

KOCAELİ ÜNİVERSİTESİ*MÜHENDİSLİK FAKÜLTESİ

HİPERSPEKTRAL GÖRÜNTÜLERDE HEDEF TESPİTİ

6. DÖNEM PROJESİ

Şafak GÜNEŞ - 170208015

Bölümü: Elektronik ve Haberleşme Mühendisliği

Danışman: Doç. Dr. Alp ERTÜRK

KOCAELİ, 2021

İÇİNDEKİLER

İÇİNDEKİLER.....	i
ŞEKİLLER DİZİNİ	ii
TABLOLAR DİZİNİ.....	ii
KISALTMALAR DİZİNİ.....	iii
ÖZET.....	iv
ABSTRACT.....	v
1.GİRİŞ.....	1
2.UZAKTAN ALGILAMA.....	2
2.1 Giriş.....	2
2.2. Pankromatik görüntüleme.....	4
2.3 Multispektral Görüntüleme	4
2.4. Hiperspektral Görüntüleme.....	4
3.HEDEF TESPİTİ.....	6
3.1 MF (Matched Filter, Eşleştirme Filtresi).....	7
3.2 CEM (Constrained Energy Minimization, Koşullu Enerji Minimizasyonu).....	8
3.3 ACE (Adaptive Coherence Estimator, Uyarlanabilir Tutarlılık Yaklaşımı).....	8
3.4 GLRT (Generalized Likelihood Ratio Test, Genelleştirilmiş Görecelik Oranı Testi).....	9
3.5 HUD (Hybrid Unstructured Detector, Yapısal Olmayan Hibrit Bulucu).....	9
3.6 AMSD (Adaptive Matched Subspace Detector, Uyumlu Eşleştirmeli Altuzay Bulucu).....	10
3.7 OSP (Orthogonal Subspace Projection, Ortogonal Altuzay İzdüşümü).....	10
4.KULLANILAN VERİSETLERİ.....	11
5. BAŞARIM.....	11
5.1 Roc Eğrileri.....	11
5.2 Area Under Curve.....	11
5.3 Recall-Precision.....	12
5.4 F-skoru.....	12
6.DENEYSEL SONUÇLAR.....	14
4.1 Airport Veriseti sonuçları.....	13
5.1 Airport Verisinin İstatiksel Sonuçları.....	15
5.2 Urban Veriseti Sonuçları.....	16
5.1 Urban Verisinin İstatiksel Sonuçları.....	18
REFERANSLAR.....	19
KAYNAK KODLARI.....	20

ŞEKİLLER DİZİNİ

Şekil 1.1 Hiperspektral görüntülerin kübik gösterimi.....	1
Şekil 2.1 Dalga boyuna göre hiperspektral veriler.....	3
Şekil 2.2 Hiperspektral küp.....	5
Şekil 2.3. Hiperspektral verilerde spektrumun elde edilmesi.....	5
Şekil 3.1. Hiperspektral görüntülemenin çalışma şeması.....	6
Şekil 4.1. Airport hiperspektral verisi.....	11
Şekil 4.2. Urban hiperspektral verisi.....	11
Şekil 5.1 Confusion matrisi.....	12
Şekil 6.1 Airport verisinin deneysel sonuçları.....	13
Şekil 6.2 Airport verisinin eşikten geçirilmiş sonuçları.....	13
Şekil 6.3 Airport verisine uygulanan yöntemlerin ROC eğrileri.....	14
Şekil 6.4 Airport verisinin ROC eğrisinin X eksenini logaritmik şekilde çizdirilmesi.....	14
Şekil 6.5. Airport verisine uygulanan yöntemlerin AUC ve F1-score sonuçları.....	15
Şekil 6.6 Urban verisinin deneysel sonuçları.....	16
Şekil 6.7 Urban verisinin eşikten geçirilmiş sonuçları.....	16
Şekil 6.8 Urban verisine uygulanan yöntemlerin ROC eğrileri.....	17
Şekil 6.9 Urban verisinin ROC eğrisinin X eksenini logaritmik şekilde çizdirilmesi.....	17
Şekil 6.10 Urban verisine uygulanan yöntemlerin AUC ve F1-score sonuçları.....	18

TABLolar DİZİNİ

Tablo 6.1 Airport verisetine uygulanan yöntemlerin istatistiksel karşılaştırılmaları.....	15
Tablo 6.2 Gulfport verisetine uygulanan yöntemlerin istatistiksel karşılaştırılmaları.....	18

KISALTMALAR DİZİNİ

ACE : Adaptive Cosine/Coherent Estimator
AMSD : Adaptive Matched Subspace Detector
AUC : Area Under Curve
CEM : Constrained Energy Minimization
FP : False Positives
GLRT : Generalized Likelihood Test Ratio
HUD : Hybrid Unstructured Detector
OSP : Orthogonal Subspace Projection
ROC : Receiver Operating Characteristic
SMF : Spectral Matched Filter
TP : True Positives

HİPERSPEKTRAL GÖRÜNTÜLERDE HEDEF TESPİTİ

Şafak GÜNEŞ

Anahtar Kelimeler: Hiperspektral Görüntüleme, Hedef Tespiti, Spektral İmza

Özet: Hiperspektral görüntüleme yüzey materyallerinden yansıyan enerjinin dar ve bitişik çok sayıda dalga boyu bandında ölçümüdür. Hiperspektral alıcıların en belirgin özelliği çok bantlı (multispektral) görüntülerden farklı olarak çok daha fazla banda sahip olmalarıdır. Ancak, alıcıları hiperspektral yapan özellik bant sayısından çok, bantların ne kadar dar ve birbirine ne kadar bitişik olduğudur. Hiperspektral bir veri dendiğinde üç boyutlu bir veri küpünden bahsedilmektedir. İlk iki boyut yersel bilgiyi gösterirken üçüncü boyut spektral bilgiyi göstermektedir. Bir noktadaki spektral veri incelendiğinde ortaya tek boyutlu bir işaret olan spektral işaret elde edilir. Bu işaretlere bakılarak farklı sınıfların birbirinden ayırt edilmesine çalışılır. [1]

Hiperspektral görüntülerde hedef tespiti birçok uygulama açısından önem arz etmektedir. Bu tür uygulamalara örnek olarak arama kurtarma faaliyetleri, savunma sistemlerinde hedefin bulunması, maden arama, sınır güvenliği ve mayın tespiti verilebilir[1]. Bu çalışmada iki farklı verisetinde hedeflerin tespit edilmesi amaçlanmıştır. Bunun için altı ayrı hiperspektral hedef tespit algoritması karşılaştırılmıştır ve sonuçlar ROC eğrileriyle karşılaştırılmış ve MF, CEM ve HUD algoritmalarının diğer algoritmalara oranla yüksek başarılarla ulaşabildiğini göstermektedir.

TARGET DETECTION IN HYPERSPECTRAL IMAGES

Şafak GÜNEŞ

Keywords : Hyperspectral Imaging, Target Detection, Spectral Signature

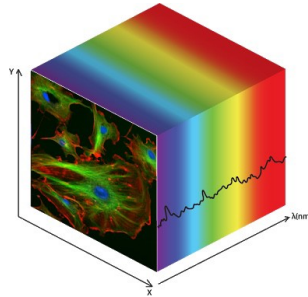
Abstract: Hyperspectral imaging is the measurement of the energy reflected from surface materials in narrow and contiguous many wavelength bands. The most distinctive feature of hyperspectral receivers is that they have much more bands, unlike multispectral images. However, what makes receivers hyperspectral is not so much the number of bands as how narrow and how adjacent the bands are. When a hyperspectral data is mentioned, a three-dimensional data cube is mentioned. The first two dimensions show the terrestrial information, while the third dimension shows the spectral information. When the spectral data at a point is examined, a one-dimensional spectral signal is obtained. By looking at these signs, it is tried to distinguish different classes from each other. [1]

Target detection in hyperspectral images is important for many applications. Examples of such applications are search and rescue activities, target detection in defense systems, mine exploration, border security and mine detection[1]. In this study, it is aimed to detect targets in two different datasets. For this, six different hyperspectral target detection algorithms are compared and the results are compared with ROC curves and show that MF, CEM and HUD algorithms can achieve high success compared to other algorithms.

1. GİRİŞ

Hiperspektral görüntü, aynı bölgenin farklı dalgaboylarındaki çok sayıda görüntüsünün birleşimidir. Yerden yansıyan güneş ışınları farklı dalgaboyundaki ışığı ölçebilen hiperspektral sensörlere düşer ve hiperspektral görüntü oluşur. Oluşan bu görüntü spektral parlaklık (radiance) görüntüsüdür. Yansıtıcılık cisimlerin ışığı yansıtma oranına göre 0 ile 1 arasında değişen bir değerdir. Maddelerin farklı dalgaboylarındaki ışıklar için yansıtıcılık değerleri farklıdır. Bir maddenin tüm bantlardaki yansıtıcılık değerlerinin birleşimi o maddenin spektral imzasıdır. Maddelerin spektral imzaları birbirinden farklıdır. Maddelerin bu özelliği sayesinde hedef tespit uygulamaları geliştirilebilir. Hedef tespit yöntemlerinde hedefin spektral imzası ile görüntüdeki pikseller arasındaki korelasyon göz önünde bulundurulur.[2]

Hiperspektral görüntüler, hedef tespit uygulamaları için yaygın olarak kullanılmaktadır. Ana amaç, spektral verinin piksellerini hedeflerin varlığı için aramaktır. Temel hedef tespit algoritmaları hedefin imzasına ihtiyaç duyar. Hedefler doğal ışık spektrumlarından farklı bilinen yansıma spektrumuna sahiptir. Spektral imza, bir malzemenin yansıma ya da yayılma dalga boylarına göre değişimdir. Bu tezde kullanılan hedef tespit uygulamalarında spektral imza kullanılmıştır.



