, \quad i = 1, 2, \dots, N \quad (4.3)$$

$y_i = i'inci \text{ veri için etiket(sınıfına göre } + 1 \text{ veya } - 1)$

Sınıflandırılan verilerden sadece destek vektörlerine karşılık gelenlere sıfır hata değerinin verilmesi ve onun dışında kalanlara ayrılma payı hiper düzleminden uzaklaşma miktarı oranında hata değerinin verilmesi, LS-SVM ile birlikte gelen diğer farktır. (4.3) numaralı koşulda hata terimi ele alınacak olursa,

$$\varepsilon_i = 1 - y_i(\boldsymbol{\omega}^T \mathbf{x}_i + b)$$

eşitliğinin her iki tarafı y_i ile çarpılırsa ($y_i = \pm 1$)

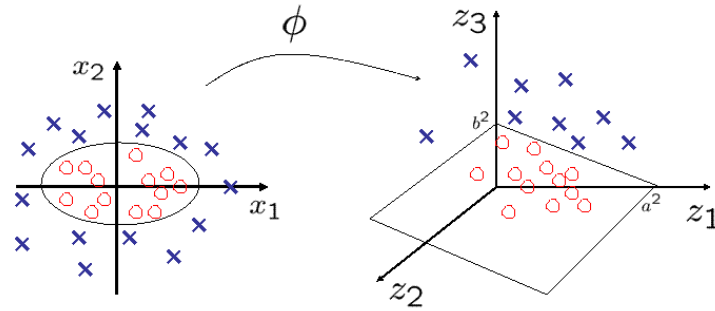
$$y_i \varepsilon_i = y_i - (\boldsymbol{\omega}^T \mathbf{x}_i + b)$$

hata tanımı elde edilir. En iyileştirme denkleminde hatanın karesi alınarak işlem yapıldığı için her iki tarafın karesi alınır,

$$\varepsilon_i^2 = (y_i - (\boldsymbol{\omega}^T \mathbf{x}_i + b))^2$$

eşitliği elde edilir. En iyileştirme denkleminde hatanın karesi kullanıldığı için, hata terimi olarak $\varepsilon_i = y_i - (\boldsymbol{\omega}^T \mathbf{x}_i + b)$ değerinin alınmasında her hangi bir sakınca yoktur. Bu durum, sınıflardan -1 etiketini alan için hata teriminde işaretin değiştirilmesi anlamına gelir. Hatanın değerinde değişim yoktur, sadece işareti terslenmiş olur. Bu durum en iyileştirme formülünde ve ulaşılan hiper düzlemde her hangi bir etki oluşturmamaktadır. Yapılacak işlemlerin kolaylığı açısından bu şekilde tercih edilmektedir.

Ortaya çıkan yeni problem, bir koşullu en iyileştirme problemidir ve çözümü Lagrange katsayıları yöntemidir. Bu Lagrange probleminin çözümüne geçmeden önce, şu ana kadar ele alınan sınıflandırma problemde, girdi verileri üzerinde ön işlem yapılmadan doğrudan girdi uzayında analiz yapıldı. Girdi verilerini lineer olmayan ve çok daha yüksek boyutlu öznelik uzayına (feature space) aktarmak ve onun üzerinden işlemleri yapıp sonuca ulaşmak, örüntü tanıma uygulamalarında çokça kullanılan bir yöntemdir. Örneğin Şekil 28’de lineer olarak sınıflandırılmayan iki boyutlu girdi verileri, daha yüksek boyuta aktarılarak lineer olarak ayrılabilen bir forma dönüştürülmüştür [44].

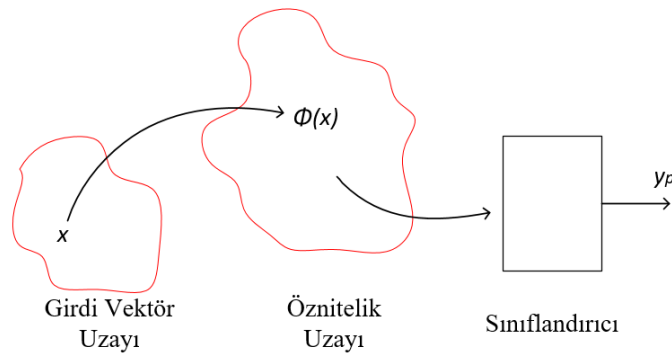


$$\phi : (x_1, x_2) \longrightarrow (x_1^2, \sqrt{2}x_1x_2, x_2^2)$$

$$\left(\frac{x_1}{a}\right)^2 + \left(\frac{x_2}{b}\right)^2 = 1 \longrightarrow \frac{z_1}{a^2} + \frac{z_3}{b^2} = 1$$

Şekil 28: Girdi verisini daha yüksek boyutlu uzaya taşıma

Buradan hareketle, Şekil 29’ dan de görüldüğü gibi, girdi verileri öncelikle öznelik uzayına aktarılıp daha sonra sınıflandırma işlemine tabi tutulur. Yani, girdi verisi olarak x değil, $\phi(x)$ kullanılır.



Şekil 29: Girdi verisini daha yüksek boyutlu uzaya taşıma

4.2.1. LS-SVM Eğitim

Problem, N adet eğitim verisi verildiğinde $\{(x_i, y_i)\}_{i=1}^N$

$$\min Q(\boldsymbol{\omega}, \varepsilon) = \frac{1}{2} \boldsymbol{\omega}^T \boldsymbol{\omega} + \frac{C}{2} \sum_{i=1}^N \zeta_i \varepsilon_i^2$$

Q fonksiyonunun değerini en küçük yapan, aynı zamanda

$$\varepsilon_i = (y_i - (\boldsymbol{\omega}^T \boldsymbol{\phi}(x_i) + b))$$

koşullarını sağlayan hiper düzlemin bulunması, olarak ifade edilebilir.

Yukarıda belirtilen LS-SVM problemine ait Lagrangian şu şekilde yazılır.

$$\mathcal{L}(\boldsymbol{\omega}, b, \alpha, \varepsilon) = \frac{1}{2} \boldsymbol{\omega}^T \boldsymbol{\omega} + \frac{C}{2} \sum_{i=1}^N \zeta_i \varepsilon_i^2 - \sum_{i=1}^N \alpha_i ([\boldsymbol{\omega}^T \boldsymbol{\phi}(x_i) + b] + \varepsilon_i - y_i)$$

En iyilik koşulları aşağıdaki gibi elde edilir.

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\omega}} = 0 \rightarrow \boldsymbol{\omega} = \sum_{i=1}^N \alpha_i \boldsymbol{\phi}(x_i) \quad (4.4)$$

$$\frac{\partial \mathcal{L}}{\partial b} = 0 \rightarrow 0 = \sum_{i=1}^N \alpha_i \quad (4.5)$$

$$\frac{\partial \mathcal{L}}{\partial \varepsilon_i} = 0 \rightarrow \alpha_i = C \zeta_i \varepsilon_i, \quad i = 1, \dots, N \quad (4.6)$$

$$\frac{\partial \mathcal{L}}{\partial \alpha_i} = 0 \rightarrow y_i = \boldsymbol{\omega}^T \boldsymbol{\phi}(x_i) + b + \varepsilon_i \quad (4.7)$$

Yukarıdaki $\boldsymbol{\omega}$ ve ε_i değişkenleri için tanımlı birinci ve üçüncü denklem son denklemde yerine yazılırsa aşağıdaki eşitlik elde edilir.

$$y_i = \left(\sum_{i=1}^N \alpha_i \boldsymbol{\phi}(x_i) \right)^T \boldsymbol{\phi}(x_i) + b + \frac{\alpha_i}{C \zeta_i}, \quad i = 1, \dots, N$$

Eşitlik düzenlenecek olursa, LS-SVM sınıflandırıcı için nihai denklem aşağıdaki gibi elde edilir.

$$y_i = \sum_{j=1}^N \alpha_j \boldsymbol{\phi}(x_j)^T \boldsymbol{\phi}(x_i) + b + \frac{\alpha_i}{C \zeta_i}, \quad i = 1, \dots, N$$

Eşitlikteki toplama işlemi vektör çarpımı olarak yazılırsa, i verisi için denklem aşağıdaki gibi olur.

$$[\phi(x_1)^T \phi(x_i) \quad \phi(x_2)^T \phi(x_i) \quad \dots \quad \phi(x_N)^T \phi(x_i)] \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_N \end{bmatrix} + b + \frac{\alpha_i}{C\zeta_i} = [y_i]$$

i verisi için geçerli yukarıdaki denklem, tüm N adet veri için yazılırsa ve henüz kullanılmamış olan 2 numaralı koşul da dahil edilirse, matris vektör çarpımı cinsinden aşağıdaki denklem elde edilir. Bu denklem, N adet eğitim verisini için tüm koşulları karşılayan LS-SVM eğitim denklemdir.

$$\begin{bmatrix} \phi(x_1)^T \phi(x_1) & \dots & \phi(x_N)^T \phi(x_1) & 1 \\ \vdots & \ddots & \vdots & \vdots \\ \phi(x_1)^T \phi(x_N) & \dots & \phi(x_N)^T \phi(x_N) & 1 \\ 1 & \dots & 1 & 0 \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_N \\ b \end{bmatrix} + \begin{bmatrix} 1/C\zeta_1 & 0 & \dots & 0 \\ 0 & \ddots & \vdots & \vdots \\ \vdots & \dots & 1/C\zeta_N & 0 \\ 0 & \dots & 0 & 0 \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_N \\ b \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \\ 0 \end{bmatrix} \quad (4.8)$$

x_i ve x_j eğitim verilerinin oluşturduğu öznitelik vektörlerinin çarpımı, $\mathbf{K}(x_i, x_j) = \phi^T(x_j)\phi(x_i)$ tanımlaması ile gösterilecek olursa [45], bu elemanların oluşturduğu matris \mathbf{K} , çekirdek matrisi olarak adlandırılır ve SVM sınıflandırıcılarında işlem kolaylığı açısından sıklıkla kullanılır. Bu matris, öznitelik vektörlerini tek tek hesap etme zorunluluğunu ortadan kaldırarak, doğrudan x_i ve x_j eğitim verilerinin daha yüksek boyutlu uzaya taşınmış hallerini kullanma imkanı verir. (4.8) numaralı denklemde en soldaki matris, çekirdek matrisi kullanılarak aşağıdaki gibi yazılabilir.

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}(x_1, x_1) & \mathbf{K}(x_1, x_2) & \dots & \mathbf{K}(x_1, x_N) \\ \mathbf{K}(x_2, x_1) & \mathbf{K}(x_2, x_2) & \dots & \mathbf{K}(x_2, x_N) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{K}(x_N, x_1) & \mathbf{K}(x_N, x_2) & \dots & \mathbf{K}(x_N, x_N) \end{bmatrix}$$

Matrisin elemanlarını hesap etmek için (çekirdek fonksiyonu olarak), çeşitli fonksiyonlar kullanılabilir, bunlardan en sık kullanılanları Tablo 4'te verilmiştir.

Tablo 4: İçler Çarpımı Çekirdek Tipleri

Çekirdek Tipi	Çekirdek Fonksiyonu	Yorum
Polinom Çekirdeği	$\mathbf{K}(x_i, x_j) = (x_i^T x_j + 1)^p$	Fonksiyonlarda kullanılan p , σ ve β değerleri kullanıcı tarafından seçilir. Bu parametrelerin aldığı değere göre, sınıflandırıcının genelleştirme özelliği ya da verileri ezberleme özelliği ağırlık kazanır.
Gauss Çekirdeği	$\mathbf{K}(x_i, x_j) = e^{-\frac{\ x_i - x_j\ }{2\sigma^2}}$	
Hiperbolik Tanjant Çekirdeği	$\mathbf{K}(x_i, x_j) = \tanh(\beta_0 x_i^T x_j + \beta_1)$	

Bu tez kapsamındaki çalışmada Gauss çekirdeğinin kullanılması tercih edilmiştir.

$\alpha = [\alpha_1, \alpha_2, \dots, \alpha_N]^T$, $\mathbf{y} = [y_1, y_2, \dots, y_N]^T$, $\mathbf{Z} = \text{diag}\{\zeta_1^{-1}, \zeta_2^{-1}, \dots, \zeta_N^{-1}\}$ ve $\mathbf{1}_N = [1, \dots, 1]^T$ olacak şekilde (4.8) numaralı formülü sade bir gösterimle aşağıdaki gibi yazabiliriz.

$$\begin{bmatrix} \mathbf{K} + \frac{1}{C}\mathbf{Z} & \mathbf{1}_N \\ \mathbf{1}_N^T & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ b \end{bmatrix} = \begin{bmatrix} \mathbf{y} \\ 0 \end{bmatrix} \quad (4.9)$$

Elde edilen (4.9) denklemi lineer bir denklemdir ve çözümü matris tersini alma ve çarpma işleminden ibarettir. Diğer taraftan standart SVM'de ortaya çıkan problem ikinci derecen bir problemdir (Quadratic Problem) ve çözümü ikinci dereceden en iyileme (quadratic optimization) sonucunda elde edilir. (4.9) lineer denkleminin çözümü ile elde edilen α ve b Lagrange katsayıları ile (4.4) formülü kullanılarak LS-SVM sınıflandırıcı için verilen eğitim verilerinden yola çıkılarak en iyi hiper düzlem elde edilmiş olur.