şekil 1.1 : kübik gösterim[3]

2. UZAKTAN ALGILAMA

2.1 Giriş

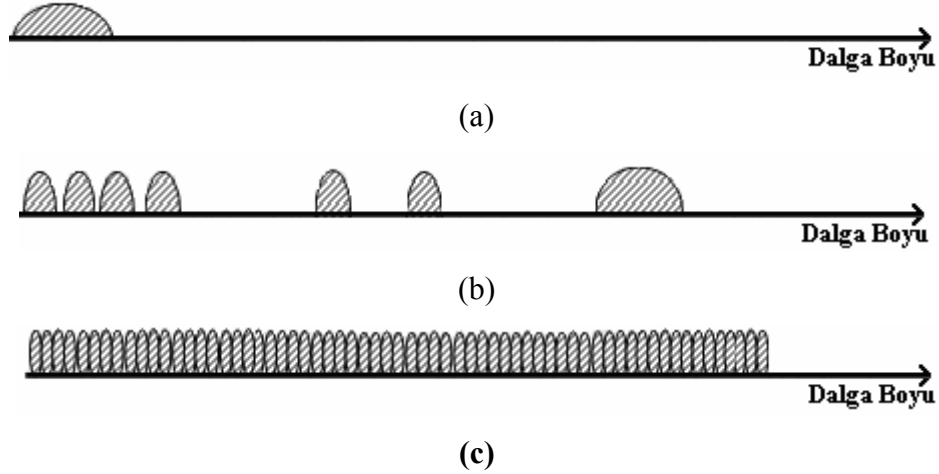
Uzaktan algılama elektromanyetik spektrumun mor ötesi ve mikrodalga bantlarında havadan ve uzaydan cisimlerin özelliklerini kaydetme ve inceleme tekniği olarak tanımlanmaktadır. Uzaktan algılama, nesnelerin belirli bir uzaklıktan mekanik veya elektronik cihaz kullanılarak gözlenmesi sonucu yeryüzünün ve yer kaynaklarının fiziksel bağlantı kurmadan incelenmesini sağlamaktadır. Yeryüzünün fiziksel yapısı hakkındaki pek çok bilgi uzaktan algılama teknikleri ile elde edilmektedir. Hızlı nüfus artışı, kısıtlı doğal kaynaklar ve çevre kirliliği, yeryüzü hakkında hızlı ve doğru bilgiye olan ihtiyacın artmasına neden olmaktadır. Temel veri kaynakları hava fotoğrafları, hava tarayıcıları ve uydu görüntüleridir. Bu tür veriler için kullanıcı ihtiyaçları gittikçe artmaktadır. Bu nedenlerden dolayı, uzaktan algılama özellikle gelişmekte olan ülkeler için değerli bir veri kaynağı oluşturmaktadır [4].

Ülkelerin ekonomik gelişmesinin temeli, doğal kaynaklarının verimli olarak kullanılmasına bağlıdır. Gelişmiş ülkeler kaynaklarını bu şekilde kullanırken, gelişmekte olan ülkeler genellikle doğal kaynaklarının nitelik ve niceliklerine ilişkin yeterli bilgilerden henüz yoksundurlar. Bu nedenle, özellikle az gelişmiş ülkelerde doğal kaynakların yeterli biçimde haritalanmamış olması sonucu, toprak ve su kaynakları, işlenen toprakların dağılımı, orman ve otlaklar ile madenlerin yerleri ve miktarları hakkında tam ve kesin veriler bulunmamaktadır.

Doğal çevrenin önemli bir bölümünün dinamik nitelikte olması, bunların bir kez belirlenmesi ile yetinilmeyip takip edilmelerini gerektirmektedir. Bunun için de, modern havacılık ve uzay teknolojilerinden yararlanılmaktadır. Uzaktan algılama yöntemleriyle havadan ve uzaydan elde edilen görüntüler yorumlanabilmektedir. Hava fotoğrafları, fotogrametrik yöntemle harita yapımında kullanılmakla birlikte, çeşitli mühendislik çalışmaları ve özellikle foto-yorumlama yöntemi ile doğal

kaynakların bulunmasında da kullanılmakta ve böylece, uzaktan algılama yönteminin en önemli verilerinden birini oluşturmaktadır. Uzaktan algılamanın en önemli kaynağını uzay fotoğrafları ve uydular oluşturmaktadır. Uzaktan algılama yöntemi; otoyol, devlet yolu, demiryolu ve boru hattı koridor seçimleri, sulama, baraj, madencilik ve ormancılık ön etüdlerinde, stereo uydu görüntülerinden etüd haritaları ve 3 boyutlu sayısal arazi modellerinin hazırlanmasında, deniz ve kıyı kirliliği etüdlerinde, uydu görüntülerinin işlenip uygun filitremeler yapılarak kirlilik haritalarının yapımında, tarımsal amaçlı, arazi kullanım ve toprak haritalarının etüdünde, orman kaynaklarının ön dökümlerinin yapımı ve haritalanmasında, orman yangınlarının yaptığı hasarların tespitinde ve görüntülerin işlenmesi-haritalandırılmasında, maden aramalarında, jeolojik etüdlerin yapımında, yer çalışmalarını süre ve maliyet açısından en aza indirmek amacıyla ve bunlara benzer daha birçok çalışmada başarıyla uygulanmaktadır.[4]

Pankromatik, multispektral, hiperspektral, algılayıcılar ile elde edilen görüntülere ait yaklaşık bant sayıları Şekil 2.1’de gösterilmektedir ve elde edilen görüntüler ile ilgili detaylı bilgi Bölüm 2.1, 2.2 ve 2.3’de sırasıyla anlatılmaktadır.



Şekil 2.1: (a): Pankromatik görüntülerde (b): multispektral görüntülerde (c): hiperspektral görüntülerde dalga boyuna göre elde edilen ortalama bant miktarları

2.2 Pankromatik Görüntüleme

Pankromatik görüntüler [5] elektromanyetik spektrumun geniş bir bölümünden yansıyan enerjiyi ölçebilen algılayıcılar tarafından elde edilmektedir. Birçok pankromatik algılayıcılar için elde edilen tek bant genellikle spektrumun görünen (visible) ile yakın kızılötesi bölümünü kapsamaktadır.

2.3 Multispektral Görüntüleme

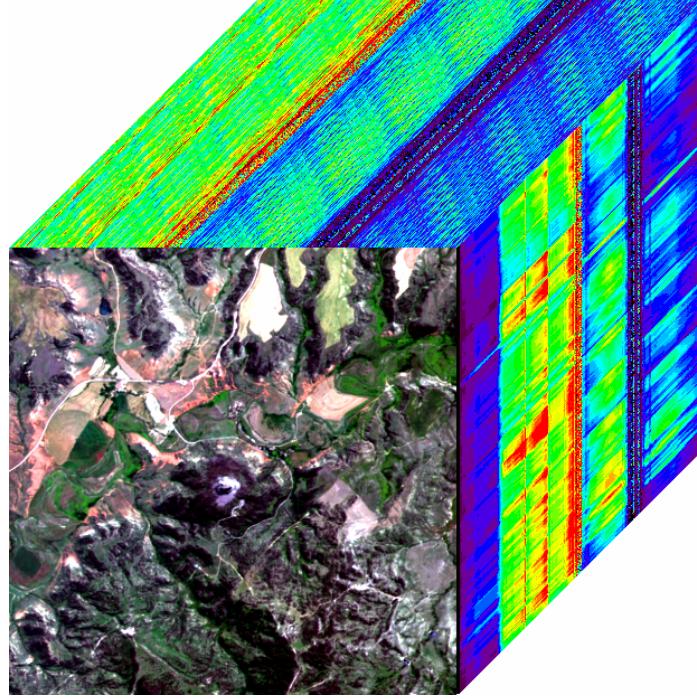
Multispektral görüntüler [6], elektromanyetik spektrumdaki birden fazla bantta ölçüm alan sensörlerle elde edilmektedir. Örneğin; alıcıların bir bölümü görünür kırmızı bölgesinde yansıyan enerjiyi ölçerken, diğer bir grup yakın kızılötesi enerjiyi ölçmektedir. Aynı iki algılayıcı dizilimi, aynı dalga boyunun değişik bölümlerindeki enerjiyi ölçebilmektedir. Bu şekilde elde edilen çoklu yansıma değerleri; renkli görüntü oluşturmak için birleştirilmektedir.

2.4 Hiperspektral Görüntüleme

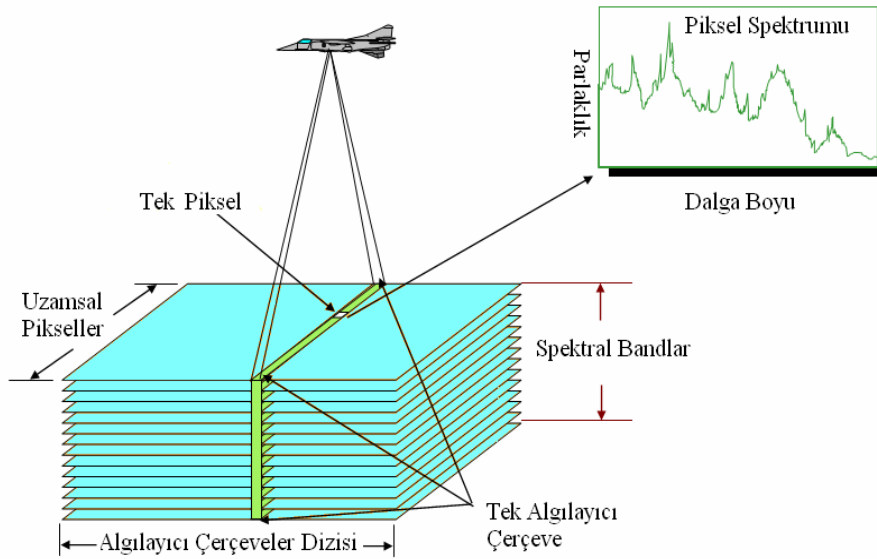
Günümüzde yaygın olarak kullanılan standart görüntü algılayıcıları (örneğin optik kamera veya kızılötesi kameralar) oldukça geniş bir dalga boyunda yansıyan ışığı tek bir imge üzerine düşürmektedir. Uydu ya da uçak üzerine yerleştirilmiş algılayıcılardan elde edilen hiperspektral görüntüler [7,8] ile birçok dar dalga boyu bandına ait imge verileri elde edilmekte ve bunun sonucunda dalga boyuna göre bir ayrışım sağlanmaktadır. Şekil 2.2’de birçok banttan oluşan örnek bir hiperspektral küp gösterilmektedir.

Hiperspektral görüntülerde her imge noktası, yani görüntü elemanı (piksel) için dalga boyuna bağlı bir spektrum bilgisi elde edilmektedir ve Şekil 2.3’de hiperspektral görüntülerdeki her piksel için dalga boyuna bağlı olarak elde edilen spektrum gösterilmektedir. Hiperspektral algılayıcılar ile dar bantlı ve sürekli aralıklandırılmış yüzlerce spektral bantta veri toplanabilmektedir. Farklı nesneler farklı dalga boylarında, farklı yansıma özellikleri gösterir. Bu özellik sayesinde cisimler birbirlerinden ayırt edilebilmektedir.

Hiperspektral görüntülerde piksel değerlerinin dalga boyuna göre değişimi elde edildiği için sınıflandırma ve hedef tespit yöntemlerinde oldukça başarılı sonuçlar elde edilmektedir. Sınıflandırma ve hedef tespit işlemleri standart görüntü algılayıcılarına oranla çok daha yüksek bir başarımla sağlanabilmektedir. Bu tez kapsamında hiperspektral görüntüler kullanarak hedef tespiti yöntemleri geliştirilmiştir.



Şekil 2.2: Hiperspektral küp

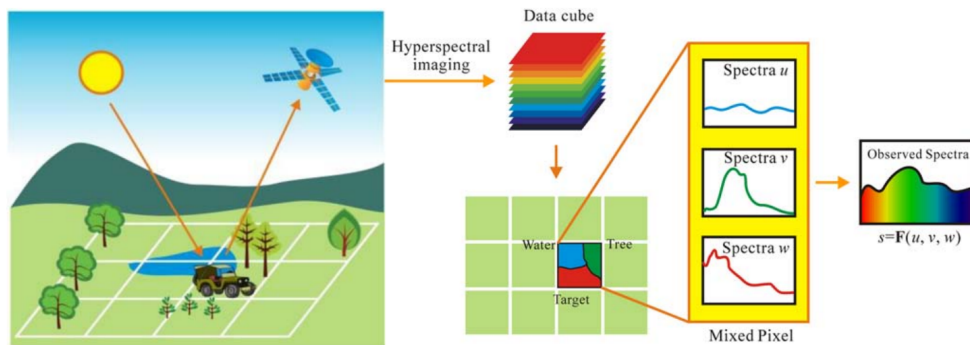


Şekil 2.3: Hiperspektral görüntülerdeki her piksel için dalga boyuna bağlı spektrumun elde edilmesi

3. HEDEF TESPİTİ

Hiperspektral bir görüntü hedef ve hedef olmayan olarak iki ayrı katman olarak düşünülebilir. Hedef genellikle çok küçük bir alan kaplamaktadır. Veride belirli bir şeyin varlığını veya konumunu tespit etme uygulaması literatürde hedef tespiti olarak adlandırılmaktadır. Tespit edilecek hedef, askeri bir hedef olabileceği gibi, belirli bir tarım ürünü, ağaç türü veya belirli bir mineral gibi birçok farklı şey olabilir. Hiperspektral verilerde hedef tespiti, spektral imza kavramı kullanılarak yapılır. Veride belirli bir spektral imzanın varlığı veya konumu tespit edilmek istenir. Spektral imzalar mevsim şartlarına, hava koşullarına, ilgili materyalin fiziksel veya kimyasal yapısındaki şartlara ve hiperspektral görüntünün alındığı alıcının yapısına göre değişim gösterebilmektedir. Hiperspektral görüntülerde uzamsal çözünürlük sıklıkla düşüktür. Uzamsal çözünürlüğünün yetersiz olduğu durumlarda ise bir piksel, kapladığı alanda kalan birden çok materyalin karışımı olarak yanıt vermektedir. Karışımli pikseller ayrıca uzamsal çözünürlük yeterli olduğunda da içkin karışım şeklinde gerçekleşebilmektedir. Hedef içeren piksel sayısı, içermeyen piksel sayısına göre çok az olduğu için, yüksek başarımlı istatistiksel kestirimler elde etmek zorlayıcıdır.[9]

Bu tezde kullanılan algoritmalar aşağıda maddeler halinde açıklanmıştır.



şekil 3.1 : hiperspektral görüntülemenin çalışma şeması[10]

3.1 SPECTRAL MATCHED FILTER (MF)

Denetimli hedef tespitinde, hedef spektral özellikleri genellikle tek bir hedef tarafından tanımlanır. spektrum veya bir hedef alt uzay olarak düşünülebilir. Standart bir teknik olarak spektral eşleştirilmiş filtre (SMF) hiperspektral görüntülerdeki ilgili nesnelerin tespitinde yaygın olarak kullanılır.[11]

Spektral eşleştirilmiş filtre (SMF) kullanarak hedef saptama, hiperspektral görüntülemeye ilgilenilen nesneleri tespit etmede iyi bilinen bir yaklaşımdır. SMF, hedefin spektral imzasının ve arka plan dağınıklık kovaryans matrisinin bulunduğu varsayılan bir doğrusal model varsayımına dayanmaktadır. Tipik olarak, hedef spektral imza, bir spektral kitaplıktan veya arka plan dağınıklık verilerinin tahmini kovaryans matrisi ile birlikte kullanılan bir dizi eğitim verisinden elde edilir.[12]

$$\begin{aligned} H_0: \mathbf{x} &= \mathbf{n} , & \text{hedef yok} \\ H_1: \mathbf{x} &= \mathbf{as} + \mathbf{n} , & \text{hedef var} \end{aligned}$$

şeklinde modellenir.

$$SMF(\mathbf{x}) = \mathbf{w}^T \mathbf{x} = \frac{\mathbf{s}^T \mathbf{C}^{-1} \mathbf{x}}{\mathbf{s}^T \mathbf{C}^{-1} \mathbf{s}}$$

Bu denklemde, \mathbf{x} test edilen piksel, \mathbf{s} hedefin spektral imzası, \mathbf{C} ise arkaplan kovaryans matrisidir. Bu kovaryans matrisi, imge genelinde veya test edilen piksel komşuluğunda hesaplanabilir. SMF, arkaplan ile hedefin aynı kovaryans matrisine, ancak farklı ortalamalara sahip olduğu varsayımından yola çıkar. Eşlenik süzgeç tabanlıdır ve süzgecin işaret-gürültü oranını en büyükleme amaçlanır.[10]

3.2 CONSTRAINED ENERGY MINIMIZATION (CEM)

Sınırlandırılmış Enerji Minimizasyonu (CEM), hiperspektral uzaktan algılama görüntülerinde hedef tespiti için yaygın olarak kullanılmaktadır. Ortalama çıkış gücünü minimize ederek gürültü ve bilinmeyen sinyal kaynaklarını bastırır bu sayede hedefi tespit eder. CEM, bir seferde yalnızca bir hedef kaynağı tespit edebilir. Tek bir görüntüde birden fazla hedefi eşzamanlı olarak tespit etmek için Çoklu Hedef CEM (MTCEM), Toplam CEM (SCEM) gibi yaklaşımlar kullanılabilir.[13]

Kısıtlı enerji minimizasyonu (CEM) algoritmasında, istenen hedef imzası, sonlu bir dürtü yanıt filtresi kullanılarak filtrelendir. Aynı anda arka plan bilgisini bastırmak için, bileşik bilinmeyen arka planın karakterizasyonu için arka plan kovaryans matrisi kullanılır .

CEM'in hesaplanması aşağıdaki gibidir:

$$D_{CEM}(\mathbf{x}) = \frac{\mathbf{S}^T \hat{\mathbf{R}}^{-1} \mathbf{x}}{\mathbf{S}^T \hat{\mathbf{R}}^{-1} \mathbf{S}}$$

3.3 ADAPTIVE COSINE ESTIMATOR (ACE)

Uyarlanabilir Tutarlılık Tahmincisi (ACE), Genelleştirilmiş Olabilirlik Oranı (GLR) yaklaşımından türetilmiştir. ACE, giriş spektrumlarının göreceli ölçeklendirmesine göre değişmez ve bu tür ölçeklemeye göre Sabit Yanlış Alarm Oranına (CFAR) sahiptir. Kısıtlı Enerji Minimizasyonuna (CEM) ve Eşleştirilmiş Filtrelemeye (MF) benzer şekilde ACE, bir görüntü sahnesindeki tüm son üyelerin bilgisini gerektirmez. [14]

Uyarlanabilir tutarlılık yaklaşımı (ACE), beyazlatılmış bir koordinat uzayında bilinen bir hedef vektör ile bir örnek vektör arasındaki açının karesinin kosinüsünü tahmin eder. Alan, istatistiğin hedef algılayıcı olarak performansını doğrudan etkileyen arka plan istatistiklerinin bir tahminine göre beyazlatılır.[14]

$$D_{ACE}(\mathbf{x}) = \frac{[(\mathbf{S} - \mu)^T \hat{\Gamma}^{-1} (\mathbf{x} - \mu)]^2}{[(\mathbf{S} - \mu)^T \hat{\Gamma}^{-1} (\mathbf{S} - \mu)] \cdot [(\mathbf{x} - \mu)^T \hat{\Gamma}^{-1} (\mathbf{x} - \mu)]}$$

3.4 GENERALIZED LIKEHOOD RATIO TEST (GLRT)

Genelleştirilmiş olabilirlik oranı testi (GLRT) ilk defa 1986'da Kelly tarafından geliştirilmiştir. GLRT, birden fazla birincil veriye ve yalnızca gürültü içeren bir dizi olan ikincil veriye dayalı olarak türetilir. Ayrıca, bilinen bir bozulma kovaryans matrisi için GLRT'nin ilk türetildiği iki aşamalı yaklaşımlarda kullanılmaktadır. Eşleşen sinyallerin tespiti ile uyumsuz sinyallerin reddi arasında iyi bir denge sağlayan dedektörler bulmak için, boş hipoteze hayali bir sinyal dahil edilerek iki aşamalı GLRT'de kullanılmaktadır. Dedektörlerin sabit yanlış alarm oranı özellikleri analiz edilir. Küçük numune boyutları için yeni önerilen GLRT'nin Kelly'nin GLRT'sinden daha iyi performans gösterebilir[15].