4.2.2. LS-SVM Sezim

N adet giriş verisi ile yapılan eğitim sonucu elde edilen model parametresi ω aşağıda tekrar yazılan (4.4) denklemi ile elde edilir.

$$\omega = \sum_{i=1}^N \alpha_i \phi(x_i)$$

Burada $\phi(x_i)$, i -inci girdi verisi x_i kullanılarak oluşturulmuş bir öznitelik vektörüdür (feature vector). Öznitelik vektörü, girdi verisini daha yüksek boyutlu hatta (belki de) sonsuz boyutlu bir uzaya taşımaktadır. Elde edilen bu model ile LS-SVM sınıflandırıcıya ait hiper düzlem elde edilmiş olur. Eğitim sonrası sınıfı tespit edilecek herhangi bir test verisi x_t için sınıf bilgisi, o verinin hiper düzlemin hangi tarafında yer aldığı bilgisi ile ortaya çıkar. Burada aşağıdaki (4.10) ile verilen verinin hiper düzleme olan uzaklık bilgisinden yararlanılır.

$$y_p = \omega^T \phi(x_t) + b \quad (4.10)$$

Yukarıda verilen (4.4) denklemi (4.10) denkleminde yerine konulursa, herhangi bir test verisinin t_j sezimi aşağıdaki gibi elde edilir.

$$y_p = \sum_{i=1}^N \alpha_i \phi^T(x_i) \phi(t_j) + b \quad (4.11)$$

Burada tek bir test elemanı için verilen sezim y_p formülü, M adet test verisi için yazılacak olursa aşağıdaki formül elde edilir.

$$\begin{bmatrix} y_{p1} \\ y_{p2} \\ \vdots \\ y_{pM} \end{bmatrix} = \begin{bmatrix} K(t_1, x_1) & K(t_1, x_2) & \dots & K(t_1, x_N) \\ K(t_2, x_1) & K(t_2, x_2) & \dots & K(t_2, x_N) \\ \vdots & \vdots & \ddots & \vdots \\ K(t_M, x_1) & K(t_M, x_2) & \dots & K(t_M, x_N) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_N \end{bmatrix} + b \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \quad (4.12)$$

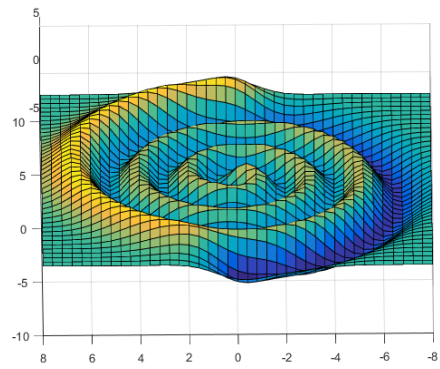
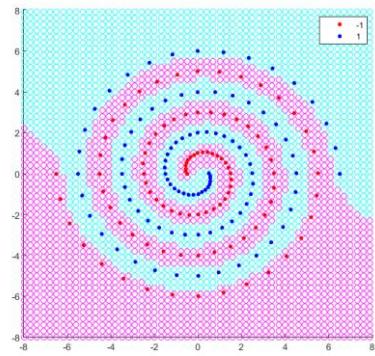
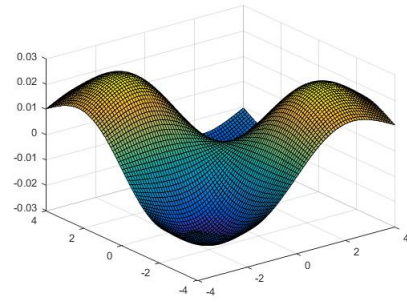
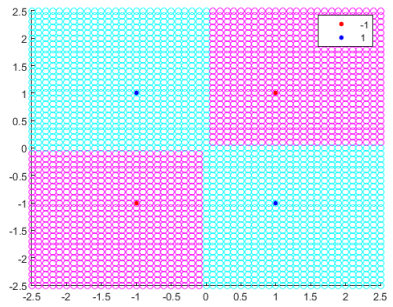
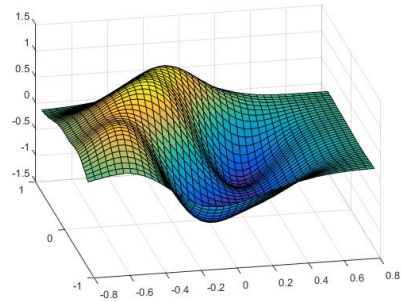
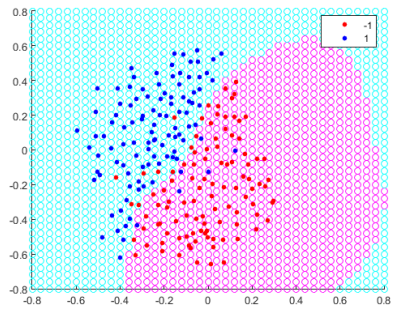
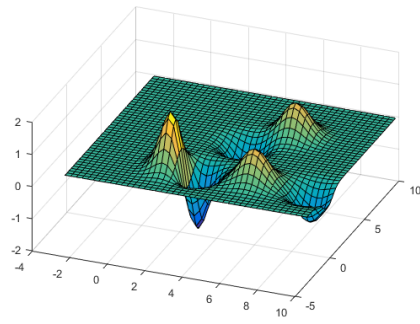
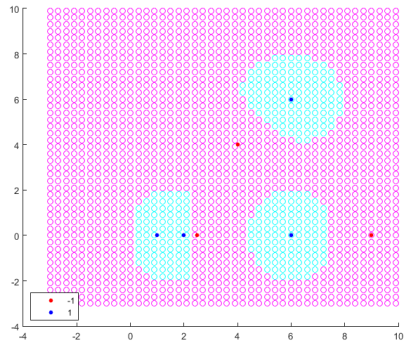
t_j : test noktaları $j = 1, 2, \dots, M$

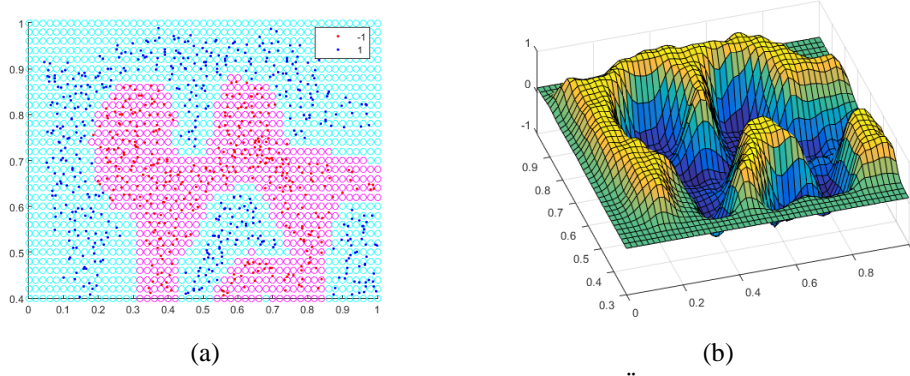
x_i : eğitim noktaları $i = 1, 2, \dots, N$

(4.10) formülünde çarpma işlemini yapıp Çekirdek ile Lagrange katsayılarını birleştirirsek denklem aşağıdakine dönüşür.

$$\begin{bmatrix} y_{p1} \\ y_{p2} \\ \vdots \\ y_{pM} \end{bmatrix} = \begin{bmatrix} \underbrace{K(t_1, x_1)\alpha_1 + K(t_1, x_2)\alpha_2 + \dots + K(t_1, x_N)\alpha_N}_{\text{Merkezi } x_1 \text{ Olan Gauss}} & \underbrace{K(t_2, x_1)\alpha_1 + K(t_2, x_2)\alpha_2 + \dots + K(t_2, x_N)\alpha_N}_{\text{Merkezi } x_2 \text{ Olan Gauss}} & \dots & \underbrace{K(t_M, x_1)\alpha_1 + K(t_M, x_2)\alpha_2 + \dots + K(t_M, x_N)\alpha_N}_{\text{Merkezi } x_N \text{ Olan Gauss}} \end{bmatrix} + b \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \quad (4.13)$$

Veri uzayında sınıflandırıcının görünümü, merkezi veri noktalarında olan, büyüklüğü ve polaritesi $\alpha_1, \alpha_2, \dots, \alpha_N$ Lagrange katsayılarının değeri olan Gaussların toplamından oluşan bir eğridir. Eğriye göre test verilerinden değeri pozitif olanlar bir sınıfa, değeri negatif olanlar da diğer sınıfa aittir. Böylece, eğrinin değerinin sıfır olduğu düzlem hiper düzlem olmuş olur. Aşağıda Şekil 30'da LS-SVM sınıflandırıcının farklı dağılımlardaki verileri sınıflandırma başarısı gözükmektedir. Örneklerde (a) sütunundaki grafiklerde kırmızı ve mavi olan noktalar eğitim verilerini gösterir. Camgöbeği ve açık mor ile taralı alanlar, LS-SVM sınıflandırıcıya göre o bölgedeki verilerin hangi sınıfa atandığını belirtir. (b) sütunundaki grafikler ise (4.13) denklemi ile ifade edilen sınıflandırıcının, aslında eğitim verilerine tekabül eden noktalarda büyüklükleri Lagrange katsayıları α olan Gauss eğrilerinin toplamından ibaret olduğunu gösterir. Bu grafiğe göre en iyi hiper düzlem, z ekseninin 0 a karşılık gelen noktalarının birleşiminden oluşur.



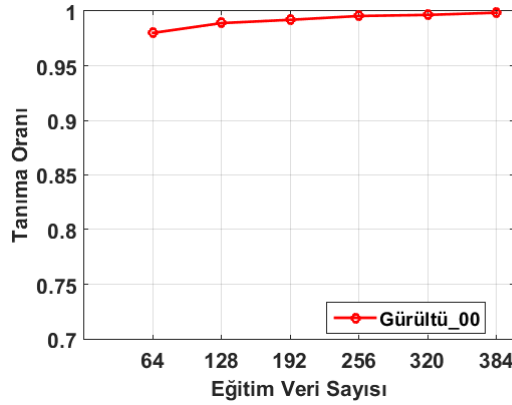


Şekil 30: LS-SVM Sınıflandırma Örnekleri

4.2.3. GPR Verilerinde LS-SVM

LS-SVM sınıflandırıcının iki boyutlu uzaydaki farklı sınıfa ait veriler üzerindeki sezim başarısının yüksekliği Şekil 30'da gösterilmiştir. Gerçek GPR verileri üzerindeki sınıflandırıcının başarısını görmek için oluşturulan sınıflar LS-SVM algoritması ile eğitilmektedir. Algoritma her çalıştırıldığında mevcut pozitif ve negatif verinin rastgele 100'er tanesi test için ayrılmakta; geri kalanları da eğitim için kullanılmaktadır. Böylece her eğitim ve test döngüsünde farklı veri kümeleri ile işlem yapılarak çıkan sonuçlar karşılaştırılmaktadır.

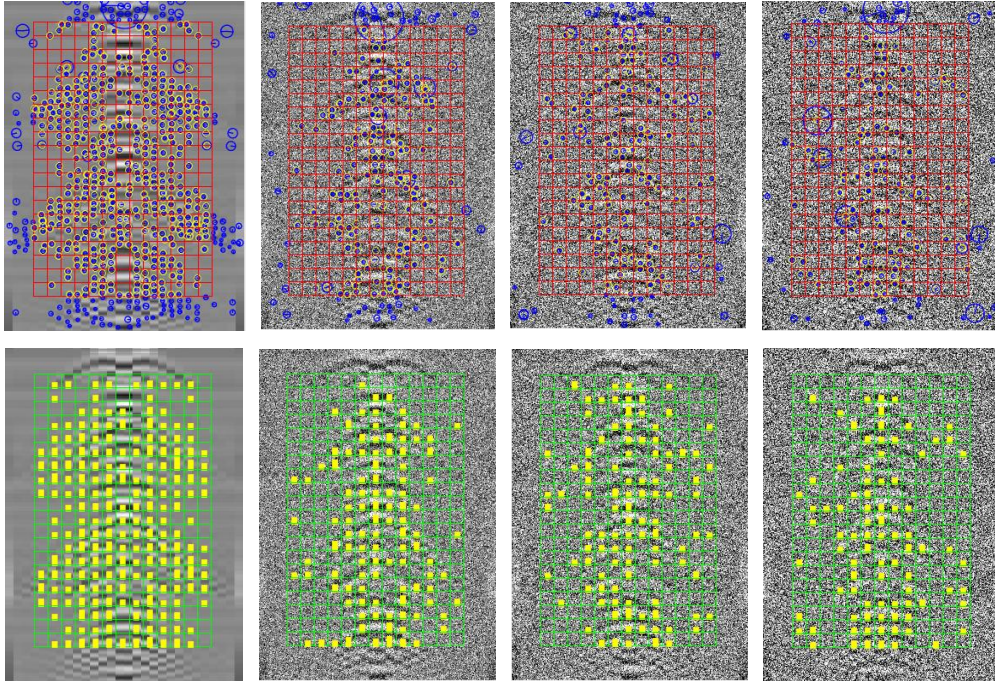
Toplanan pozitif ve negatif veriler üzerinde herhangi bir işlem yapmadan doğrudan eğitim ve test işleminin uygulanması durumunda ortaya çıkan başarı oranları oldukça yüksek olmaktadır. Örnek olarak, Şekil 31'de, "ıslak kum içinde tüp" sınıfı için sezim oranları görülmektedir. Bu sınıfta bulunan toplam 486 pozitif verinin rastgele 100 tanesi test için ayrılmış ve eğitim verisi döngüsel olarak 64'den başlamak suretiyle kademeli olarak 384'e kadar artırılmıştır.



Şekil 31: "Islak kum içinde tüp" sınıfı için sezim sonuçları.

Grafikten de görüleceği üzere başarı oranı düşük veri sayılarında bile oldukça yüksektir. Bu durum, GPR görüntülerinin simülasyon ile elde edilmiş olmasından kaynaklanmaktadır. Simülasyon ortamında, alıcı-verici sistem ideal ve kusursuzdur; aynı zamanda, nesnenin gömülü olduğu toprak homojen ve tekdüze yapıdadır. Bu sebeple, elde edilen simülasyon görüntülerinde aradığımız özellikleri veren şekiller; düzgün, belirgin ve simetrik. Bu sebeple, simülasyonla elde edilen görüntüleri, gerçek alıcı-verici ve gerçek ortam ile elde edilen GPR görüntülerine yaklaştırmak için, oluşturulan görüntülere gürültü eklenebileceği öngörülmüştür.

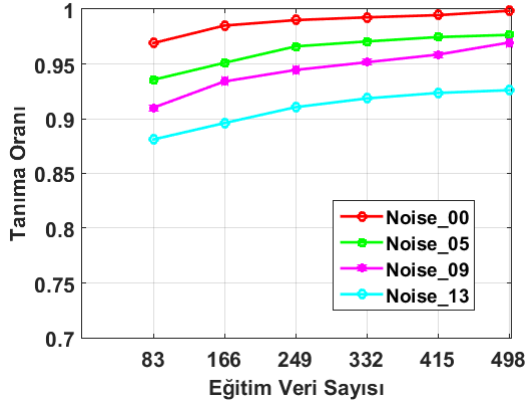
Bu kapsamda GPR görüntüleri üzerine eklediğimiz gürültü, gauss gürültüsüdür. Gürültünün ortalama değeri olarak 0 seçilmiştir. Varyans değerine bağlı olarak da görüntü kalitesinin ve sezim oranlarının değiştiği gözlemlenmiştir. Şekil 32’de eklenen gürültünün varyansının artırılması ile birlikte ortaya çıkan SIFT özellikleri ve tekabül eden tanımlayıcılar verilmiştir. Şekilde soldan sağa eklenen gauss gürültüsünün varyans değerleri: 0, 0.05, 0.09 ve 0.13 olarak alınmıştır. GPR Benzetimi için “kuru kumtaşı içinde tüp” sınıfı kullanılmıştır. Toprak parametreleri $\epsilon_r:5$ ve $\mu_r:2$ dir.



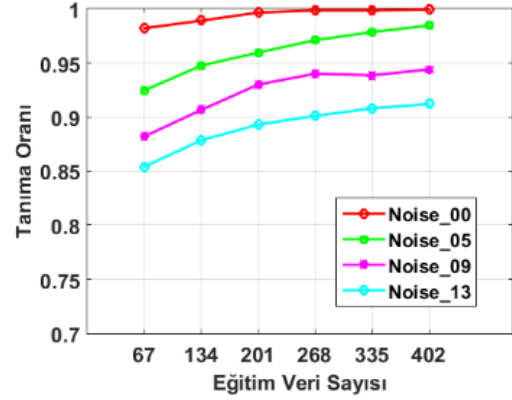
Şekil 32 GPR görüntülerine gürültü eklenmesi

Farklı varyans değerlerine sahip gauss gürültüsünün eklenmesi ile birlikte tanımlayıcıdaki bilginin azaldığı şekilden görülmektedir. Bu sayede, alıcı-verici sisteminde ve nesnenin gömülü olduğu toprağın homojenlik ve tekdüzeliğinde oluşabilecek farklı seviyelerde ideal

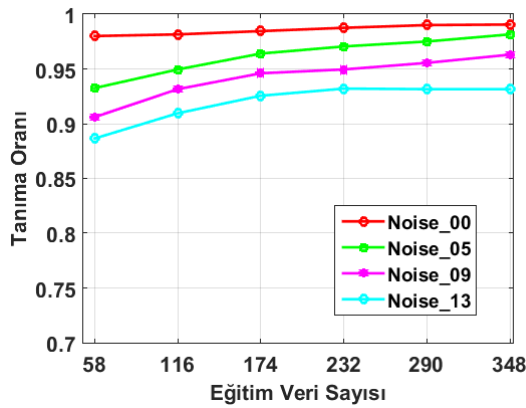
olmama durumu, simülasyon görüntülerine eklenen farklı seviyelerde gürültü ile ele alınmıştır. Belirtilen varyans değerleri ile elde edilen veri sayısına karşı tanıma oranlarını gösteren grafik Şekil 33'te verilmiştir.