$$\begin{aligned} H_0 : \mathbf{x} &= \mathbf{v} && \text{not a target} \\ H_1 : \mathbf{x} &= \mathbf{S} + \mathbf{v} && \text{target} \end{aligned}$$

$$\begin{aligned} D_{GLRT}(x) \\ = \frac{[(\mathbf{S} - \mu)^T \hat{\mathbf{\Gamma}}^{-1} (\mathbf{x} - \mu)]^2}{[(\mathbf{S} - \mu)^T \hat{\mathbf{\Gamma}}^{-1} (\mathbf{S} - \mu)] \cdot [1 + \left(\frac{1}{M}\right) \cdot (\mathbf{x} - \mu)^T \hat{\mathbf{\Gamma}}^{-1} (\mathbf{x} - \mu)]} \end{aligned}$$

3.5 HYBRID UNSTRUCTURED DETECTOR (HUD)

Yapısal olmayan hibrit dedektör, arka planında ACE'ye benzer çok değişkenli normal dağılım gösterir. ACE dedektörü halihazırda beyazken HUD algoritması basitçe bolluk tahminlerinin yerine beyazlatılmış FCLS muadillerini kullanır[16].

$$D_{HUD}(x) = \frac{\mathbf{x}^T \hat{\mathbf{\Gamma}}^{-1} \mathbf{S} \hat{\mathbf{a}}}{\mathbf{x}^T \hat{\mathbf{\Gamma}}^{-1} \mathbf{x}}$$

3.6 ADAPTIVE MATCHED SUBSPACE DETECTOR

AMSD, büyük ölçüde arka plan alt uzayına ve hedef alt uzayına bağlıdır. Bununla birlikte, temsili bir hedef alt uzay oluşturmak, toplanan bir hiperspektral görüntüde hedef spektrumların sınırlı kullanılabilirliği nedeniyle zordur. Spektrum vektörü, spektral bant uzayının düşük boyutlu bir alt uzayında yer alacak şekilde sınırlandırılmıştır. Hedef ve arka plan tabanlarının doğrusal bir kombinasyonu olarak modellenir[16]

Şimdiye kadar tartışılan diğer tüm yöntemlerden farklı olarak, AMSD arka planda bollukları ve son üyeleri dikkate alır. "B" son üyeler olsun ve "ab" son üyelerin bolluğu olsun. Bu durumda hipotez aşağıdaki gibi olur[17]:

$$\begin{aligned} H_0: \mathbf{x} &= \mathbf{B}\mathbf{a}_b + \mathbf{v} && \text{hedef değil} \\ H_1: \mathbf{x} &= \mathbf{S} + \mathbf{B}\mathbf{a}_b + \mathbf{v} && \text{hedef} \end{aligned}$$

Bu hipotez ile AMSD şu şekilde tanımlanabilir:

$$D_{AMSD}(\mathbf{x}) = \frac{\mathbf{x}^T (\mathbf{P}_B^+ - \mathbf{P}_Z^+) \mathbf{x}}{\mathbf{x}^T \mathbf{P}_Z^+ \mathbf{x}}$$

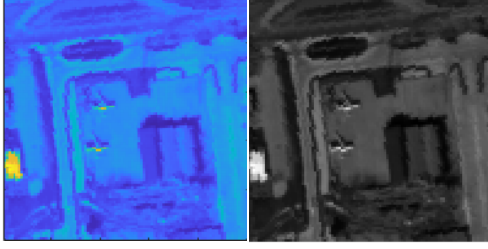
3.7 ORTHOGONAL SUBSPACE PROJECTION

OSP de AMSD gibidir, ancak yalnızca arka planda son üye matrisini kullanır[17]. OSP, önce hedef olmayanların tepkisini ortadan kaldırmak için bir ortogonal alt uzay projektörü tasarlar daha sonra istenen hedefi verilerden eşleştirmek için Eşleşen Filtre (MF) uygulanır. OSP, hedef imzalar farklı olduğunda verimli ve etkilidir. Hedef imza ve hedef olmayan imza arasındaki spektral açı küçük olduğunda, hedef sinyalin zayıflaması kaçınılmazdır ve OSP'nin performansı zayıf olabilir.[18]

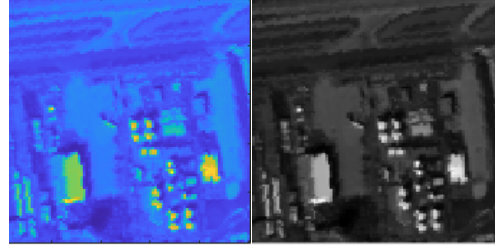
$$D_{OSP}(\mathbf{x}) = \frac{\mathbf{S}^T \mathbf{P}_B^+ \mathbf{x}}{\mathbf{S}^T \mathbf{P}_B^+ \mathbf{S}}$$

4. KULLANILAN VERİSETLERİ

Yapılan çalışmada kullanılan hiperspektral verilerden biri, AVIRIS alıcısı ile Los Angeles üzerinden alınmış bir hiperspektral veriden oluşmaktadır. İkinci veriseti ise abu URBAN Hiperspektral verisinden oluşmaktadır.



şekil 4.1: airport verisi



şekil 4.2 : urban verisi

5. BAŞARIM

5.1 ROC EĞRİLERİ

Sinyal algılama teorisinde, alıcı işletim karakteristiği (orijinal adıyla; Receiver Operating Characteristic - ROC) ya da sade biçimde ROC eğrisi olarak tanımlanmaktadır. ROC eğrisi, ikili sınıflandırma sistemlerinde ayırım eşik değerinin farklılık gösterdiği durumlarda, hassasiyetin kesinliliğe olan oranıyla ortaya çıkmaktadır. ROC daha basit anlamda doğru pozitiflerin, yanlış pozitiflere olan kesri olarak da ifade edilebilir[19].

5.2 AREA UNDER CURVE (AUC)

Bir başarımlı ölçütü olarak ROC eğrisinin altında kalan alanı ifade eder.

Eğrinin altındaki alan 0 ila 1 arasında olması gerekmektedir. Alan 1'e ne kadar yakınsarsa başarımlı o ölçüde artmış demektir.[20]

5.3 RECALL - PRECISSION

Recall (Hatırlama), Tüm pozitif sınıfların dışında, kaç örnek doğru tanımlandığını gösterir[20].

$$\text{Recall} = \frac{tp}{tp + fn}$$

Precision (keskinlik) ,Öngörülen tüm olumlu örneklerden, kaçının doğru tahmin edildiğini gösterir [20].

$$\text{Precision} = \frac{tp}{tp + fp}$$

5.4 F1-Skoru

Doğruluk (Accuracy) yerine F1 Score değerinin kullanılmasının en temel sebebi eşit dağılmayan veri kümelerinde hatalı bir model seçimi yapmamaktır. Ayrıca sadece False Negative ya da False Positive değil tüm hata maliyetlerini de içerecek bir ölçme metriğine ihtiyaç duyulduğu içinde F1 Score bizim için çok önemlidir. Aşağıda gösterildiği gibi hesaplanır[20]:

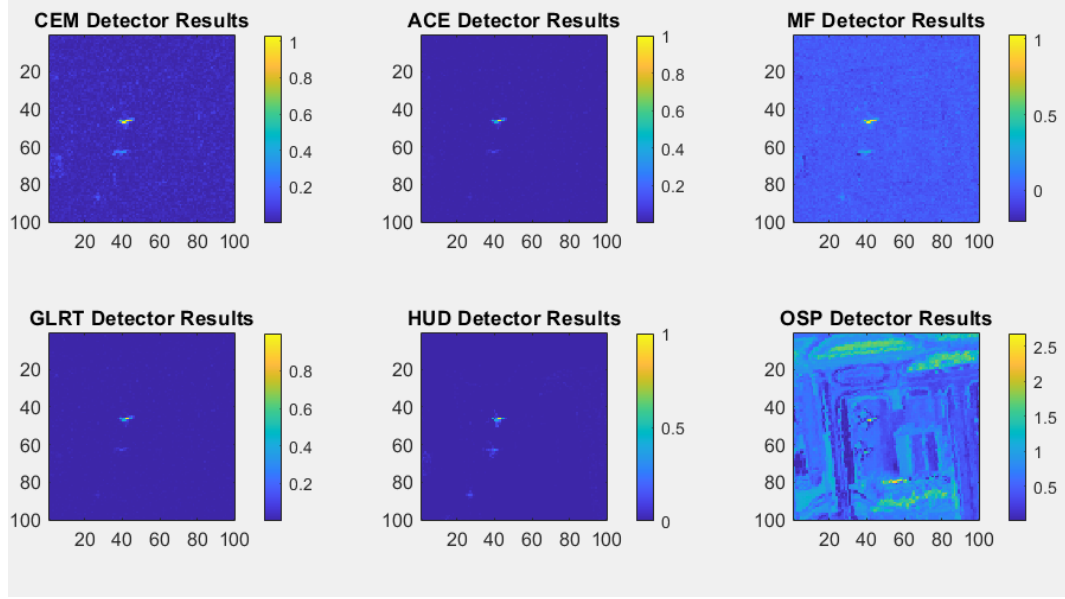
$$F - \text{Score}(\text{Balanced}) = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Tahminlenen (Predicted)	
Gerçekleşen (Actual)	True Positives (TP)
	False Negatives (FN)
Gerçekleşen (Actual)	False Positives (FP)
	True Negatives (TN)

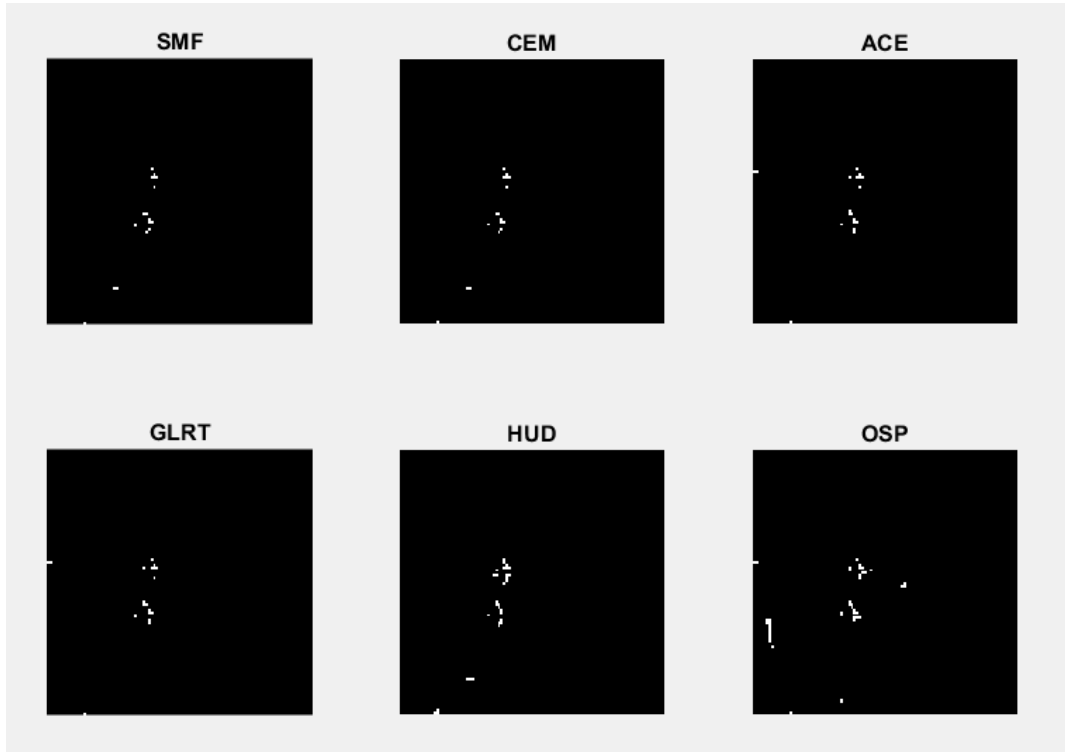
şekil 5.1 : confusion matrix[20]

6. DENEYSEL SONUÇLAR

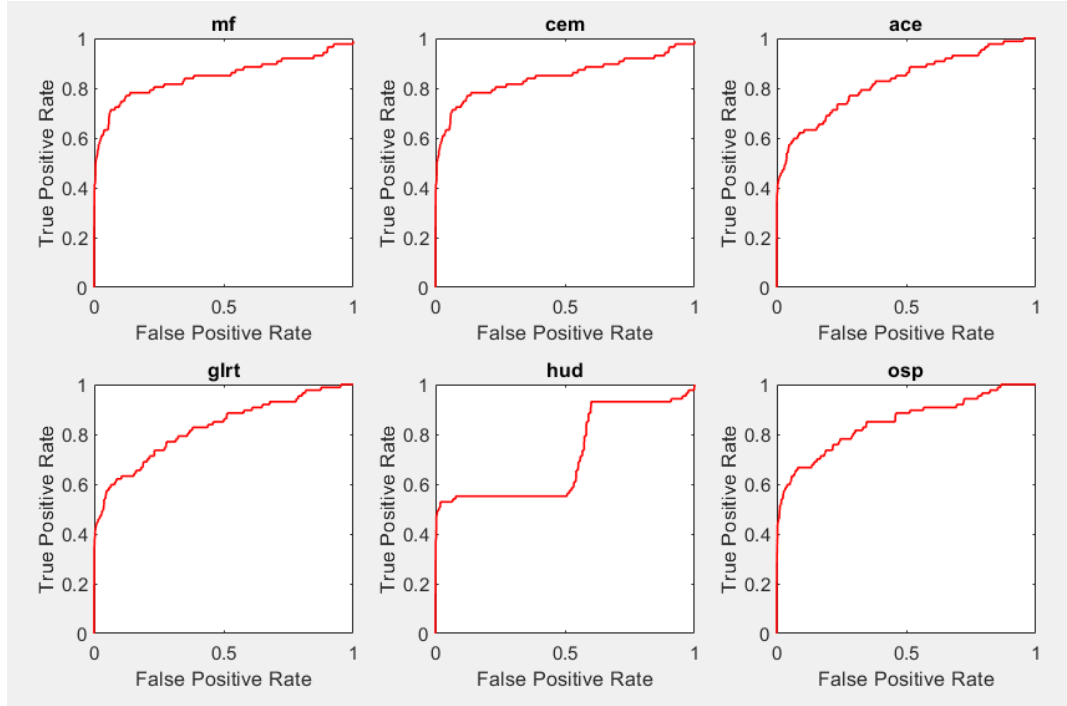
6.1 AIRPORT VERİSETİ SONUÇLARI



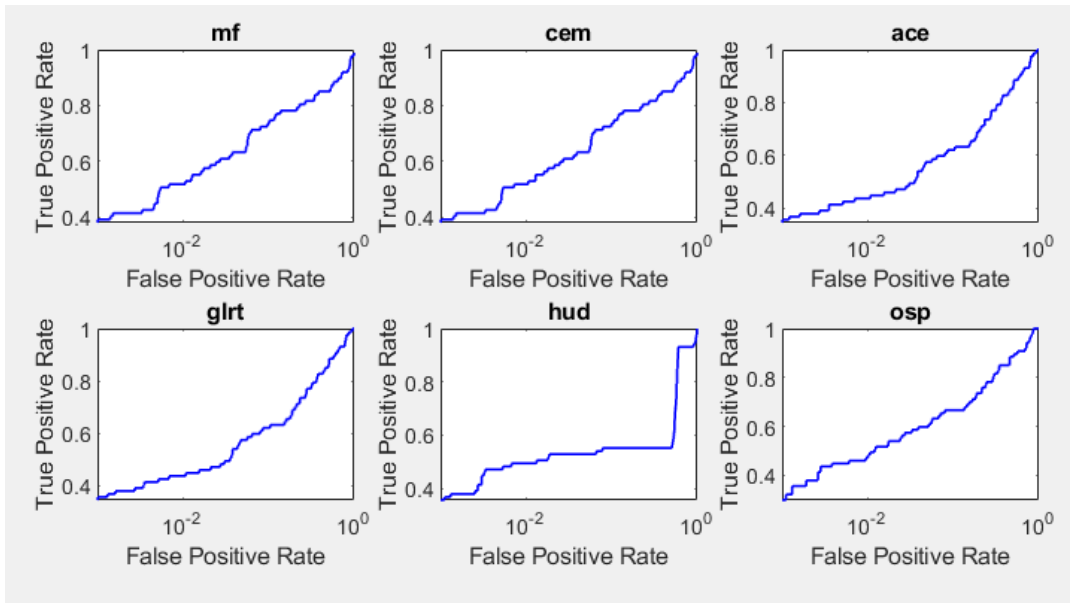
şekil 6.1: Airport verisinin deneysel hedef tespit sonuçları



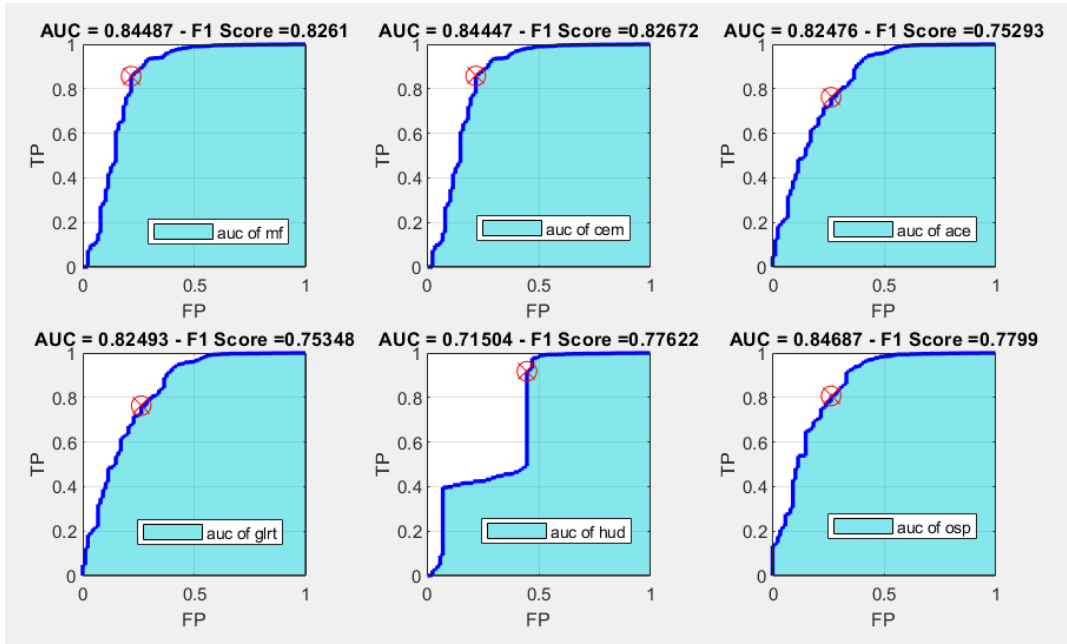
şekil 6.2: Airport verisinin eşikten geçirilmiş hedef tespit sonuçları