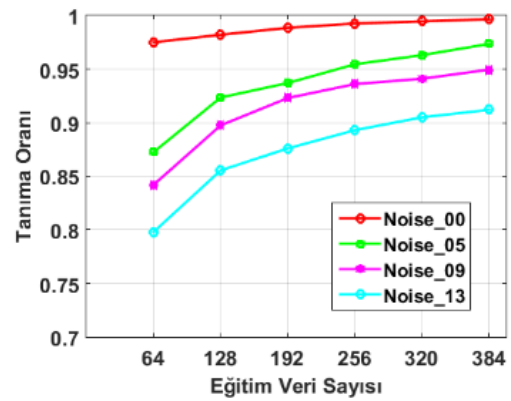
(a) Kuru kumtaşı içinde tüp



(b) Kuru kum içinde tüp



(c) Islak kumtaşı içinde tüp

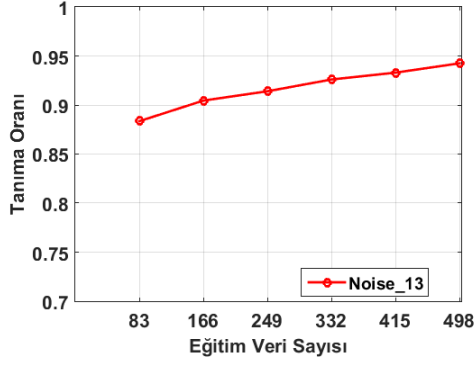


(d) Islak kum içinde tüp

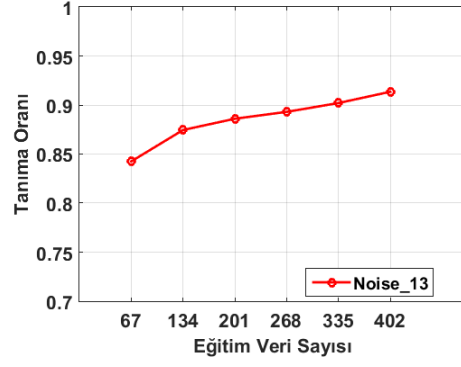
Şekil 33: Gürültü seviyelerine göre tanıma başarımları grafikleri

Bu durum, görüntüleme kullanılan radar sistemi ve nesnenin içinde bulunduğu ortamın gürültü seviyesindeki değişimleri yansıtmaktadır. Bahsedilen gürültünün seviyesindeki değişim, sezim oranlarına şekilde görüldüğü gibi yansıtmaktadır. Şekil 33'ten de görüleceği üzere, eklenen gürültünün varyansının artması ile birlikte başarımlar azalmaktadır.

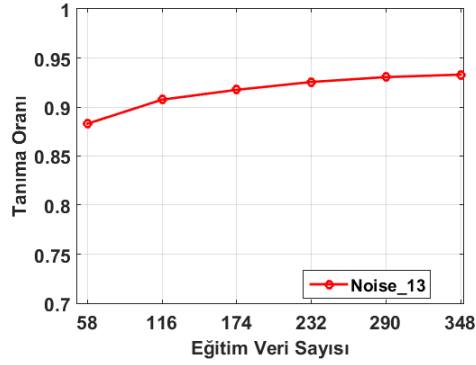
Bu sonuçlardan yola çıkarak bu tez kapsamında elde edilen GPR görüntülerine ortalama değeri 0, varyansı da 0.13 olan gauss gürültüsü eklenmiştir. Böylece elde edilen kaynak ve hedef sınıflarına ait LS-SVM sınıflandırıcı başarımları grafikleri Şekil 34'teki gibi elde edilmiştir.



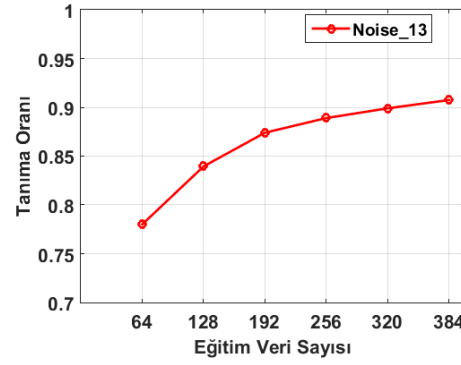
(a) Kuru kumtaşı içinde tüp



(b) Kuru kum içinde tüp



(c) Islak kumtaşı içinde tüp



(d) Islak kum içinde tüp

Şekil 34: Gürültü eklenmiş verilerde sınıflara ait tanıma başarımları

Sınıflara ait Şekil 34'teki grafikler elde edilirken, hepsinde aynı yöntem uygulanmıştır. Örneğin, kuru kumtaşı içinde tüp sınıfı için, toplam 598 görüntünün rastgele 100 tanesi test için seçilmiş, geri kalan 498 pozitif veri de rastgele 6 eşit parçaya ayrılmıştır. Aynı şekilde, negatif veri kümesi içinden rastgele 100 tane görüntü test için seçilmiş, rastgele seçilen 498 görüntü de eğitim için 6 eşit parçaya ayrılmıştır. Birinci döngüde, 83 pozitif ve 83 negatif görüntü ile sınıflandırıcı eğitilmiş ve seçilen 200 görüntü üzerinde (100 pozitif, 100 negatif) teste tabi tutulup başarımları kaydedilmiştir. İkinci döngüde, ilk döngüde seçilen 83 veriye ilave olarak diğer bir 83'lük pozitif ve negatif veri kümesi dahil edilip eğitim ve test tekrar edilmiş ve başarımları kaydedilmiştir. Bu döngüsel işlem tüm veri kümesini kapsayacak şekilde tekrarlanmış ve elde edilen sonuçlar grafik üzerinde gösterilmiştir. Tüm döngülerde, test verisi olarak, ilk başta seçilen 200 lük veri kümesi kullanılmıştır.

5. BİLGİ AKTARIMI

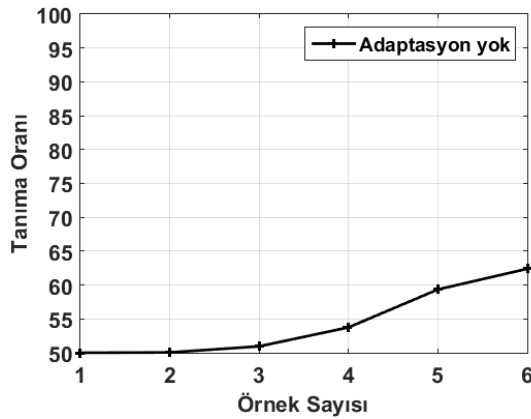
Bölüm 4.2.3'te, yeterli sayıda verinin bulunması durumunda sınıflandırıcının başarısının oldukça yüksek olduğu görülmüştü. Fakat her zaman sınıflandırıcıyı eğitmek için yeterli miktarda eğitim verisi bulunmayabilir. Böyle durumlarda, klasik sınıflandırıcılar ile yapılabilecek iki seçenek bulunmaktadır.

- Sınıflandırıcıyı az sayıda veri ile eğitip kullanmak
- Benzer bir sınıfa ait sınıflandırıcıyı yeni veri kümesi üzerinde kullanmak.

Bu iki durumda elde edilen sezim başarı oranını GPR verileri üzerinde inceleyelim. Hedef verisi olarak “Islak kum içerisinde tüp” sınıfını seçmiştik. Bu sınıftan sırası ile birden altıya kadar veri rastgele seçilerek eğitime tabi tutulursa ve geri kalanlardan 100 tanesini doğrulama için kullanılırsa ortaya çıkan başarıım grafiği Şekil 35'teki gibi elde edilmiştir.

Bu grafiği elde etmek için,

- İlk eğitimde rastgele bir veri eğitimde kullanılmış ve rastgele diğer 100 veri üzerinde test yapılmıştır.
- İkinci eğitimde kalan verilerden rastgele bir tanesi daha seçilerek eğitime dahil edilmiş ve aynı 100 veri üzerinde test yapılmıştır.
- Üçüncü ve devam eden eğitimlerde sırasıyla rastgele birer veri daha eğitime dahil edilmiş ve aynı seçilen 100 veri üzerinde test yapılmıştır.



Şekil 35: Az sayıda eğitim verisi ile tanıma oranları.

Şekilden de görüleceği üzere, az sayıda veri ile yapılan eğitim ile elde edilen başarıım oranı oldukça yetersizdir. Diğer taraftan, hedef sınıfa benzer olan diğer üç kaynak sınıfına ait sınıflandırıcıların hedef sınıfını sezim oranları Tablo 5'te verilmiştir.

- Hedef sınıfına ait rastgele 100 veri üzerinde kaynak sınıfına ait model ile sezim yapılmıştır.

Tablo 5: Kaynak modellerinin, ıslak kum içerisinde tüp hedef verisini tanıma oranları

Kaynak Sınıf	Hedef Sınıf Verilerini Tahmin Oranları
Kuru Kum içinde piknik tüpü	60
Kuru Kumtaşı içinde piknik tüpü	40
Islak Kumtaşı içinde piknik tüpü	57

Tablodan görüleceği üzere, kaynak modellerin hedef modeli tahmin etme başarısı yeterli değildir. Üç farklı kaynak sınıfı için eğitilmiş sınıflandırıcının, hedef sınıf verilerinin seziminde kullanılması durumunda, elde edilen en iyi başarı oranı %60 civarında iken, Şekil 35'ten görüleceği üzere, az sayıda hedef eğitim verisi ile eğitilmiş sınıflandırıcının kendi sınıfına ait verileri sezim oranı %62 civarındadır. Peki, bu noktada, insana has bir özellik olan, daha önce öğrenilmiş bilgilerden yararlanarak yeni bilgilerin daha hızlı ve kolay öğrenilmesi olgusundan yararlanılabilir mi? Tezin bundan sonraki kısmı, öğrenme aktarımı tekniğini kullanarak az sayıda veri ile eğitim gerçekleştirirken aynı zamanda daha önceden öğrenilmiş kaynak verilerinden yararlanma üzerine olacak.

5.1. Uyarlamalı LS-SVM

Genel olarak SVM sınıflandırıcılar için geçerli olan öğrenme problemi, yukarıdaki bölümde de ele alındığı gibi şu şekilde ifade edilmektedir.

$$\min_{\omega, b} \left\{ \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^N \ell(\omega^T \phi(x_i) + b, y_i) \right\} \quad (5.1)$$

Yukarıdaki en iyileme probleminde ilk terim model parametresidir. Bölüm 4.1'de anlatıldığı gibi, sınıflandırıcının genelleştirmeyi en iyi şekilde yapabilmesini sağlamak ve hiper düzlem ayrılma payını en büyük yapmak için normunun en aza indirilmesi gerekmektedir. İkinci terim ise, elde edilen model sonucu yapılan sezimin, verinin etiketine göre sapmasının bedeli olan kayıp fonksiyonudur. C katsayısı, tüm girdi verileri kullanılarak elde edilen kayıp toplamı ile hiper düzlem ayrılma payı (model parametresi) arasındaki önceliği ayarlamak için kullanılan parametredir. C değeri kullanıcı tarafından seçilir ve sınıflandırıcının genelleştirme özelliği ile ezberleme özelliği arasındaki ayarlamayı yapar. C değeri büyük

seçilirse kayıp en aza indirilecek şekilde modelin girdi verilerini ezberlemesi (girdi verilerine uyması, overfitting) sağlanır. Diğer taraftan, C değeri küçük seçilirse modelin genelleştirme özelliği kuvvetlendirilmiş olur.

Eğitimde kullanılan verilerin sayısının (N) yeterli olması durumunda, (5.1) denklemi yardımı ile herhangi bir sınıfa ait model elde edilebilir. 3.2.3 Numaralı bölümde GPR verileri üzerinde bu yöntemle elde edilen sınıflandırıcıların başarısının yüksek olduğu görülmüştü. Eğitimde kullanılacak veri sayısının az olması durumunda ise (5.1) formülünün doğrudan kullanılıp sıfırdan öğrenmenin gerçekleştirilmesi yerine, önceden elde edilen benzer modellerin uyarlanmasından (Adaptive Learning) yararlanılabilir. Bunu sağlamak için, yeni modelin öğrenilmiş modele mümkün olduğunca benzemesinin ve aynı zamanda da yeni veri kümesini en az hata ile tanıyan olmasının sağlanması gerekmektedir.

Daha önceden elde edilmiş, kaynak sınıfına ait modele $\hat{\omega}$ diyelim. Yeni modelin ω , kaynak modele mümkün olduğunca benzemesini sağlamak için problemi aşağıdaki gibi şekillendirebiliriz.

$$\min_{\omega, b} \left\{ \frac{1}{2} \|\omega - \hat{\omega}\|^2 + C \sum_{i=1}^N \ell(\omega^T \phi(x_i) + b, y_i) \right\} \quad (5.2)$$

İleri sürülen bu en iyileştirme probleminde, ilk terim ile hedef modelin ω kaynak modele $\hat{\omega}$ benzer olması; ikinci terim ile de yeni modelin kendi sınıfına ait verileri en az hata ile tanıyan olması sağlanmaktadır. Bu formül, hedef sınıf modelinin oluşturulması için tek bir kaynak sınıfı modelinin kullanılması durumudur. Yani, bir adet kaynaktan bilgi aktarımının gerçekleştirildiği durumdur. Eğer hedef sınıf modeline benzer, birden fazla kaynak sınıf modeli varsa, bunların tümünden bilgi aktarımı gerçekleştirilebilir. Sonraki bölümde, çok modelden bilgi aktarımı ile hedef modelin elde edilmesi ele alınacaktır.