şekil 6.3: Airport verisine uygulanan yöntemlerin ROC eğrileri



şekil 6.4: ROC eğrisinin X eksenini logaritmik olacak şekilde çizdirilmesi

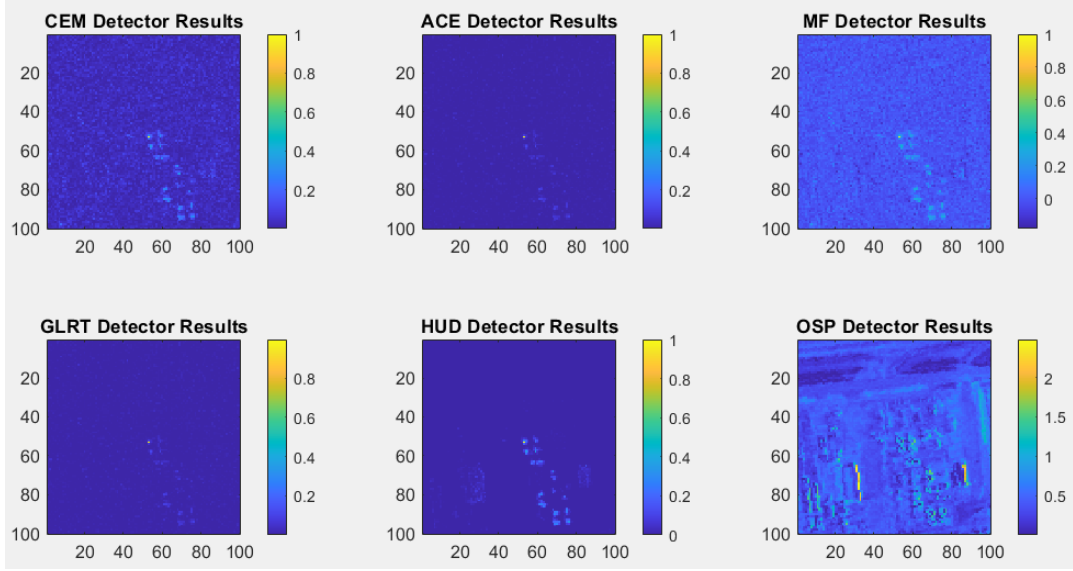


şekil 6.5: AUC ve F1 sonuçlarını gösteren ROC altında kalan alanlar

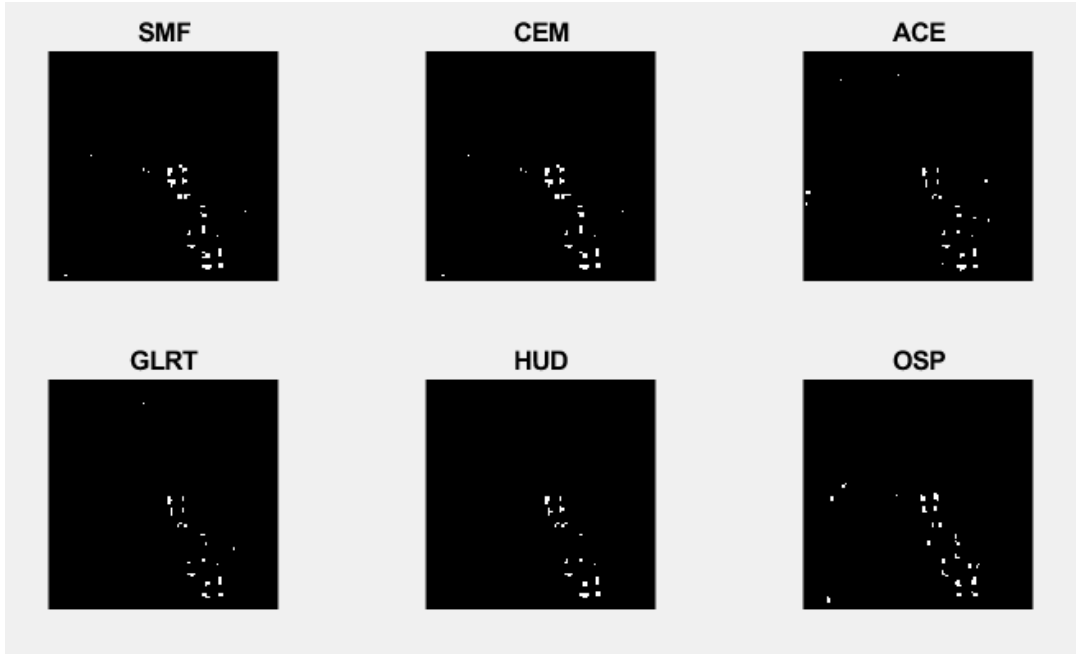
	MF	CEM	ACE	GLRT	HUD	OSP
AUC	0.8449	0.8445	0.8248	0.8249	0.7150	0.8469
F1-score	0.8261	0.8267	0.7529	0.7535	0.7762	0.7799
Accuracy	0.8197	0.8203	0.7499	0.7503	0.7354	0.7722
Recall	0.8565	0.8576	0.7623	0.7632	0.9178	0.8070
Precision	0.7978	0.7980	0.7438	0.7440	0.6725	0.7546

tablo 6.1 : Airport verisine uygulanan yöntemlerin istatistiksel sonuçları

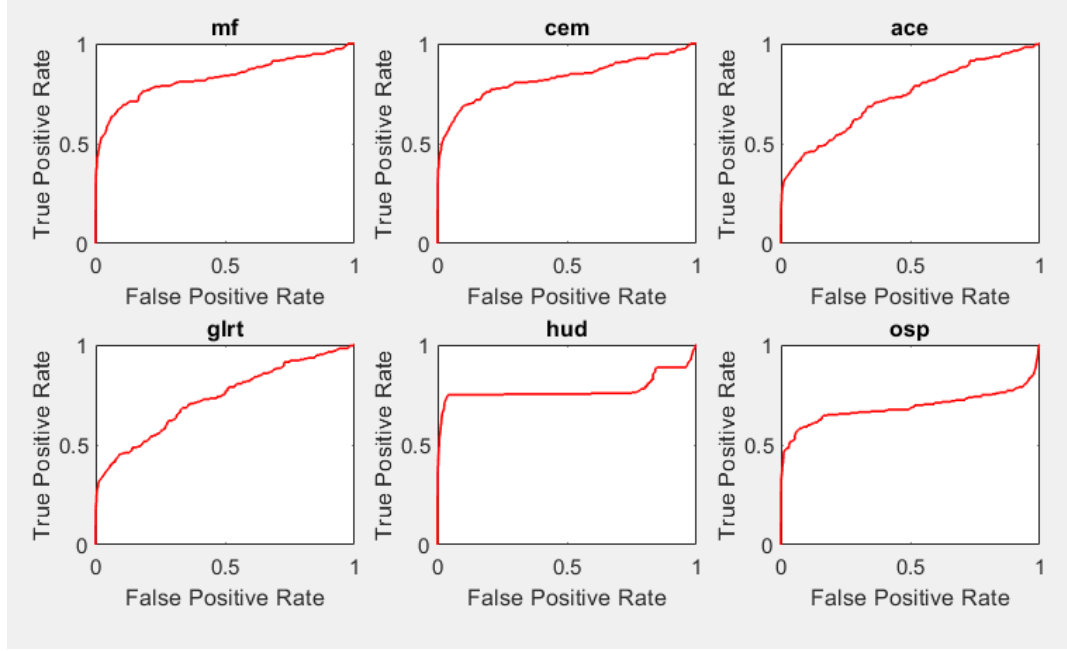
6.2 URBAN VERİSETİ SONUÇLARI



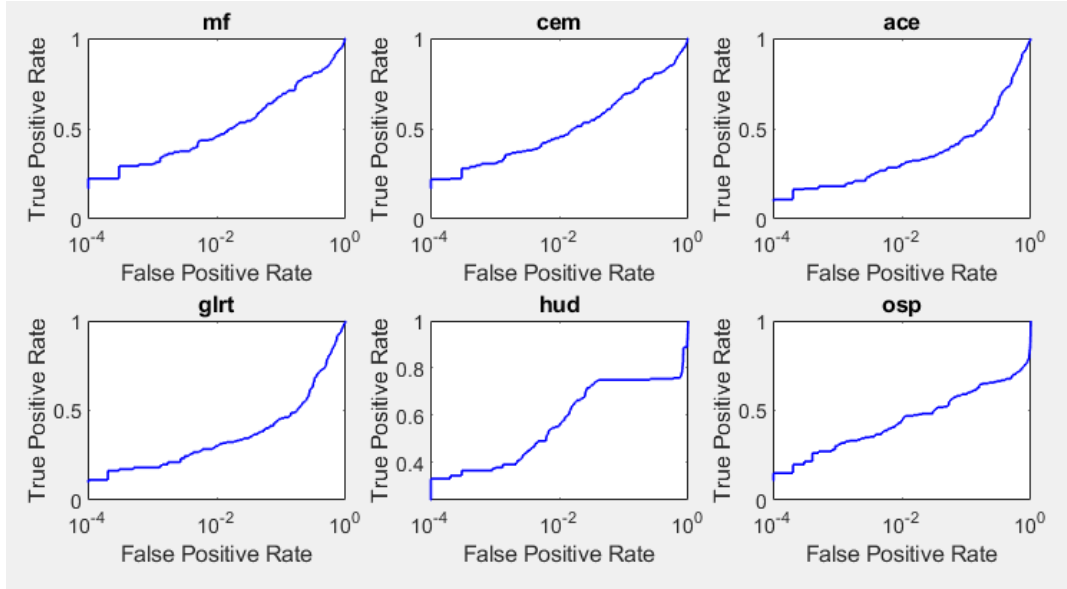
şekil 6.6 : Urban verisetinin deneysel hedef tespit sonuçlar



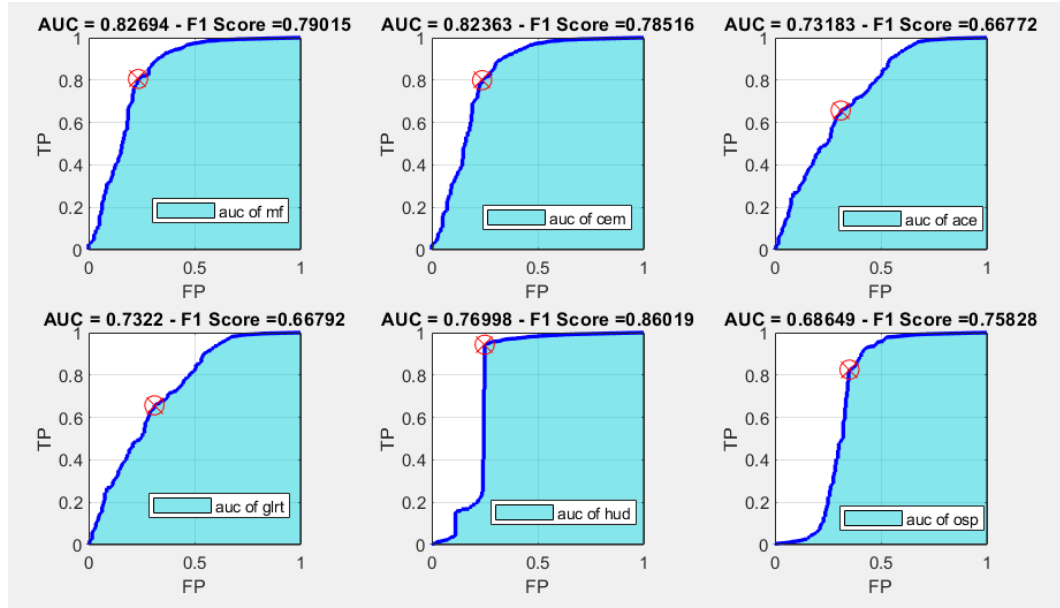
şekil 6.7 : Urban verisetinin eşikten geçirilmiş hedef tespit sonuçlar



şekil 6.8 : Urban verisetine uygulanan yöntemlerin ROC eğrileri



şekil 6.9 : ROC eğrilerinin X eksenine logaritmik olacak şekilde çizdirilmesi



şekil 6.10: AUC ve F1 sonuçlarını gösteren ROC altında kalan alanlar

	MF	CEM	ACE	GLRT	HUD	OSP
AUC	0.8269	0.8236	0.7318	0.7322	0.7700	0.6865
F1-score	0.7902	0.7852	0.6677	0.6679	0.8602	0.7583
Accuracy	0.7861	0.7812	0.6732	0.6734	0.8467	0.7369
Recall	0.8054	0.7998	0.6567	0.6570	0.9432	0.8255
Precision	0.7755	0.7710	0.6791	0.6792	0.7906	0.7012

tablo 6.2 : Urban verisetine uygulanan yöntemlerin istatistiksel sonuçları

REFERANSLAR

- [1] Oğuz Güngör, Esra Tunç Görmüş, Hiperspektral Görüntüler ve Analizi, Karadeniz Teknik Üniversitesi
- [2] <https://hoasys.fr/en/products/modular-hyper-spectral-camera>
- [3] Ahmet Karakaya, Target Detection in Hyperspectral Images, Elektrik ve Elektronik Mühendisliği Bölümü, Hacettepe Üniversitesi, Ankara, Türkiye
- [4] L. Zhang, L. Zhang, D. Tao, X. Huang and B. Du, "Hyperspectral Remote Sensing Image Subpixel Target Detection Based on Supervised Metric Learning," in IEEE Transactions on Geoscience and Remote Sensing, vol. 52, no. 8, pp. 4955-4965, Aug. 2014, doi: 10.1109/TGRS.2013.2286195.
- [5] Mahmood, N.N., Loh, K.F., Ahmad, S., "Remote sensing research in Malaysia", IEEE International Geoscience and Remote Sensing, 3, 1418-1420 (1997)
- [6] <https://eos.com/make-an-analysis/panchromatic/>
- [7] Biehl, L.L., Silva, L.F., "A multilevel multispectral data set analysis in the visible and infrared wavelength regions", Proceedings of the IEEE, 63, 164 -175 (1975)
- [8] <http://www.microimages.com>
- [9] Perkinson, M.-C., Lobb, D., Cutter, M., Renton, R., "Low Cost Hyperspectral Imaging from Space", 5th IAA Symposium on Small Satellites for Earth Observation, Berlin, Germany, (2001)
- [10] Alp Ertürk, "Uzaktan Algılama Teknolojileri" Lisans Ders Sunumu
- [11] Gao, L., Yang, B., Du, Q., & Zhang, B. (2015). Adjusted Spectral Matched Filter for Target Detection in Hyperspectral Imagery. Remote. Sens., 7, 6611-6634
- [12] N. M. Nasrabadi, "Regularized Spectral Matched Filter for Target Recognition in Hyperspectral Imagery," in IEEE Signal Processing Letters, vol. 15, pp. 317-320, 2008, doi:10.1109/LSP.2008.917805.
- [13] Qian Du, Hsuan Ren, and Chein-I Cheng. A Comparative Study of Orthogonal Subspace Projection and Constrained Energy Minimization. IEEE TGRS. Volume 41. Number 6. June 2003.
- [14] X Jin, S Paswater, H Cline. "A Comparative Study of Target Detection Algorithms for Hyperspectral Imagery." SPIE Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XV. Vol 7334. 2009
- [15] Vincent, François and Besson, Olivier One-step Generalized Likelihood Ratio Test for Subpixel Target Detection in Hyperspectral Imaging. (2020) IEEE Transactions on Geoscience and Remote Sensing.
- [16] Broadwater, J., & Chellappa, R. (2007). Hybrid Detectors for Subpixel Targets. IEEE Transactions on Pattern Analysis and Machine Intelligence, 29(11), 1891–1903. doi:10.1109/tpami.2007.1104
- [17] Seniha Esen Yuksel, Ahmet Karakaya. "Fusion of Target Detection Algorithms in Hyperspectral Images". ISSN:2147-67992.
- [18] Shen, S.S., et al., "A comparative study of target detection algorithms for hyperspectral imagery". 2009.
- [19] Signal detection theory and ROC analysis in psychology and diagnostics : collected papers; Swets, 1996
- [20] <https://bernatas.medium.com/roc-e%C4%9Frisi-ve-e%C4%9Fri-alt%C4%B1nda-kalan-alan-auc-97b058e8e0cf>
- [21] <http://davidkun.github.io/HyperSpectralToolbox/>

```

close all;
clear;
clc;
start = cputime;
%% dataseti iceri al ve normalize hale getir
load abu-urban-5;
M = data;
% load airport;
% M = data;

normalized_m= (M-min(min(min(M))))/(max(max(max(M)))-
min(min(min(M))));
M = normalized_m;
figure;imagesc(M(:,:,30));title('Hyperspectral Image');
axis image;
[h, w, p] = size(M);
%% spektral imzayi hesapla
target = squeeze(M(53,53,:)); %urban
% target = squeeze(M(46,42,:)); %airport
M = hyperConvert2d(M);
%% Matched Filter
r = MF(M, target);
out_mf = hyperConvert3d(r, h, w, 1);
subplot(3,3,3);imagesc(out_mf); title('MF Detector Results');
axis image;
%% Constrained Energy Minimization (CEM)
r = CEM(M, target);
out_cem = hyperConvert3d(r, h, w, 1);
figure;
subplot(3,3,1); imagesc(abs(out_cem)); title('CEM Detector
Results');
axis image;
%% Adaptive Cosine Estimator (ACE)
r = ACE(M, target);
out_ace = hyperConvert3d(r, h, w, 1);
subplot(3,3,2); imagesc(out_ace); title('ACE Detector Results');
axis image;
%% Generalized Likelihood Ratio Test (GLRT) detector
r = GLRT(M, target);
out_glrt = hyperConvert3d(r, h, w, 1);
subplot(3,3,4);imagesc(out_glrt); title('GLRT Detector Results');
axis image;
%% Estimate background endmembers
U = hyperAtgp(M, 5);
%% Hybrid Unstructured Detector (HUD)
r = HUD(M, U, target);
out_hud = hyperConvert3d(r, h, w, 1);

```

```

subplot(3,3,5);imagesc(abs(out_hud)); title('HUD Detector
Results');
axis image;

%% Orthogonal Subspace Projection (OSP)
r = OSP(M, U, target);
out_osp = hyperConvert3d(r, h, w, 1);
subplot(3,3,6);imagesc(abs(out_osp)); title('OSP Detector
Results');
axis image;

%% PLOTS
figure;
imshow(normalized_m(:,:,30),[]);title('Original Data')
figure;
subplot(3,3,1);
imshow(out_mf(:,:,1)>0.15,[]);title('SMF')
subplot(3,3,2);
imshow(out_cem(:,:,1)>0.15,[]);title('CEM')
subplot(3,3,3);
imshow(out_ace(:,:,1)>0.045,[]);title('ACE')
subplot(3,3,4);
imshow(out_glrt(:,:,1)>0.05,[]);title('GLRT')
subplot(3,3,5);
imshow(out_hud(:,:,1)>0.15,[]);title('HUD')
subplot(3,3,6);
imshow(out_osp(:,:,1)>0.6,[]);title('OSP')

figure;
subplot(3,3,1);
plot_roc(out_mf,:),map(:),'b');legend('auc of mf');
subplot(3,3,2);
plot_roc(out_cem,:),map(:),'b');legend('auc of cem');
subplot(3,3,3);
plot_roc(out_ace,:),map(:),'b');legend('auc of ace');
subplot(3,3,4);
plot_roc(out_glrt,:),map(:),'b');legend('auc of glrt');
subplot(3,3,5);
plot_roc(out_hud,:),map(:),'b');legend('auc of hud');
subplot(3,3,6);
plot_roc(out_osp,:),map(:),'b');legend('auc of osp');

stop = cputime;
disp(['target detection is finished in ' num2str(stop-start)
'seconds.']);

```

```

function [M] = hyperConvert2d(M)
%CONVERTS AN HSI CUBE TO A 2D MATRIX
if (ndims(M)>3 || ndims(M)<2)
    error('Input image must be m x n x p or m x n');
end
if (ndims(M) == 2)
    numBands = 1;
    [h, w] = size(M);
else
    [h, w, numBands] = size(M);
end
M = reshape(M, w*h, numBands).';
return;

function [img] = hyperConvert3d(img, h, w, numBands)
%CONVERTS AN 2D MATRIX TO A 3D DATA CUBE
if (ndims(img) ~= 2)
    error('Input image must be p x N. ');
end
[numBands, N] = size(img);
if (1 == N)
    img = reshape(img, h, w);
else
    img = reshape(img.', h, w, numBands);
end
return;

function [C] = hyperCov(M)
%COMPUTE COVARIANCE MATRIX
[p, N] = size(M);
u = mean(M.').';
for k=1:N
    M(:,k) = M(:,k) - u;
end
C = (M*M.)/(N-1);
End

function [R] = hyperCorr(M)
%COMPUTE THE SAMPLE AUTOCORRELATION MATRIX OF A 2D MATRIX.
[p, N] = size(M);
R = (M*M.)/N;
end