5.2. Çok Modelden Bilgi Aktarımı

Tek modelden bilgi aktarımı için elde ettiğimiz denklemi çok modelden bilgi aktarıma uyarlama için yapacağımız işlem, (5.2) numaralı problemde kaynak modele tekabül eden ifadeyi çoklu model için yazmaktır. Bunun için, birden çok kaynağa ait modelin lineer birleşimini $\sum_{j=1}^J \beta_j \hat{\omega}_j$ kullanacağız. β_j katsayısı, kaynak sınıfının hedef sınıfına benzerliğini gösteren bir parametredir. (5.2) numaralı denklemde, eğitim verilerini tahmin

etme kaybı olarak, LS-SVM'in bir özelliği olarak ağırlıklandırılmış hatanın karesi kullanılmaktadır $\ell(\boldsymbol{\omega}^T \boldsymbol{\phi}(\mathbf{x}_i) + b, y_i) = \zeta_i (\boldsymbol{\omega}^T \boldsymbol{\phi}(\mathbf{x}_i) + b - y_i)^2$ [46]. Buradaki ağırlıklandırma terimi ζ_i farklı sayıda negatif veri ve pozitif veri kullanılması durumunda dengelemeyi yapmak içindir ve veri sayıları ile ters orantılı bir terimdir. Bu durumda en iyileştirme problemi aşağıdaki gibi yazılır.

$$\min_{\boldsymbol{\omega}, b} \left\{ \frac{1}{2} \left\| \boldsymbol{\omega} - \sum_{j=1}^J \beta_j \hat{\boldsymbol{\omega}}_j \right\|^2 + \frac{C}{2} \sum_{i=1}^N \zeta_i \varepsilon_i^2 \right\} \quad (5.3)$$

$$\text{Öyle ki, } y_i = \boldsymbol{\omega}^T \boldsymbol{\phi}(\mathbf{x}_i) + b + \varepsilon_i, \quad \forall i = 1, \dots, N$$

Burada denkleme ilave edilen ε_i yapay değişken terimi (slack variable), girdi verisi \mathbf{x}_i 'nin yanlış sınıflandırılmasından kaynaklı hata değerini belirtmektedir. Ortaya konan bu kısıtlı en iyileme (constrained optimization) probleminin çözümü için kullanılacak Lagrangian Formülü \mathcal{L} aşağıdaki gibidir.

$$\mathcal{L} = \frac{1}{2} \left\| \boldsymbol{\omega} - \sum_{j=1}^J \beta_j \hat{\boldsymbol{\omega}}_j \right\|^2 + \frac{C}{2} \sum_{i=1}^N \zeta_i \varepsilon_i^2 - \sum_{i=1}^N a_i (\boldsymbol{\omega}^T \boldsymbol{\phi}(\mathbf{x}_i) + b + \varepsilon_i - y_i) \quad (5.4)$$

Buradaki $a_i \in \mathbb{R}^N$ terimleri Lagrange katsayılarıdır. Vektör içler çarpımının türevi $\partial(\mathbf{x}^T \mathbf{x})/\partial \mathbf{x} = 2\mathbf{x}$ eşitliğinden yararlanarak, Lagrangian formülündeki tüm parametreler için en iyilik koşulları aşağıdaki gibi yazılabilir.

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\omega}} = 0 \quad \rightarrow \quad \boldsymbol{\omega} = \sum_{j=1}^J \beta_j \hat{\boldsymbol{\omega}}_j + \sum_{i=1}^N a_i \boldsymbol{\phi}(\mathbf{x}_i) \quad (5.5)$$

$$\frac{\partial \mathcal{L}}{\partial \varepsilon_i} = 0 \quad \rightarrow \quad a_i = C \zeta_i \varepsilon_i \quad (5.6)$$

$$\frac{\partial \mathcal{L}}{\partial a_i} = 0 \quad \rightarrow \quad 0 = \boldsymbol{\omega}^T \boldsymbol{\phi}(\mathbf{x}_i) + b + \varepsilon_i - y_i \quad (5.7)$$

$$\frac{\partial \mathcal{L}}{\partial b} = 0 \quad \rightarrow \quad 0 = \sum_{i=1}^N a_i \quad (5.8)$$

(5.5) ve (5.6) numaralı eşitliklerde elde edilen terimleri (5.7) numaralı denkleme yerine yazılırsa:

$$\left(\sum_{j=1}^J \beta_j \hat{\boldsymbol{\omega}}_j + \sum_{k=1}^N a_k \boldsymbol{\phi}(\mathbf{x}_k) \right)^T \boldsymbol{\phi}(\mathbf{x}_i) + b + \frac{a_i}{C \zeta_i} - y_i = 0$$

eşitliği elde edilir. (5.5) numaralı denklem kullanılırken, indis karışıklığını önlemek için i yerine k yazılmıştır. Parantez açılırsa,

$$\sum_{j=1}^J \beta_j \hat{\omega}_j^T \phi(\mathbf{x}_i) + \sum_{k=1}^N a_k \phi(\mathbf{x}_k)^T \phi(\mathbf{x}_i) + b + \frac{a_i}{C\zeta_i} - y_i = 0$$

elde edilir. Eşitliğin sol tarafındaki ilk ve son terim sağ tarafa aktarılırsa,

$$\sum_{k=1}^N a_k \phi(\mathbf{x}_k)^T \phi(\mathbf{x}_i) + b + \frac{a_i}{C\zeta_i} = y_i - \sum_{j=1}^J \beta_j \hat{\omega}_j^T \phi(\mathbf{x}_i) \quad (5.9)$$

denklemini elde edilir. Bu denklemin girdileri: $\phi(\mathbf{x}_k)$, $\phi(\mathbf{x}_i)$ ile belirtilen eğitim verilerinden (\mathbf{x}) elde edilen öznitelik vektörleri; bu eğitim verilerine tekabül eden y_i veri etiketleri; kaynak sınıflarına ait modeller $\hat{\omega}_j$ ve dengeleme katsayıları C ve ζ_i parametreleridir. Denklem ile elde edilen çıktı veriler ise, Lagrange katsayıları a_i ($i = 1, \dots, N$), bias b ve kaynak sınıflarından bilgi aktarmada benzerlik oranını temsil eden β_j parametreleridir.

(5.9) numaralı denklem incelendiğinde, sonucun tek bir giriş verisi \mathbf{x}_i için elde edildiği görülür. N adet eğitim verisinden kaynaklı tüm denklemleri içerecek şekilde ifade bir matris denklemini olarak yazılırsa

$$\begin{bmatrix} \phi(\mathbf{x}_1)^T \phi(\mathbf{x}_1) & \dots & \phi(\mathbf{x}_N)^T \phi(\mathbf{x}_1) & 1 \\ \vdots & \ddots & \vdots & \vdots \\ \phi(\mathbf{x}_1)^T \phi(\mathbf{x}_N) & \dots & \phi(\mathbf{x}_N)^T \phi(\mathbf{x}_N) & 1 \\ 1 & \dots & 1 & 0 \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_N \\ b \end{bmatrix} + \begin{bmatrix} 1/C\zeta_1 & 0 & \dots & 0 \\ 0 & \ddots & \vdots & \vdots \\ \vdots & \dots & 1/C\zeta_1 & 0 \\ 0 & \dots & 0 & 0 \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_N \\ b \end{bmatrix} = \begin{bmatrix} y_1 - \sum_{j=1}^J \beta_j \hat{\omega}_j^T \phi(\mathbf{x}_1) \\ \vdots \\ y_N - \sum_{j=1}^J \beta_j \hat{\omega}_j^T \phi(\mathbf{x}_N) \\ 0 \end{bmatrix}$$

denklemini elde edilir. Bu denklemin son satırına tekabül eden eşitlik Lagrangian en iyileştirme sonucunda elde edilen (5.8) numaralı formülü vermektedir.

$\mathbf{K}_{ji} = \mathbf{K}(\mathbf{x}_j, \mathbf{x}_i) = \phi(\mathbf{x}_j)^T \phi(\mathbf{x}_i)$ çekirdek matrisi, $\mathbf{a} = [a_1, \dots, a_N]^T$ Lagrange katsayıları vektörü, $\mathbf{y} = [y_1, \dots, y_N]^T$ etiket vektörü ve $\mathbf{1}_N = [1, \dots, 1]^T$ olacak şekilde yukarıdaki eşitliğin sade gösterimi aşağıdaki gibidir.

$$\begin{bmatrix} \mathbf{K} + \frac{1}{C} \mathbf{Z} & \mathbf{1}_N \\ \mathbf{1}_N^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ b \end{bmatrix} = \begin{bmatrix} \mathbf{y} - \sum_{j=1}^J \beta_j \hat{\mathbf{y}}_j \\ 0 \end{bmatrix} \quad (5.10)$$

Denkleminde $\mathbf{Z} = \text{diag}\{\zeta_1^{-1}, \dots, \zeta_N^{-1}\}$ farklı etiketli örnekleri dengelemek için kullanılan diyagonal matrisdir. N^+ Pozitif veri sayısını N^- de negatif veri sayısını gösterecek şekilde diyagonal elemanların değerleri şu şekilde yazılır.

$$\zeta_i = \begin{cases} \frac{N}{2N^+} & \text{Eğer } y_i = +1 \text{ ise} \\ \frac{N}{2N^-} & \text{Eğer } y_i = -1 \text{ ise} \end{cases} \quad (5.11)$$

Ayrıca $\hat{y}_j = [\hat{\omega}_j^T \phi(\mathbf{x}_1), \dots, \hat{\omega}_j^T \phi(\mathbf{x}_N)]^T$ önceki modellerin yeni veriyi tahmin etmesi ile alakalı bir terim olarak kullanılmaktadır.

(5.10) numaralı denklemde model parametrelerini elde etmek, en fazla matris tersini alma işlemi kadar zordur ve diğer algoritma ile çözümlenebilen SVM yöntemleri ile karşılaştırıldığında oldukça kolaydır.

$$\mathbf{M} = \begin{bmatrix} \mathbf{K} + \frac{1}{C} \mathbf{Z} & \mathbf{1}_N \\ \mathbf{1}_N^T & 0 \end{bmatrix} \quad (5.12)$$

dönüştürmesi yapılır ve $\mathbf{M}^{-1} = \mathbf{P}$ olacak şekilde (5.10) numaralı denklem çözümlürse

$$\begin{bmatrix} \mathbf{a} \\ b \end{bmatrix} = \mathbf{P} \begin{bmatrix} \mathbf{y} - \sum_{j=1}^J \beta_j \hat{\mathbf{y}}_j \\ 0 \end{bmatrix} \quad (5.13)$$

denklemini elde edilir. Burada matris tersini almak için Moore-Penrose pseudoinverse tersini alma yöntemi kullanılır [47]. Bu denklemin çözümü için gerekli tüm parametreler β_j hariç bilinmektedir. Eğer β_j değerleri 0 kabul edilirse, yani öğrenme aktarımının olmadığı durum, normal LS-SVM formülü elde edilir. En uygun beta değerlerini elde etme ise bilgi aktarımında kaynakların olumlu katkı yapıp yapmayacağı ve farklı kaynakların hangisinden ne zaman ve ne kadar bilgi aktarılacağı problemdir.

5.3. Ne Zaman ve Ne Kadar Bilgi Aktarılmalı

En uygun beta ağırlık değerlerini bulmak, bilginin aktarıldığı kaynakları uygunluğuna göre sıralayıp hangisinden aktarım yapılacağına ve ne kadar aktarım yapılacağına karar vermeye tekabül eder. Bu problemin çözümü için bu çalışmada yararlanılan yöntem, birini hariç tutarak hata değerini (leave-one-out error) en aza indirme yöntemidir. Yani eğitimde kullanılan verilerden bir tanesi eğitim sürecine dahil edilmez ve diğer verilerle eğitilen sınıflandırıcının hariç tutulan veriyi tahmin etme hata değerine bakılır. Beta değerleri, hata değerini en az yapacak şekilde güncellenir ve başka bir eğitim verisi için işlem tekrarlanır.

Bu şekilde, Beta değerleri güncellenerek döngü devam ettirilir, öyle ki hata değerindeki değişim hiç değişmeyene ya da ihmal edilebilir bir seviyeye gelene kadar [46].

(5.12) numaralı terimi açık şekilde yazıp birinci satırı ve birinci sütunu izole edersek aşağıdaki gibi elde edilir.

$$M = \begin{bmatrix} (\boldsymbol{\phi}(x_1)^T \boldsymbol{\phi}(x_1) + 1/C\zeta_1) & [\boldsymbol{\phi}(x_2)^T \boldsymbol{\phi}(x_1) & \dots & \boldsymbol{\phi}(x_N)^T \boldsymbol{\phi}(x_1) & 1] \\ \left[\begin{array}{c} \boldsymbol{\phi}(x_1)^T \boldsymbol{\phi}(x_2) \\ \vdots \\ \boldsymbol{\phi}(x_1)^T \boldsymbol{\phi}(x_N) \\ 1 \end{array} \right] & \left[\begin{array}{c} \boldsymbol{\phi}(x_2)^T \boldsymbol{\phi}(x_2) + 1/C\zeta_2 \\ \vdots \\ \boldsymbol{\phi}(x_2)^T \boldsymbol{\phi}(x_N) \\ 1 \end{array} \right] & \dots & \left[\begin{array}{c} \boldsymbol{\phi}(x_N)^T \boldsymbol{\phi}(x_2) \\ \vdots \\ \boldsymbol{\phi}(x_N)^T \boldsymbol{\phi}(x_N) + 1/C\zeta_N \\ 1 \end{array} \right] & 1 \\ & & & & 0 \end{bmatrix}$$

Yukarıda terimleri açık olarak gösterilen \mathbf{M} matrisi aşağıdaki blok matris gösterimi ile belirtilebilir.

$$\mathbf{M} = \begin{bmatrix} \mathbf{K} + \frac{1}{C}\mathbf{Z} & \mathbf{1}_N \\ \mathbf{1}_N^T & 0 \end{bmatrix} = \begin{bmatrix} m_{11} & \mathbf{m}_1^T \\ \mathbf{m}_1 & \mathbf{M}_{(-1)} \end{bmatrix} \quad (5.14)$$

Başlangıç olarak, birinci eğitim verisinin eğitim kümesinden çıkarıldığını farz edelim. Bu durumda, model parametrelerini $a_{(-1)}$ ve $b_{(-1)}$ olarak gösterirsek ve $\mathbf{y}_{(-1)} = [y_2, \dots, y_N, 0]^T$, $\hat{\mathbf{y}}_{j(-1)} = [\hat{\boldsymbol{\omega}}_j^T \boldsymbol{\phi}(x_2), \dots, \hat{\boldsymbol{\omega}}_j^T \boldsymbol{\phi}(x_N), 0]^T$ olarak yazarsak, (5.13) numaralı denklemin bu durum için ifade edilme şekli aşağıdaki gibi olur.

$$\begin{bmatrix} \mathbf{a}_{(-1)} \\ b_{(-1)} \end{bmatrix} = \mathbf{P}_{(-1)} \begin{bmatrix} \mathbf{y}_{(-1)} - \sum_{j=1}^J \beta_j \hat{\mathbf{y}}_{j(-1)} \\ 0 \end{bmatrix} \quad (5.15)$$

$\mathbf{M}_{(-1)}^{-1} = \mathbf{P}_{(-1)}$ olarak yazılmıştır. Bu şekilde eğitilmiş modelin birinci veriyi tahmin değeri $f(\mathbf{x}) = y_p = \boldsymbol{\omega}^T \boldsymbol{\phi}(\mathbf{x}) + b$ genel sezim formülünden ve (5.5) numaralı denklemden yararlanarak şu şekilde bulunur.

$$\tilde{y}_1 = \left(\sum_{j=1}^J \beta_j \hat{\boldsymbol{\omega}}_j + \sum_{i=2}^N a_{(-1)} \boldsymbol{\phi}(x_i) \right)^T \boldsymbol{\phi}(x_1) + b_{(-1)}$$

Birinci verinin eğitimden hariç tutulması sonrası oluşan modelin, birinci eğitim verisini tahmin etmesi yukarıda \tilde{y}_1 olarak gösterilmiştir. Bu formüldeki devirme (transpose) işlemi gerçekleştirilir ve denklem düzenlenirse aşağıdaki formül elde edilir.

$$\tilde{y}_1 = \left(\sum_{i=2}^N a_{(-1)i} \boldsymbol{\phi}(x_i)^T \boldsymbol{\phi}(x_1) + b_{(-1)} \right) + \sum_{j=1}^J \beta_j \hat{\boldsymbol{\omega}}_j^T \boldsymbol{\phi}(x_1)$$

Yukarıdaki formülde parantez içerisindeki terim, daha önce elde edilen değişkenler cinsinden yazılırsa aşağıdaki eşitlik elde edilir.

$$\tilde{y}_1 = \mathbf{m}_1^T \begin{bmatrix} a_{(-1)} \\ b_{(-1)} \end{bmatrix} + \sum_{j=1}^J \beta_j \hat{\omega}_j^T \boldsymbol{\phi}(x_1)$$

(5.15) numaralı formül ile elde edilen model parametreleri yukarıdaki formülde yerine konulursa aşağıdaki denklem elde edilir.

$$\tilde{y}_1 = \mathbf{m}_1^T \mathbf{P}_{(-1)} \left[\mathbf{y}_{(-1)} - \sum_{j=1}^J \beta_j \hat{\mathbf{y}}_{j(-1)} \right] + \sum_{j=1}^J \beta_j \hat{\omega}_j^T \boldsymbol{\phi}(x_1)$$

Yukarıdaki parantez içerisindeki terim, (5.10) numaralı denklemde eşitliğin sağ tarafının son N satırıdır. Bu terim, formüle girdidir ve değeri birini hariç tutma (loo) hesabından bağımsızdır. (5.14) numaralı gösterimden yararlanarak, bu terimin eşdeğeri (5.10) numaralı denklemin son N satırı alınarak elde edilir.

$$[\mathbf{m}_1 \quad \mathbf{M}_{(-1)}] \begin{bmatrix} \mathbf{a} \\ b \end{bmatrix} = \left[\mathbf{y}_{(-1)} - \sum_{j=1}^J \beta_j \hat{\mathbf{y}}_{j(-1)} \right]$$

Bu eşitlik, \tilde{y}_1 için yazılan yukarıdaki formülde yerine yazılırsa, aşağıdaki blok matris çarpımını içeren denkleme dönüşür.

$$\tilde{y}_1 = \mathbf{m}_1^T \mathbf{P}_{(-1)} [\mathbf{m}_1 \quad \mathbf{M}_{(-1)}] \begin{bmatrix} \mathbf{a} \\ b \end{bmatrix} + \sum_{j=1}^J \beta_j \hat{\omega}_j^T \boldsymbol{\phi}(x_1)$$

Çarpma işlemi ilk iki terim için uygulanırsa, denklem şuna dönüşür:

$$\tilde{y}_1 = [\mathbf{m}_1^T \mathbf{P}_{(-1)} \mathbf{m}_1 \quad \mathbf{m}_1^T \mathbf{P}_{(-1)} \mathbf{M}_{(-1)}] \begin{bmatrix} \mathbf{a} \\ b \end{bmatrix} + \sum_{j=1}^J \beta_j \hat{\omega}_j^T \boldsymbol{\phi}(x_1)$$

$\mathbf{P}_{(-1)} = \mathbf{M}_{(-1)}^{-1}$ olduğu için, yukarıdaki eşitlikte bu kullanılabilir.

$$\tilde{y}_1 = [\mathbf{m}_1^T \mathbf{P}_{(-1)} \mathbf{m}_1 \quad \mathbf{m}_1^T] \begin{bmatrix} \mathbf{a} \\ b \end{bmatrix} + \sum_{j=1}^J \beta_j \hat{\omega}_j^T \boldsymbol{\phi}(x_1)$$

Burada yine çarpma işlemi ilk iki terim için uygulanırsa aşağıdaki denklem elde edilir.

$$\tilde{y}_1 = \mathbf{m}_1^T \mathbf{P}_{(-1)} \mathbf{m}_1 a_1 + \mathbf{m}_1^T [a_2, \dots, a_N, b] + \sum_{j=1}^J \beta_j \hat{\omega}_j^T \boldsymbol{\phi}(x_1)$$

(5.10) numaralı denklemin son N satırı kullanılarak yukarıdaki denklem elde edilmişti. Şimdi yine (5.14) kullanılarak aynı denklemin ilk terimi alınır, aşağıdaki elde edilir.

$$[m_{11} \quad \mathbf{m}_1^T] \begin{bmatrix} \mathbf{a} \\ b \end{bmatrix} = y_1 - \sum_{j=1}^J \beta_j \hat{\omega}_j^T \boldsymbol{\phi}(x_1)$$

Benzer şekilde, bu denklem aşağıdaki şekilde yazılabilir.

$$m_{11} a_1 + \mathbf{m}_1^T [a_2, \dots, a_N, b] = y_1 - \sum_{j=1}^J \beta_j \hat{\omega}_j^T \boldsymbol{\phi}(x_1)$$

Burada elde edilen formül, \tilde{y}_1 içi bulunan son formülde yerine yazılırsa

$$\tilde{y}_1 = \mathbf{m}_1^T \mathbf{P}_{(-1)} \mathbf{m}_1 a_1 + \mathbf{m}_1^T [a_2, \dots, a_N, b] + y_1 - m_{11} a_1 - \mathbf{m}_1^T [a_2, \dots, a_N, b]$$

eşitliği elde edilir. Sadeleştirme yapılırsa, denklem şuna dönüşür.

$$\tilde{y}_1 = y_1 - a_1 (m_{11} - \mathbf{m}_1^T \mathbf{P}_{(-1)} \mathbf{m}_1)$$

\mathbf{M} matrisinde tersleme işlemi yapılır ve blok matris tersleme teoremi kullanılırsa,

$$\mathbf{P} = \mathbf{M}^{-1} = \begin{bmatrix} m_{11} & \mathbf{m}_1^T \\ \mathbf{m}_1 & \mathbf{M}_{(-1)} \end{bmatrix}^{-1}$$

$$\mathbf{P} = \begin{bmatrix} \mu^{-1} & -\mu^{-1} \mathbf{m}_1 \mathbf{P}_{(-1)} \\ \mathbf{P}_{(-1)} + \mu^{-1} \mathbf{P}_{(-1)} \mathbf{m}_1^T \mathbf{m}_1 \mathbf{P}_{(-1)} & -\mu^{-1} \mathbf{P}_{(-1)} \mathbf{m}_1^T \end{bmatrix}$$

μ parametresi için aşağıdaki formül elde edilir. Bu terim \mathbf{P} matrisinin ilk elemanın P_{11} tersidir.

$$\mu = m_{11} - \mathbf{m}_1^T \mathbf{P}_{(-1)} \mathbf{m}_1$$

Bulunan bu terim yerine yazılırsa, birinci terimi hariç tutma sonucu oluşan modelin birinci terimi sezim ifadesi şu şekilde elde edilmiş olur:

$$\tilde{y}_1 = y_1 - a_1 \mu = y_1 - \frac{a_1}{P_{11}}$$

Birinci eğitim verisi için elde edilen bu sonuç, tüm veriler için genelleştirilebilir. Yukarıda elde edilen sonucun tüm verileri içerecek şekilde yazımı da aşağıdaki şekildedir:

$$\hat{y}_i = y_i - a_i \mu = y_i - \frac{a_i}{P_{ii}} \quad (5.16)$$

Burada elde edilen sonucun, kaynak verileri ve beta değerleri cinsinden yazılması gerekmektedir. Daha sonra, bulunan bu sezimden yola çıkarak en az hatanın olduğu beta değerleri için bir algoritma geliştirilmelidir. (5.13) numaralı denklemden yola çıkılırsa,

$$\begin{bmatrix} \mathbf{a} \\ b \end{bmatrix} = \mathbf{P} \left[\mathbf{y} - \sum_{j=1}^J \beta_j \hat{\mathbf{y}}_j \right]$$

elde edilir. Bu eşitlik, sağ taraftaki çarpma işlemi yapılarak açılırsa, şu şekilde yazılabilir.

$$\begin{bmatrix} \mathbf{a} \\ b \end{bmatrix} = \mathbf{P} \begin{bmatrix} \mathbf{y} \\ 0 \end{bmatrix} - \sum_{j=1}^J \beta_j \left(\mathbf{P} \begin{bmatrix} \hat{\mathbf{y}}_j \\ 0 \end{bmatrix} \right)$$

$$\hat{\mathbf{y}}_j = \begin{bmatrix} \hat{\omega}_j^T \boldsymbol{\phi}(\mathbf{x}_1) \\ \vdots \\ \hat{\omega}_j^T \boldsymbol{\phi}(\mathbf{x}_N) \end{bmatrix}$$

Aşağıda yazılan dönüştürme, daha sonra bulunacak olan formülleri sadeleştirecektir.

$$\begin{bmatrix} \mathbf{a}' \\ b' \end{bmatrix} = \mathbf{P} \begin{bmatrix} \mathbf{y} \\ 0 \end{bmatrix}, \quad \begin{bmatrix} \mathbf{a}_j'' \\ b_j'' \end{bmatrix} = \mathbf{P} \begin{bmatrix} \hat{\mathbf{y}}_j \\ 0 \end{bmatrix}$$

İkinci eşitlik bir kaynak verisi için yazılmıştır, bu tüm J adet kaynak verilerini içerecek şekilde genişletildiğinde, aşağıdaki matris çarpımı şeklinde yazılabilir.

$$\begin{bmatrix} \mathbf{a}_1'' & \dots & \mathbf{a}_J'' \\ b_1'' & \dots & b_J'' \end{bmatrix} = \mathbf{P} \begin{bmatrix} \hat{\omega}_1^T \boldsymbol{\phi}(\mathbf{x}_1) & \dots & \hat{\omega}_J^T \boldsymbol{\phi}(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ \hat{\omega}_1^T \boldsymbol{\phi}(\mathbf{x}_N) & \dots & \hat{\omega}_J^T \boldsymbol{\phi}(\mathbf{x}_N) \\ 0 & \dots & 0 \end{bmatrix}_{(N+1) \times J}$$

\mathbf{a}_j'' vektörlerinin oluşturduğu, j 'nin satırlarını, i 'nin de sütunlarını belirttiği \mathbf{A}'' matrisini aşağıdaki gibi tanımlarsak, elde edilecek formül daha sade bir gösterimde olacaktır. \mathbf{A}'' matrisinde satırlar (J adet) her bir kaynak verisine, sütunlar da (N adet) her bir eğitim verisine tekabül etmektedir.

$A'' = \begin{bmatrix} \mathbf{a}_1''^T \\ \vdots \\ \mathbf{a}_j''^T \end{bmatrix} \in \mathbb{R}^{J \times N}$ matrisinin sütunları \mathbf{A}_i'' ile satırları da \mathbf{A}_j'' ile gösterilmektedir.

Bu gösterimle birlikte Lagrange katsayılarının Beta değerleri cinsinden yazımı aşağıdaki gibi elde edilir.

$$\begin{bmatrix} \mathbf{a} \\ b \end{bmatrix} = \begin{bmatrix} \mathbf{a}' \\ b' \end{bmatrix} - \sum_{j=1}^J \beta_j \begin{bmatrix} \mathbf{a}_j'' \\ b_j'' \end{bmatrix}$$

Bu formülde \mathbf{a}' değerleri, hedef eğitim verileri ve tekabül eden etiketlerden; \mathbf{a}_j'' değerleri de kaynak modellerin hedef eğitim verileri sezimlerinden elde edilir. \mathbf{a} Lagrange katsayıları için hedef eğitim verileri ve kaynak modeller eğitim sürecinde girdi olduğu için, tek bilinmeyen parametre beta değerleridir. Yukarıdaki formül için elde ettiğimiz sade gösterim aşağıdaki eşitliktir.

$$a_i = a'_i - \boldsymbol{\beta}^T \mathbf{A}_i''$$

(5.16) numaralı denklem ile elde edilen birini hariç tutma sezim değeri, yukarıda elde edilen a parametresinin yeni gösterimi cinsinden yazılırsa ulaşılmak istenen formül elde edilir.

$$\tilde{y}_i = y_i - \frac{a_i}{P_{ii}} \quad \Rightarrow \quad \tilde{y}_i = y_i - \frac{a'_i}{P_{ii}} + \frac{\boldsymbol{\beta}^T \mathbf{A}_i''}{P_{ii}} \quad (5.17)$$

Beta değerlerinin elde edilmesiyle birlikte hedef sınıfı için model elde edilmiş olacaktır. SVM/LS-SVM sınıflandırıcı için Genel sezim formülü $f(\mathbf{x}) = \boldsymbol{\omega}^T \boldsymbol{\phi}(\mathbf{x}) + b$ kullanılarak hedef model için çıktı değerleri aşağıdaki gibi elde edilecektir. Burada $\boldsymbol{\omega}$ için Lagrangian en iyileme ile elde edilen (5.5) numaralı formül kullanılacaktır.

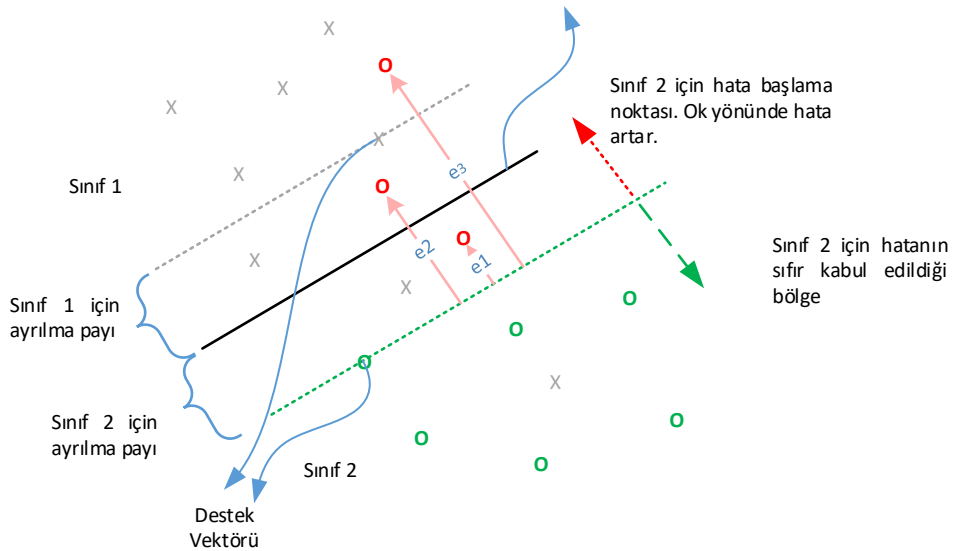
$$y_i = \left(\sum_{j=1}^J \beta_j \hat{\boldsymbol{\omega}}_j + \sum_{k=1}^N a_k \boldsymbol{\phi}(\mathbf{x}_k) \right)^T \boldsymbol{\phi}(\mathbf{x}_i) + b$$

En iyi Beta değerlerini elde etmek için eğitim verisinin işareti ile birini hariç tutma sezim değerinin işaretinin aynı olması kuralına bakılabilir. Bu durumda ortaya çıkacak olan yöntem dışbükey olmayan bir formül olacağı için bunun yerine önerilen yöntem aşağıdakidir.

$$\ell(\tilde{y}_i, y_i) = \zeta_i |1 - \tilde{y}_i y_i|_+ = \zeta_i \left| y_i \frac{a'_i - \boldsymbol{\beta}^T \mathbf{A}_i''}{P_{ii}} \right|_+ \quad (5.18)$$

$|x|_+ = \max\{0, x\}$ tanımı geçerlidir. Yani, sezimin mutlak değeri 1 den fazla ve işareti gerçek veri etiketinin işareti ile aynı ise bu sezim kayıp olarak değerlendirilmez. Diğer taraftan sezimin değeri 1 den küçük ve işareti gerçek veri etiketinin işareti ile aynı ise bu sezim kayıp olarak değerlendirilir ve kayıp değeri olarak 1 den uzaklaşma miktarı alınır. Sezim değeri ile etiket farklı işaretli ise bu her durumda kayıp olarak değerlendirilir. Her eğitim verisindeki kayıp değeri için, (5.11) denkleminde verilen ilgili verinin önemini belirten ζ_i parametresi hata değerinde çarpan olarak yer almaktadır. Bu şekilde bir kayıp formülü çıkartılmasının mantığı şudur: Sezim değerinin ayrılma payının (margin of separation) dışında yer alması desteklenirken (sıfır hata ile ödüllendirme), ayrılma payının içinde ama doğru tarafta yer alması cezalandırılmaktadır.