```

```

function [results] = MF(M, t)
%MATCH FILTER
[p, N] = size(M);
u = mean(M.').';
M = M - repmat(u, 1, N);
t = t - u;
R_hat = hyperCov(M);
G = inv(R_hat);
results = zeros(1, N);
tmp = t.'*G*t;
for k=1:N
    x = M(:,k);
    results(k) = (x.'*G*t)/tmp;
end

function [results] = CEM(M, target)
%CEM
if ndims(M) ~= 2
    error('Input image must be p x N.');
```

end

```

p = size(M,1);
if ~isequal(size(target), [p,1])
    error('Input target must be p x 1.');
```

end

```

R_hat = hyperCorr(M);
invRtarget = R_hat\target;
weights = ( target'*invRtarget ) \ invRtarget;
results = weights'*M;
end

function [results] = ACE(M, S)
% ACE
[p, N] = size(M);
u = mean(M.').';
M = M - repmat(u, 1, N);
S = S - repmat(u, 1, size(S,2));
R_hat = hyperCov(M);
G = inv(R_hat);
results = zeros(1, N);

tmp = (S.'*G*S);
for k=1:N
    x = M(:,k);
    results(k) = (S.'*G*x)^2 / (tmp*(x.'*G*x));
end
end

```

```

function [results] = GLRT(M, t)
%GLRT
[p, N] = size(M);
u = mean(M.').';
M = M - repmat(u, 1, N);
t = t - u;
R = inv(hyperCov(M));
results = zeros(1, N);
for k=1:N
    x = M(:,k);
    results(k) = ((t'*R*x)^2) / ((t'*R*t)*(1 + x'*R*x));
end

function [ U, indices ] = hyperAtgp( M, q, Maug )
%HYPERSPSCTRAL UNMIXING USING THE ATGP ALGORITHM
[p, N] = size(M);
U = [];
indices = [];
c = zeros(N, 1);
for x=1:N
    c(x) = (M(:, x).') * M(:, x);
end
[dummy, idx] = max(c);
indices = [idx];
U = [M(:,idx)];
start = 1;
if (nargin == 3)
    U = Maug;
end
for n=start:q-1
    P = eye(p) - U * inv(U.' * U) * (U. ');
    for x=1:N
        tmp = (P*M(:, x));
        c(x) = tmp.' * tmp;
    end
    [dummy, idx] = max(abs(c));
    indices = [indices idx];
    U = [U M(:,idx)];
end
end

```

```

function [ X ] = hyperNnls( M, U )
%NON-NEGATIVE CONSTRAINED LEAST SQUARES ON PIXELS OF M.
if (ndims(M) ~= 2)
    error('M must be a p x N matrix.');
```

```

end
if (ndims(U) ~= 2)
    error('U must be a p x q matrix.');
```

```

end
[p1, N] = size(M);
[p2, q] = size(U);
if (p1 ~= p2)
    error('M and U must have the same number of spectral
bands.');
```

```

end
Minv = pinv(U);
X = zeros(q, N);
MtM = U.'*U;
for n1 = 1:N
    drawnow;
    X(:, n1) = fnnls(MtM, U.' * M(:,n1));
end
return;
```

```

function [results] = HUD(M, B, S)
%HUD
[p, N] = size(M);
u = mean(M.').';
M = M - repmat(u, 1, N);
S = S - repmat(u, 1, size(S,2));
numTargets = size(S,2);
E = [S B];
q = size(E, 2);
R_hat = (M*M.)/N;
G = inv(R_hat);
results = zeros(1, N);
R = ones(q,1);
P = R - 1;
a_hat_tmp = hyperNnls(M, E);
for k=1:N
    x = M(:,k);
    a_hat = a_hat_tmp(:,k);
    a_hat = a_hat(1:numTargets);
    results(k) = (x.'*G*S*a_hat) / (x.'*G*x);
end
end
```

```

function [nu] = OSP(M, U, target)
%OSP
[p, N] = size(M);
P_U = eye(p) - U * pinv(U);
tmp = target'*P_U*target;
nu = zeros(N, 1);
for k=1:N
    nu(k) = (target'*P_U*M(:,k))/tmp;
end

function auc = plot_roc( predict, ground_truth,color )
%CALCULATE TP FP
x = 1.0;
y = 1.0;
pos_num = sum(ground_truth==1);
neg_num = sum(ground_truth==0);
x_step = 1.0/neg_num;
y_step = 1.0/pos_num;
[predict,index] = sort(predict);
ground_truth = ground_truth(index);
for i=1:length(ground_truth)
    if ground_truth(i) == 1
        y = y - y_step;
    else
        x = x - x_step;
    end
    X(i)=x;
    Y(i)=y;
end
%plot(X,Y,'-go','LineWidth',1,'MarkerSize',2);
semilogx(X,Y,color,'LineWidth',1,'MarkerSize',2);
xlim([10e-5 1]);
% plot(X,Y,color,'LineWidth',1,'MarkerSize',2);

xlabel('False Positive Rate');
ylabel('True Positive Rate');
title('ROC');

auc = -trapz(X,Y) ;
disp(auc)

roc_curve(X,Y); %%sonraki sayfadaki fonksiyon burada çağrıldı
end

```



```

function ROC_data = roc_curve(class_1, class_2, dispp, dispt)

    if(nargin<4), dispt = 1;    end
    if(nargin<3), dispp = 1;   end
    if(nargin<2), error('Params "class_1" or "class_2" are not
indicated.');
```

```

    class_1 = class_1(:);
    class_2 = class_2(:);
    s_data = unique(sort([class_1; class_2]));
    s_data(isnan(s_data)) = [];
    d_data = diff(s_data);
    if isempty(d_data), error('Both class data are the same!');
```

```

end
    d_data(length(d_data)+1,1) = d_data(length(d_data));
    thres(1,1) = s_data(1) - d_data(1);
    thres(2:length(s_data)+1,1) = s_data + d_data./2;
    curve = zeros(size(thres,1),2);
    distance = zeros(size(thres,1),1);
    for id_t = 1:1:length(thres)
        TP = length(find(class_2 >= thres(id_t)));
        FP = length(find(class_1 >= thres(id_t)));
        FN = length(find(class_2 < thres(id_t)));
        TN = length(find(class_1 < thres(id_t)));
        curve(id_t,1) = TP/(TP + FN);    % Sensitivity
        curve(id_t,2) = TN/(TN + FP);    % Specificity

    distance(id_t)= sqrt((1-curve(id_t,1))^2+(curve(id_t,2)-1)^2);
    end
    % Optimum threshold and parameters
    [~, opt] = min(distance);
    TP = length(find(class_2 >= thres(opt)));
    FP = length(find(class_1 >= thres(opt)));
    FN = length(find(class_2 < thres(opt)));
    TN = length(find(class_1 < thres(opt)));
    % Output parameters
    param.Threshold = thres(opt);
    param.Sensi = curve(opt,1);          % Sensitivity
    param.Speci = curve(opt,2);          % Specificity
    param.AROC = abs(trapz(1-curve(:,2),curve(:,1))); %AUC
    param.Accuracy = (TP+TN)/(TP+TN+FP+FN); % Accuracy
    param.PPV = TP/(TP+FP); % Positive predictive value
    param.NPV = TN/(TN+FN); % Negative predictive value
    param.FNR = FN/(FN+TP); % False negative rate
    param.FPR = FP/(FP+TN); % False positive rate
    param.FDR = FP/(FP+TP); % False discovery rate
    param.FOR = FN/(FN+TN); % False omission rate
    param.F1_score = 2*TP/(2*TP+FP+FN); % F1 score

```

```

param.TP = TP;      % No. true positives
param.FP = FP;      % No. false positives
param.FN = FN;      % No. false negatives
param.TN = TN;      % No. true negatives

if(dispp == 1)
    fill_color = [11/255, 208/255, 217/255];
    fill([1-curve(:,2); 1], [curve(:,1); 0],
fill_color, 'FaceAlpha', 0.5);
    hold on; plot(1-curve(:,2), curve(:,1), '-b',
'LineWidth', 2);
    hold on; plot(1-curve(opt,2), curve(opt,1), 'or',
'MarkerSize', 10);
    hold on; plot(1-curve(opt,2), curve(opt,1), 'xr',
'MarkerSize', 12);
    hold off; axis square; grid on; xlabel('FP');
ylabel('TP');
    title(['AUC = ' num2str(param.AROC) ' - ' 'F1 Score = '
num2str(param.F1_score)]);
end

% Log screen parameters if required
if(dispt == 1)
    fprintf('\n ROC CURVE PARAMETERS\n');
    fprintf(' ----- \n');
    fprintf(' - Distance:      %.4f\n', distance(opt));
    fprintf(' - Threshold:     %.4f\n', param.Threshold);
    fprintf(' - Sensitivity:    %.4f\n', param.Sensi);
    fprintf(' - Specificity:    %.4f\n', param.Speci);
    fprintf(' - AROC:          %.4f\n', param.AROC);
    fprintf(' - Accuracy:      %.4f\n', param.Accuracy);
    fprintf(' - PPV:           %.4f\n', param.PPV);
    fprintf(' - NPV:           %.4f\n', param.NPV);
    fprintf(' - FNR:           %.4f\n', param.FNR);
    fprintf(' - FPR:           %.4f\n', param.FPR);
    fprintf(' - FDR:           %.4f\n', param.FDR);
    fprintf(' - FOR:           %.4f\n', param.FOR);
    fprintf(' - F1 score:      %.4f\n', param.F1_score);
    fprintf(' \n');
end

% Assinging parameters and curve data
ROC_data.param = param;
ROC_data.curve = curve;

end

```