Birini hariç tutma yöntemi ile oluşturulan modelin hariç tutulan veriyi seziminden elde edilen değer Şekil 36'da gözüktüğü gibi model uzayındaki konumuna bakılarak cezalandırma işlemi yapılır. Örneğin sınıf-2 için elde edilen sezim, ilgili sınıfa ait ayrılma payı hiper düzleminin sağında kalıyorsa (yeşil, kesikli çizgili ok ile gösterilen bölge), yapılan sezim desteklenmekte ve sıfır hata olarak kabul edilmektedir. Diğer taraftan, elde edilen sezim ayrılma payı hiper düzleminin solunda kalıyorsa (kırmızı, noktalı ok ile gösterilen bölge), yapılan sezim cezalandırılır ve hata sezimin ayrılma payı hiper düzlemine olan mesafesi kadar alınır. Şekilde üç farklı sezim için elde edilen hata değerleri e_1 , e_2 ve e_3 olarak gösterilmiştir.



Şekil 36: Beta değerlerini en iyileştirirken kullanılan hata değerleri

Şekil 36’da gösterilen her bir giriş verisine karşılık gelen hata değeri (e_1, \dots, e_i) formül (5.18) ile ifade edilen kayıp değerleridir. En hızlı alçalma (steepest descent) yöntemine göre bu hata formülü kullanılarak her veri için gradyan bulunur ve bu gradyanların bileşeni doğrultusunda ters yönde ilerleme yöntemi ile hata değerini en az yapan beta değerleri hesaplanır. Tüm veriler için kullanılacak gradyan formülü aşağıda verilmiştir.

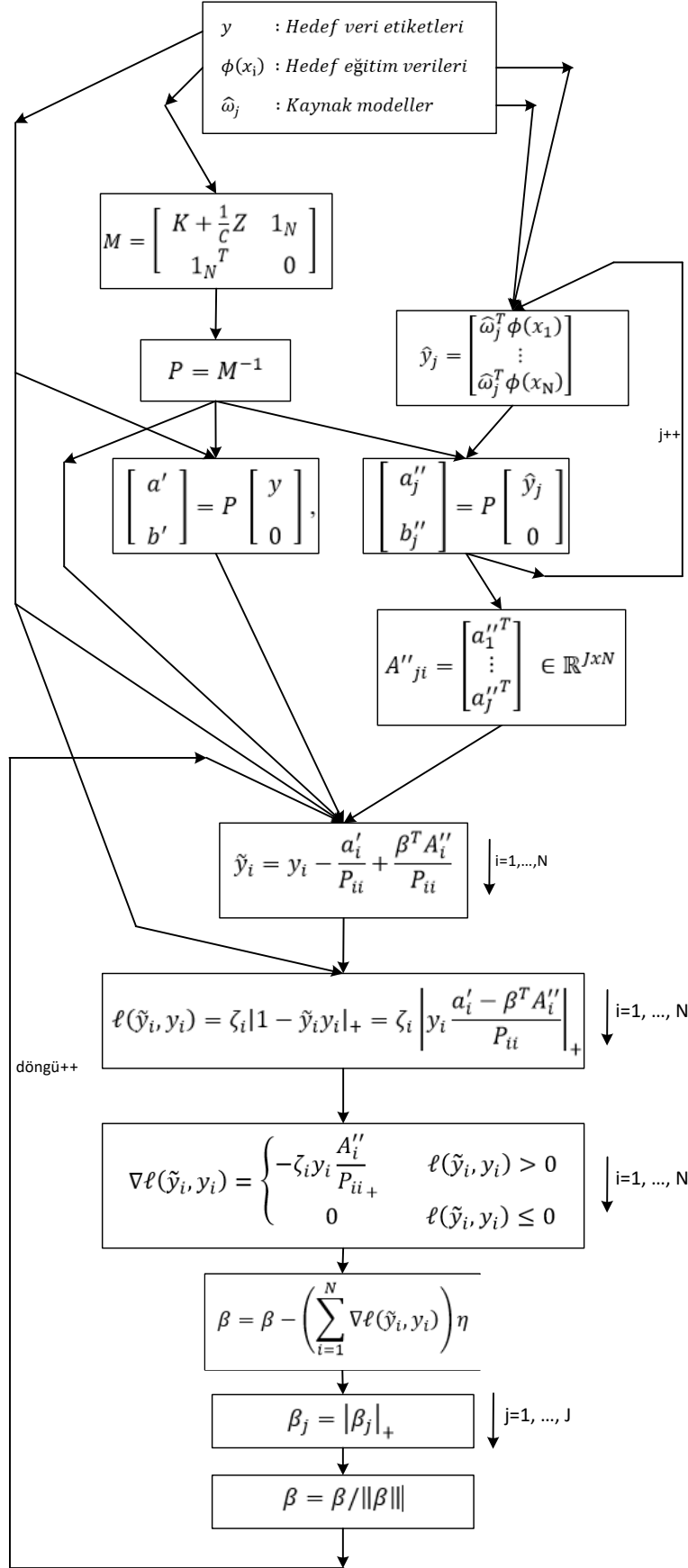
$$\nabla \ell(\tilde{y}_i, y_i) = \begin{cases} -\zeta_i y_i \frac{A_i''}{P_{ii+}} & \ell(\tilde{y}_i, y_i) > 0 \\ 0 & \ell(\tilde{y}_i, y_i) \leq 0 \end{cases} \quad (5.19)$$

Gradyanların bileşeninin alınmasından sonra η ile belirtilen adım uzaklığı katsayısı nispetinde beta değerleri en iyileştirme işlemine tabi tutulur.

$$\beta = \beta - \left(\sum_{i=1}^N \nabla \ell(\tilde{y}_i, y_i) \right) \eta \quad (5.20)$$

Beta değerlerinin yukarıda belirtilen en hızlı alçalma yöntemi ile döngüsel bir şekilde iyileştirilmesi sırasında, elde edilen gradyan değerlerine bağlı olarak beta değerlerinin negatif çıkması durumu oluşabilir. Negatif beta değeri, karşılık gelen kaynak verisinden bilginin aktarılmasından ziyade bilginin çıkartılması anlamına geleceğinden bu durum ilgili beta değerlerinin sıfır kabul edilmesi ile ortadan kaldırılır. Buradaki işlem, daha önce hata fonksiyonunda uygulanan $\beta_j = |\beta_j|_+$, $j = 1, \dots, J$ işlemidir. Daha sonra elde edilen beta değerlerine L_1 norm kısıtlaması uygulanır ve beta değerleri normleştirilir $\beta = \beta / \|\beta\|$. L_1 norm kısıtlamasının konmasının sebebi, kaynak sınıf sayısının, eğitim veri sayısından fazla olması durumuna karşı kaynak sınıfın ezberlenmesinin (overfit) önüne geçmektir. Aksi takdirde, uyarlamalı LS-SVM denkleminde kaynak sınıfına ait terimler baskın gelerek, elde edilen modelin hedef sınıfına ait modelleri öğrenme özelliği azaltılmış olacak ve elde edilen modelin kaynak sınıfına ait modellerin lineer birleşimine benzemesi sağlanacaktır.

Bu noktaya kadar elde edilen formüllerden yararlanarak bilgi aktarımı algoritmasını akış grafiği gösteriminde özetleyecek olursak Şekil 37’de yer alan grafik elde edilir.

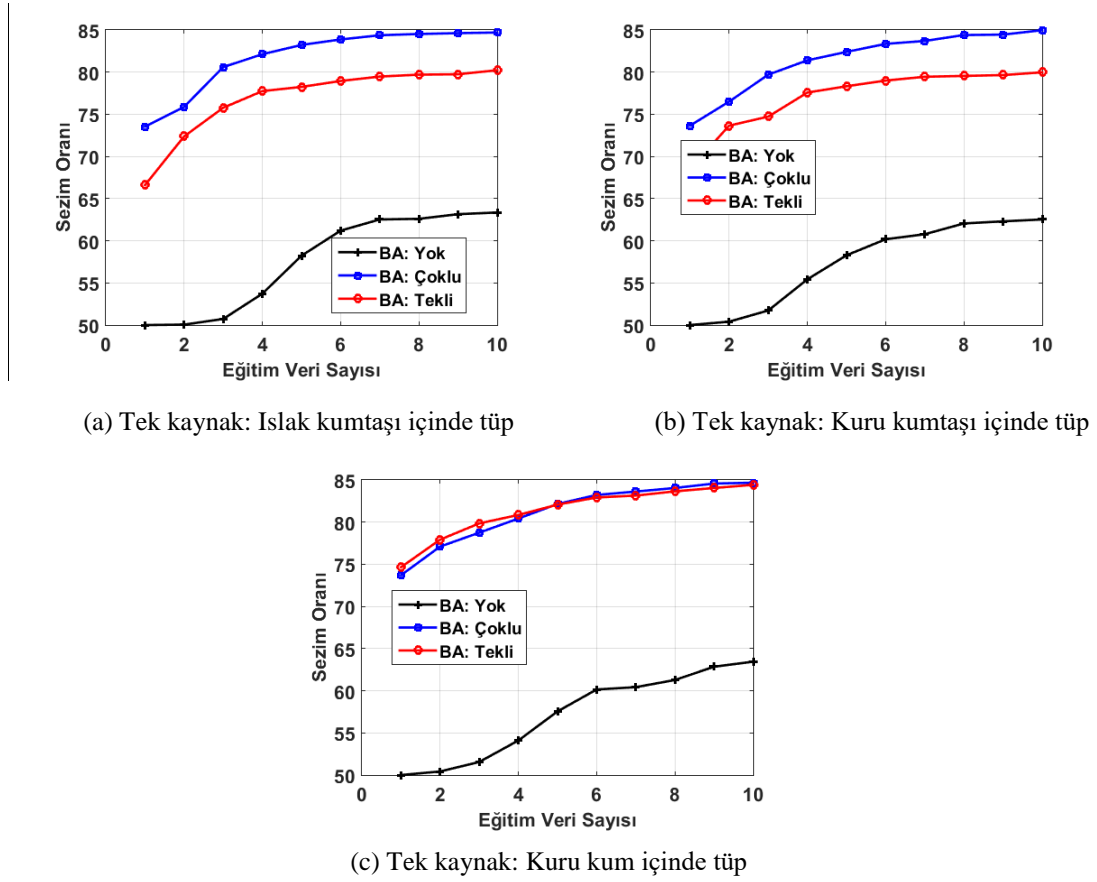


Şekil 37: Giriş verilerinden yola çıkarak Çok Modelli Bilgi Aktarımının gerçekleştirilmesinin akış grafiği

6. BİLGİ AKTARIMI İLE ELDE EDİLEN DENEYSEL SONUÇLAR

Eğer oluşturmak istediğimiz yeni sınıfa ait elimizde herhangi bir veri olmasaydı bu durumda sadece benzer sınıfları kullanarak yeni karşılaşıcağımız hedef sınıfını sezim oranları Tablo 5’te verildiği gibi olacaktı. Diğer taraftan, elimizde benzer kaynak sınıf olmasaydı fakat yeni oluşturacağımız sınıfa ait az sayıda eğitim verisi olsaydı bu durumda oluşan sezim oranları da Şekil 35’te verildiği gibi olacaktı. Her iki durumda da elde edilen başarı oranları yeterli değildir.

Eğer oluşturmak istediğimiz sınıfa ait elimizde benzer sınıflar varsa ve aynı zamanda yeni sınıfa ait az sayıda veri varsa bu durumda iki veri kümesini kullanarak başarılı sınıflandırma sonuçları elde edebiliriz. Şekil 38’de ıslak kum içinde tüp hedef sınıfı için bahsedilen bilgi aktarımının sonucunda elde edilen grafikler verilmiştir.



Şekil 38: Eğitimde kullanılan veri sayısı birden başlayarak 10'a kadar artırılırsa oluşan sezim oranları grafiği. (Siyah) Bilgi Aktarımının olmadığı durum. (Kırmızı) Tek kaynaktan Bilgi Aktarımının olduğu durum. (Mavi) 3 kaynaktan bilgi aktarımının olduğu durum.

Şekil 38’de gösterilen grafikler, LS-SVM eğitim ve testlerinin 20 defa arka arkaya çalıştırılması ile çıkan değerlerin ortalaması alınarak elde edilmiş sonuçları göstermektedir. Grafiklerdeki değerler ve standart sapmaları Tablo 6, Tablo 7 ve Tablo 8’de verilmiştir.

Tablo 6: Şekil 38 (a)'daki değerler ve standart sapmaları

Sınıflandırma Yöntemi	Eğitimde kullanılan hedef görüntü sayısına göre sezim oranları						
	(Adt) 1	(Adt) 2	(Adt) 3	(Adt) 4	(Adt) 6	(Adt) 8	(Adt) 10
BA: Yok	50±0	50±0.2	51±1.7	54±2.9	61±3.8	62±5.0	63±5.0
BA: Islak kumtaşı içinde tüp	67±9.9	72±7.9	76±6.4	78±2.2	79±2.2	80±2.2	80±2.3
BA: Çoklu	74±3.4	76±6.4	81±2.6	82±2.3	84±2.4	85±2.5	85±2.2

Tablo 7: Şekil 38 (b)'deki değerler ve standart sapmaları

Sınıflandırma Yöntemi	Eğitimde kullanılan hedef görüntü sayısına göre sezim oranları						
	(Adt) 1	(Adt) 2	(Adt) 3	(Adt) 4	(Adt) 6	(Adt) 8	(Adt) 10
BA: Yok	50±0	50±0.8	52±1.8	55±3.7	60±4.1	62±3.6	63±3.0
BA: Kuru kumtaşı içinde tüp	69±7.4	74±3.9	75±6.5	78±3.3	79±3.4	80±3.7	80±4.1
BA: Çoklu	74±3.3	77±4.1	80±4.1	81±3.4	83±2.7	84±2.6	85±2.4

Tablo 8: Şekil 38 (c)'deki değerler ve standart sapmaları

Sınıflandırma Yöntemi	Eğitimde kullanılan hedef görüntü sayısına göre sezim oranları						
	(Adt) 1	(Adt) 2	(Adt) 3	(Adt) 4	(Adt) 6	(Adt) 8	(Adt) 10
BA: Yok	50±0	50±0.7	52±1.5	54±2.6	60±5.0	61±5.0	63±3.7
BA: Kuru kum içinde tüp	75±3.9	78±3.7	80±4.0	81±3.0	83±2.7	84±2.6	84±3.0
BA: Çoklu	74±3.7	77±4.5	79±3.9	80±3.3	83±1.9	84±1.8	85±1.8

Sonuçlara bakıldığında, bilgi aktarımının olmadığı durumda (BA:yok), yani sadece az sayıda hedef verisi ile eğitimin olduğu durumda, başarı oranının oldukça düşük seviyede kaldığı görülmektedir. Örneğin, sadece 3 adet eğitim verisi ile sınıflandırıcı eğitilmiş olsaydı, tahmin oranları, rastgele tahmin değerinin (%50) çok ötesine geçemeyecektir. Diğer taraftan, sadece bir kaynaktan bilgi aktarımının olduğu durumda, 3 adet eğitim verisi ile oluşturulan sınıflandırıcı ile başarı oranı ıslak kumtaşı içinde tüp sınıfından bilgi aktarımı durumunda %76, kuru kumtaşı içinde tüp sınıfından bilgi aktarımı durumunda %75 ve kuru kum içinde

tüp sınıfından bilgi aktarımı durumunda %80 seviyesine çıkacaktır. Fakat üç kaynaktan da bilgi aktarımının olduğu durumda bu değer her zaman %80’ler seviyesinde olacaktır.

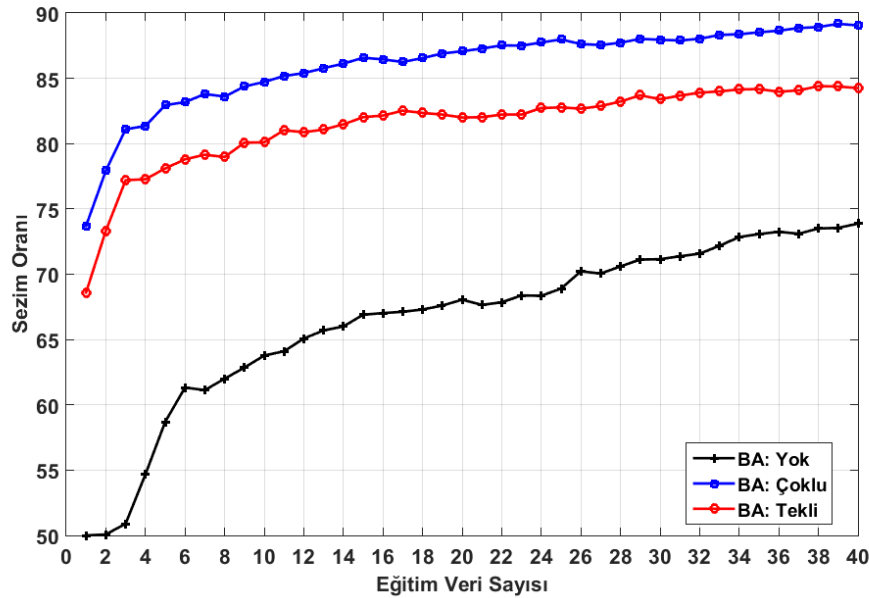
Şekil 38’de yer alan grafikler elde edilirken, 20 defa çalıştırılan uyarlamalı LS-SVM algoritmasında elde edilen beta değerlerinin ortalama değerleri ve standart sapmaları Tablo 9’da verilmiştir.

Tablo 9: Bilgi Aktarımı sonucunda elde edilen beta değerleri

Bilgi Aktarımının yapıldığı sınıf	Beta değeri	
Islak kumtaşı içinde tüp sınıfı	β_1	0.30 ± 0.05
Kuru kumtaşı içinde tüp sınıfı	β_2	0.28 ± 0.05
Kuru kum içinde tüp sınıfı	β_3	0.42 ± 0.04

Bu tablodaki değerlere bakıldığında, kuru kum içinde tüp sınıfının, bilgi aktarımında en fazla yararlanan sınıf olduğu ortaya çıkmaktadır. En az bilgi aktarımının gerçekleştirildiği sınıf ise kuru kumtaşı içinde tüp sınıfıdır. Daha önce belirtildiği gibi, L_1 norm kullanıldığı için beta değerlerinin toplamı 1’dir.

Eğitimde kullanılan veri sayısını artırdığımızda, tek kaynaktan bilgi aktarımının ıslak kumtaşı içinde tüp sınıfı olması durumunda elde edilen grafik Şekil 39’da verilmiştir.



Şekil 39: Daha fazla sayıda eğitim verisi ile elde edilmiş Bilgi Aktarımı sonuçları. Şekil, eğitim numunelerinin sayısı daha da artırıldığı durumda, Bilgi Aktarımının başarısının devam ettiğini vurgulamak için Şekil 38’nin uzatılmış versiyonudur.

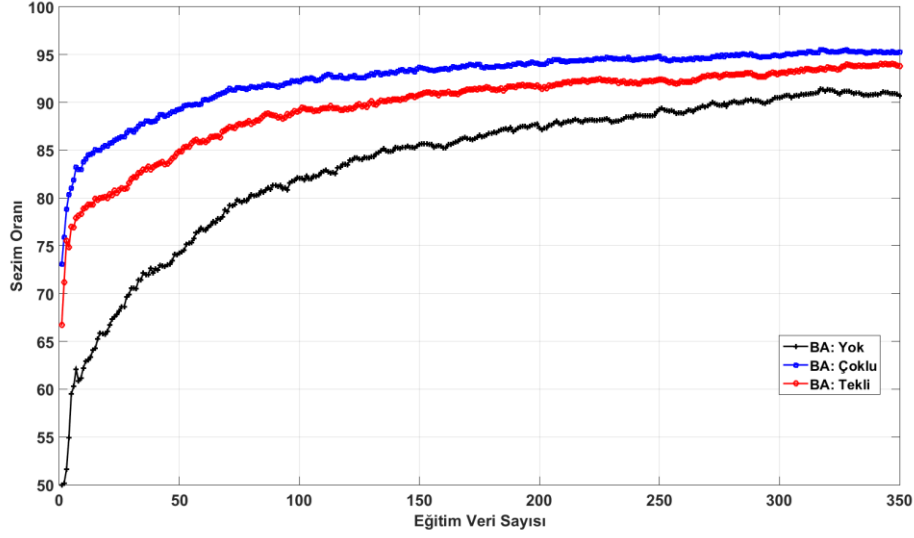
Şekil 39’da gösterilen grafik, LS-SVM eğitim ve testlerinin 20 defa arka arkaya çalıştırılması ile çıkan değerlerin ortalaması alınarak elde edilmiş sonuçları göstermektedir. Grafiklerdeki değerler ve standart sapmaları Tablo 10’da verilmiştir.

Tablo 10: Şekil 39’daki değerler ve standart sapmaları

Sınıflandırma Yöntemi	Eğitimde kullanılan hedef görüntü sayısına göre sezim oranları						
	(Adt) 10	(Adt) 15	(Adt) 20	(Adt) 25	(Adt) 30	(Adt) 35	(Adt) 40
BA: Yok	64±3.9	67±3.2	68±3.6	69±3.0	71±2.9	73±3.5	74±3.1
BA: Islak kumtaşı içinde tüp	80±2.8	82±2.7	82±2.5	83±3.1	83±2.8	84±2.9	84±2.1
BA: Çoklu	85±2.4	87±2.0	87±2.1	88±2.4	88±2.3	89±2.4	89±2.6

Şekil 34’te 4 farklı sınıf için bilgi aktarımı olmadığı durumda, veri sayısına karşı sezim başarı grafikleri verilmiştir. Bu grafiklerden hedef sınıfına ait olanına bakıldığında görülecektir ki, %90’ların hemen üzerinde başarı oranına 320’nin üzerinde veri ile erişilebilmektedir. Oysa Şekil 39’a bakıldığında görülecektir ki, 3 benzer kaynaktan bilgi aktarımı gerçekleştirildiği durumda, sadece 40 eğitim verisi ile yaklaşık %90 seviyesinde bir başarı yakalanabilmektedir. Başka bir bakış açısıyla, 40 veri ile bilgi aktarımı olduğunda %90’larda elde edilen başarı oranı, aynı sayıda eğitim verisi ile bilgi aktarımı olmaksızın %75’lerin altına inmektedir. Bilgi aktarımı ile sınıflandırıcının başarısı, bilgi aktarımı olmayanlara göre düşük eğitim sayılarında daha belirgindir. 40 eğitim örneğiyle % 15 daha iyi tahmin oranları düşük eğitim örneklerinde % 25’ten fazlaya ulaşabilmektedir. Örneğin sadece 3 eğitim verisinin olması durumunda, bilgi aktarımı olmaksızın sezim oranı %50’nin çok az üzerinde iken, 3 kaynaktan bilgi aktarımı olması durumunda bu oran %80 civarındadır.

Eğitimde kullanılan veri sayısını daha da artırdığımız durumda ise elde edilen grafik Şekil 40’da verilmiştir.



Şekil 40:Eğitimde kullanılan veri sayısının 350 olması durumunda BA sonuçları.

Şekil 40’da gösterilen grafik, LS-SVM eğitim ve testlerinin 20 defa arka arkaya çalıştırılması ile çıkan değerlerin ortalaması alınarak elde edilmiş sonuçları göstermektedir. Grafiklerdeki değerler ve standart sapmaları Tablo 11’de verilmiştir.

Tablo 11: Şekil 40’daki değerler ve standart sapmaları

Sınıflandırma Yöntemi	Eğitimde kullanılan hedef görüntü sayısına göre sezim oranları						
	(Adt) 50	(Adt) 100	(Adt) 150	(Adt) 200	(Adt) 250	(Adt) 300	(Adt) 350
BA: Yok	74±3.7	82±2.3	86±3.3	87±2.0	89±2.0	91±2.1	91±2.4
BA: Islak kumtaşı içinde tüp	85±2.4	89±1.8	91±2.1	92±1.5	92±1.7	93±1.6	94±1.4
BA: Çoklu	89±1.8	92±1.7	94±1.8	94±1.8	95±1.5	95±1.3	95±1.6

Şekilden de görüleceği üzere, veri sayısının daha da artırılması sezim oranlarını her üç grafik için de artırmaya devam etmiştir. 350 veri ile eğitim durumunda, bilgi aktarımının olmadığı durumda sezim oranı %91 civarında iken, çok kaynaktan bilgi aktarımı durumunda %95 civarındadır.

7. SONUÇLAR VE DEĞERLENDİRMELER

Bu çalışmada, farklı toprak tipi ortamlar için sınıflandırıcılar GPR simülasyon verileri kullanılarak elde edilmiştir. Oluşturulan sınıflar; ıslak kum içinde tüp, kuru kum içinde tüp, ıslak kumtaşı içinde tüp ve kuru kumtaşı içinde tüp sınıflarıdır. Bunlardan, ıslak kum içinde tüp sınıfı hedef sınıf olarak, diğer üç sınıf da kaynak sınıfı olarak seçilmiştir. Hedef olarak seçilen sınıfta az sayıda verinin, kaynak olarak seçilen sınıflarda ise çok sayıda verinin olduğu kabul edilmektedir. Farklı durumlar için elde edilen sınıflandırma sonuçları aşağıda özetlenmiştir.

Çok sayıda veri ile sınıflandırma:

Çok sayıda verinin bulunduğu kaynak sınıflar için oluşturulan sınıflandırıcılar ile tatmin edici sezim oranlarının elde edilebildiği, Şekil 34'te verilen grafiklerden görülebilmektedir. Elde edilen sonuçlar özet olarak şu şekildedir.

Tablo 12: Çok sayıda veri ile sınıflandırma

Sınıf	Eğitim Veri Sayısı	Sezim Oranı
Kuru kumtaşı içinde tüp	498	95
Kuru kum içinde tüp	402	92
Islak kumtaşı içinde tüp	348	93
Islak kum içinde tüp	384	91

Az sayıda veri ile sınıflandırma:

Az sayıda eğitim verisinin bulunması durumunda eğitilmiş bir sınıflandırıcının başarı oranının oldukça düşük olduğu, Şekil 35'te verilen grafikten görülmektedir. Elde edilen sonuçlar özet olarak şu şekildedir.

Tablo 13: Az sayıda veri ile sınıflandırma

Sınıf	Eğitim Veri Sayısı	Sezim Oranı
Islak kum içinde tüp	3	51
Islak kum içinde tüp	6	62

Bu sonuçlarla birlikte görülmüştür ki, az sayıda eğitim verisi ile oluşturulan sınıflandırıcı, verileri tahmin etmede tatmin edici sonuçlar vermemektedir.

Çapraz sınıflandırma:

Kaynak sınıfı için oluşturulmuş modellerinin, hedef sınıfa ait verileri tahmin etme oranların düşük olduğu elde edilen sonuçlar ile görülmüştür. Elde edilen sonuçlar şu şekildedir.

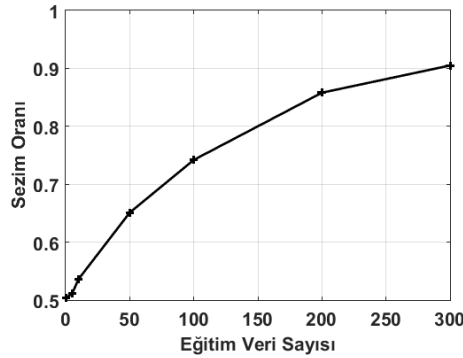
Tablo 14: Kaynak modellerinin, ıslak kum içerisinde tüp hedef verisini tanıma oranları

Kaynak Sınıf	Hedef Sınıf Verilerini Tahmin Oranları
Islak kumtaşı içinde tüp	57
Kuru kumtaşı içinde tüp	40
Kuru kum içinde tüp	60

Bu sonuçlarla birlikte görülmüştür ki, bir sınıf için oluşturulmuş model, başka bir sınıfa ait verileri tahmin etmede tatmin edici sonuçlar vermemektedir. Aynı zamanda buradaki sonuçlar, Tablo 9’da verilen beta değerleri ile örtüşmektedir.

Kaynak sınıfların birleşimi ile oluşan model ile sınıflandırma:

Uygulanan bu yöntemde, üç kaynak sınıfa ait tüm görüntüler bir araya getirilip bunlardan 400 tanesi eğitimde kullanılmak üzere rastgele seçilmiştir. Daha sonra hedef sınıftan artan sayı miktarlarında görüntü, oluşturulan bu eğitim veri kümesine dahil edilmiştir. Eğitim veri kümesine dahil edilmemiş 100 adet hedef görüntü verisi, oluşturulan bu eğitim verileri ile eğitilmiş LS-SVM modeli ile sınıflandırılmış ve çıkan grafik Şekil 41’de gösterilmiştir. Hedef sınıfından, eğitim kümesine dahil edilen veri sayıları sırasıyla 0, 5, 10, 50, 100, 200 ve 300’dir.



Şekil 41: Kaynak sınıf verilerinin birleşimi ile oluşturulan sınıflandırıcı başarı grafiği.

Şekil 41’de gösterilen grafik, LS-SVM eğitim ve testlerinin 20 defa arka arkaya çalıştırılması ile çıkan değerlerin ortalaması alınarak elde edilmiş sonuçları göstermektedir. Her döngüde veri seti ve test seti yeniden oluşturulmuştur. Kaynak sınıfların birleşimi ile üretilen sınıflandırıcı ile Bilgi Aktarımı durumunda oluşturulan sınıflandırıcıların karşılaştırması Tablo 15’te verilmiştir.

Tablo 15: Şekil 41’deki değerler ve standart sapmaları

Sınıflandırıcı Modeli	Eğitimde kullanılan hedef görüntü sayısına göre sezim oranları						
	(Adt) 0	(Adt) 5	(Adt) 10	(Adt) 50	(Adt) 100	(Adt) 200	(Adt) 300
Kaynak Sınıfların Birleşimi	50±3.8	51±4.1	54±4.6	65±4.1	74±4.7	86±3.7	91±2.6
Islak kumtaşı içinde tüpten Bilgi Aktarımı	-	78±2.2	80±2.3	85±2.4	89±1.8	92±1.5	93±1.6
Çok Kaynaktan Bilgi Aktarımı	-	83±2.3	85±1.8	89±1.8	92±1.7	94±1.8	95±1.3

Bilgi Aktarımı ile oluşturulan model, kaynak sınıfa ait verilerin birleşiminden oluşturulan sınıflandırıcı modelinden (tüm veri sayılarında) daha iyi sonuçlar vermektedir.

Kaynak sınıfların birleşimi ile üretilen sınıflandırıcı ile sadece hedef sınıf görüntülerinden üretilen sınıflandırıcının karşılaştırması Tablo 16’te verilmiştir.

Tablo 16: Kaynak sınıfların birleşim modeli ile hedef sınıf modelinin karşılaştırılması

Sınıflandırıcı Modeli	Eğitimde kullanılan hedef görüntü sayısına göre sezim oranları						
	(Adt) 0	(Adt) 5	(Adt) 10	(Adt) 50	(Adt) 100	(Adt) 200	(Adt) 300
Kaynak Sınıfların Birleşimi	50	51	54	65	74	86	91
Hedef Sınıfına ait sınıflandırıcı	-	57	64	77	82	85	90

Kaynakların birleşimi ile oluşturulan sınıflandırıcı, sadece hedef verileri ile oluşturulan sınıflandırıcıdan, 200 veriye kadar daha düşük performans verirken 200 veriden sonra daha iyi performans vermektedir

Bilgi aktarımı ile sınıflandırma:

Bölüm 6’da verilen sonuçlarla birlikte görülmüştür ki, benzer kaynak sınıfları için oluşturulmuş sınıflandırıcı modellerinden bilgi aktarımı yöntemi ile az sayıda verinin bulunduğu hedef sınıfı için oluşturulan sınıflandırıcı, tatmin edici sonuçlar vermektedir.

Eldedilen sonuların deęerlendirilmesi:

Bir toprak tipi iin oluřturulan bir sınıflandırıcının tahmin oranlarının farklı toprak tipine sahip bir sınıf iin yeterli olmadığı bulunmuřtur. Bu nedenle, simülasyon ortamında elde edilen veriler tarafından üretilen sınıflandırıcıların gerek toprakta gömülü olan tüpü doęru şekilde tahmin etmeme olasılıkları büyüktür. Öte yandan, LS-SVM tabanlı ok Kaynaklı Bilgi Aktarımı yöntemini kullanarak, farklı toprak tipleri iin eęitilmiş sınıflandırıcıların sınıflandırma bilgisini kullanarak yeni toprak tipi iin az miktarda veri kullanarak tatmin edici bir sınıflandırıcı elde etmek mümkündür. Bu, yöntemin geliştiricileri tarafından deneneni farklı gerek görüntü türleri ile elde edilen sonuçlarla karşılaştırıldığında ok benzerlik göstermektedir [2]. Bu durumda, simülasyon ortamında elde edilen veriler kullanılarak eęitilmiş bir sınıflandırıcının yukarıda belirtileni olası başarısızlığı, az sayıda gerek veri kullanarak ok kaynaktan bilgi aktarımı yöntemiyle üretilen sınıflandırıcı ile giderilecektir. Bu alıřmada, farklı toprak tipli simülasyon verileri üzerinde bilgi aktarımı yapılmıřtır. Gelecek alıřması olarak, simülasyon verileriyle elde edilen sınıflandırıcılardan bilgi aktarımı ile az sayıda gerek veri üzerinde öęrenme, denenebilir.

KAYNAKLAR

- [1] iCasualties: OIF - Deaths by IED, Wayback Machine, January 13, 2009. Online: <https://web.archive.org/web/20090113201909/http://icasualties.org/oif/IED.aspx>, (Mayıs 2018).
- [2] Tommasi, T., Orabona, F., Caputo, B., Learning categories from few examples with multi model knowledge transfer. *IEEE transactions on pattern analysis and machine intelligence*, 36(5), 928-941, 2014.
- [3] Biederman, I., Recognition-by-components: a theory of human image understanding. *Psychological review*, 94(2), 115, 1987.
- [4] Intrator, N., Edelman, S., Making a low-dimensional representation suitable for diverse tasks. In *Learning to learn* (pp. 135-157). Springer, Boston, MA, 1996.
- [5] Warren, C., Giannopoulos, A., Giannakis, I., gprMax: Open source software to simulate electromagnetic wave propagation for Ground Penetrating Radar. *Computer Physics Communications*, 209, 163-170, 2016.
- [6] Warren, C., Giannopoulos, A., Giannakis, I., An advanced GPR modelling framework: The next generation of gprMax. In *Advanced Ground Penetrating Radar (IWAGPR), 2015 8th International Workshop on* (pp. 1-4). IEEE, 2015.
- [7] Giannopoulos, A., Modelling ground penetrating radar by GprMax. *Construction and building materials*, 19(10), 755-762, 2005.
- [8] Daniels, D. J., Ground Penetrating Radar-The Institution of Electrical Engineers, London, *Google Scholar*, 2004.
- [9] Khan, U. S., Al-Nuaimy, W., Background removal from GPR data using eigenvalues. In *Ground Penetrating Radar (GPR), 2010 13th International Conference on* (pp. 1-5). IEEE, 2010.
- [10] Carevic, D., Clutter reduction and detection of minelike objects in ground penetrating radar data using wavelets. *Subsurface Sensing Technologies and Applications*, 1(1), 101-118, 2000.
- [11] Gunatilaka, A. H., Baertlein, B. A., Subspace decomposition technique to improve GPR imaging of antipersonnel mines. In *Detection and Remediation Technologies for Mines and Minelike Targets V* (Vol. 4038, pp. 1008-1019), 2000.

- [12] Bhuiyan, A., Nath, B., Anti-personnel landmine detection based on GPR and IR imaging: A review, 2006.
- [13] Borgioli, G., Capineri, L., Falorni, P., Matucci, S., Windsor, C. G., The detection of buried pipes from time-of-flight radar data. *IEEE Transactions on Geoscience and Remote Sensing*, 46(8), 2254-2266, 2008.
- [14] Yuksel, S. E., Bolton, J., Gader, P., Multiple-instance hidden markov models with applications to landmine detection. *IEEE Transactions on Geoscience and Remote Sensing*, 53(12), 6766-6775, 2015.
- [15] Gader, P. D., Mystkowski, M., Zhao, Y., Landmine detection with ground penetrating radar using hidden Markov models. *IEEE Transactions on Geoscience and Remote Sensing*, 39(6), 1231-1244, 2001.
- [16] Torrione, P. A., Morton, K. D., Sakaguchi, R., Collins, L. M., Histograms of oriented gradients for landmine detection in ground-penetrating radar data. *IEEE Transactions on Geoscience and Remote Sensing*, 52(3), 1539-1550, 2014.
- [17] Ko, K. H., Jang, G., Park, K., Kim, K., GPR-based landmine detection and identification using multiple features. *International Journal of Antennas and Propagation*, 2012..
- [18] Yuksel, S. E., Gader, P. D., Context-based classification via mixture of hidden Markov model experts with applications in landmine detection. *IET Computer Vision*, 10(8), 873-883, 2016.
- [19] Manandhar, A., Torrione, P. A., Collins, L. M., Morton, K. D., Multiple-instance hidden Markov model for GPR-based landmine detection. *IEEE transactions on geoscience and remote sensing*, 53(4), 1737-1745, 2015.
- [20] Baydar, B., Akar, G. B., Yüksel, S. E., Öztürk, S., Fusion of KLMS and blob based pre-screener for buried landmine detection using ground penetrating radar., In *Detection and Sensing of Mines, Explosive Objects, and Obscured Targets XXI* (Vol. 9823, p. 98231D), International Society for Optics and Photonics, 2016.
- [21] Núñez-Nieto, X., Solla, M., Gómez-Pérez, P., Lorenzo, H., GPR signal characterization for automated landmine and UXO detection based on machine learning techniques. *Remote sensing*, 6(10), 9729-9748, 2014.
- [22] Mesecan, I., Bucak, I. O., Efficient underground object detection for ground penetrating radar signals. *Defence Science Journal*, 67(1), 12, 2017.
- [23] Aydin, E., Yüksel, S. E., Buried target detection with ground penetrating radar using deep learning method. In *Signal Processing and*

- Communications Applications Conference (SIU), 2017 25th* (pp. 1-4). IEEE, 2017.
- [24] Lameri, S., Lombardi, F., Bestagini, P., Lualdi, M., Tubaro, S., Landmine detection from GPR data using convolutional neural networks. In *Signal Processing Conference (EUSIPCO), 2017 25th European* (pp. 508-512). IEEE..
- [25] Pham, M. T., Lefèvre, S., Buried object detection from B-scan ground penetrating radar data using Faster-RCNN. *arXiv preprint arXiv:1803.08414*, 2018.
- [26] Sun, Y., Li, J., Adaptive learning approach to landmine detection. *IEEE transactions on aerospace and electronic systems*, 41(3), 973-985, 2005.
- [27] Lowe, D. G., Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2), 91-110, 2004.
- [28] Suykens, J. A., Vandewalle, J., Least squares support vector machine classifiers. *Neural processing letters*, 9(3), 293-300, 1999.
- [29] Warren, C., Giannopoulos, A., *gprMax User Guide*, Release 3.0.0, June 12, 2016.
- [30] Tesfamariam, G. T., *Signal Processing Techniques for Landmine Detection Using Impulse Ground Penetrating Radar (ImGPR)*, Doctoral dissertation, Technische Universität, 2013.
- [31] Giannopoulos, A., *The investigation of transmission-line matrix and finite-difference time-domain methods for the forward problem of ground probing radar*, Doctoral dissertation, University of York, 1998.
- [32] Kunz, K. S., Luebbers, R. J., *The finite difference time domain method for electromagnetics*. CRC press, 1993.
- [33] Taflove, A., Hagness, S. C., *Computational electrodynamics: the finite-difference time-domain method*. Artech house, 2005.
- [34] Tu, J., Inthavong, K., Ahmadi, G., Reconstruction of the Human Airways. In *Computational Fluid and Particle Dynamics in the Human Respiratory System* (pp. 45-71). Springer, Dordrecht, 2013.
- [35] Approximating the Derivative, <http://mathfaculty.fullerton.edu/mathews/n2003/differentiation/numericaldiffproof.pdf>, (Mart 2018).
- [36] Lecture 11: LoG and DoG Filters, http://www.cse.psu.edu/~rtc12/CSE486/lecture11_6pp.pdf,(Mart 2018).

- [37] Lecture 10 Detectors and descriptors, http://cvgl.stanford.edu/teaching/cs231a_winter1314/lectures/lecture10_detector_descriptors.pdf, (Mart 2018).
- [38] VLFeat, <http://www.vlfeat.org/index.html>, (Haziran 2018).
- [39] Suykens, J. A., Van Gestel, T., De Brabanter, J., *Least squares support vector machines*. World Scientific, 2002.
- [40] Bertsekas, D. P., *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014.
- [41] Fletcher, R., *Practical methods of optimization*. John Wiley & Sons, 2013.
- [42] Haykin, S., Network, N., A comprehensive foundation. *Neural networks*, 2(2004), 41, 2004.
- [43] Least squares support vector machine, https://en.wikipedia.org/wiki/Least_squares_support_vector_machine, (Mart 2017).
- [44] Rodríguez, C. C., The Kernel Trick, <http://omega.albany.edu:8008/machine-learning-dir/notes-dir/ker1/ker1-1.html>, (Mart 2018).
- [45] Schölkopf, B., Smola, A. J., *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.
- [46] Cawley, G. C., Leave-one-out cross-validation based model selection criteria for weighted LS-SVMs. In *Neural Networks, 2006. IJCNN'06. International Joint Conference on* (pp. 1661-1668). IEEE, 2006.
- [47] The Moore-Penrose Pseudoinverse, <http://www.math.ucla.edu/~laub/33a.2.12s/mppseudoinverse.pdf>, (Mart 2018).

ÖZGEÇMİŞ

Kimlik Bilgileri

Adı Soyadı : Mehmet OTURAK
Doğum Yeri : ISPARTA
Medeni Hali : Evli
E-Posta : mehmet.oturak@gmail.com
Adresi : Oğuzlar Mahallesi 1387. Sokak 25/12
Çankaya / ANKARA

Eğitim

Lisans : Hacettepe Üniversitesi, Elektrik ve Elektronik Mühendisliği

Yabancı Dil ve Düzeyi

İngilizce : İyi

İş Deneyimi

MilSOFT Yazılım Teknolojileri A.Ş, (2007-2007)
TÜBİTAK UZAY Teknolojileri Araştırma Enstitüsü, (2007-Devam Ediyor)

Deneyim Alanları

Gömülü Sistemler, Analog ve Sayısal Devre Tasarımı, Makine Öğrenmesi

Tezden Üretilmiş Projeler ve Bütçesi

-

Tezden Üretilmiş Yayınlar

-

Tezden Üretilmiş Tebliğ ve/veya Poster Sunumu ile Katıldığı Toplantılar

-



HACETTEPE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
YÜKSEK LİSANS TEZ ÇALIŞMASI ORJİNALLİK RAPORU

HACETTEPE ÜNİVERSİTESİ
FEN BİLİMLER ENSTİTÜSÜ
ELEKTRİK ve ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI BAŞKANLIĞI'NA

Tarih: 10/07/2018

Tez Başlığı: ÖĞRENME AKTARIMI İLE SINIFLANDIRMA

Yukarıda başlığı gösterilen tez çalışmamın a) Kapak sayfası, b) Giriş, c) Ana bölümler d) Sonuç kısımlarından oluşan toplam 75 sayfalık kısmına ilişkin, 09/07/2018 tarihinde tez danışmanım tarafından *Turnitin* adlı intihal tespit programından aşağıda belirtilen filtrelemeler uygulanarak alınmış olan orijinallik raporuna göre, tezimin benzerlik oranı % 4'tür.


Uygulanan filtrelemeler:

- 1- Kaynakça hariç
- 2- Alıntılar hariç
- 3- 5 kelimedenden daha az örtüşme içeren metin kısımları hariç

Hacettepe Üniversitesi Fen Bilimleri Enstitüsü Tez Çalışması Orjinallik Raporu Alınması ve Kullanılması Uygulama Esasları'nı inceledim ve bu Uygulama Esasları'nda belirtilen azami benzerlik oranlarına göre tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Gereğini saygılarımla arz ederim.

10/07/2018


Tarih ve İmza

Adı Soyadı: Mehmet OTURAK
Öğrenci No: N12222492
Anabilim Dalı: Elektrik ve Elektronik Mühendisliği
Programı: Elektrik ve Elektronik Mühendisliği
Statüsü: Y.Lisans Doktora Bütünleşik Dr.

DANIŞMAN ONAYI

UYGUNDUR.


DR. ÖĞR. ÜYESİ SENİHA ESEN YÜKSEL
(Unvan, Ad Soyad, İmza)