





# Hiperspektral görüntülerin otomatik uyarlamalı ışıklılık dönüşümü ve 3D-DCT yöntemi kullanılarak sıkıştırılması

## Compression of hyperspectral images using automatic adaptive luminance transform and 3D-DCT method

Ergün CAN<sup>1</sup> , Ali Can KARACA<sup>2\*</sup> , Oğuzhan URHAN<sup>3</sup> , Mehmet Kemal GÜLLÜ<sup>4</sup> 

<sup>1</sup>Elektrik-Elektronik Mühendisliği Bölümü, Mühendislik Fakültesi, Piri Reis Üniversitesi, İstanbul, Türkiye.  
ecan@pirireis.edu.tr

<sup>2,3,4</sup>Elektronik-Haberleşme Mühendisliği Bölümü, Mühendislik Fakültesi, Kocaeli Üniversitesi, Kocaeli, Türkiye.  
alicankrc@gmail.com, urhano@kocaeli.edu.tr, kemalg@kocaeli.edu.tr

Geliş Tarihi/Received: 27.06.2019  
Kabul Tarihi/Accepted: 19.12.2019

Düzeltilme Tarihi/Revision: 18.12.2019

doi: 10.5505/pajes.2019.85126  
Araştırma Makalesi/Research Article

### Öz

Hiperspektral görüntüleme, farklı uygulama alanlarındaki kullanımı ile son yıllarda oldukça popüler bir konu haline gelmiştir. Yüksek depolama alanlarına ihtiyaç duyan hiperspektral görüntülerin yüksek verim ve kalite ile sıkıştırılması gerekmektedir. Bu çalışmada, hiperspektral görüntülerin kayıplı sıkıştırılması için otomatik uyarlamalı ışıklılık dönüşümü ve üç-boyutlu ayrık kosinüs dönüşümünü (3D-DCT) kullanan özgün bir yöntem önerilmektedir. Önerilen yöntemde ilk olarak hiperspektral verideki spektral bantlar gruplanmış ve ön işlem olarak otomatik uyarlamalı ışıklılık dönüşümü uygulanmıştır. Elde edilen her bant grubu ayrık kosinüs dönüşümü ve sonrasında Huffman kodlama kullanılarak sıkıştırılmıştır. Önerilen ışıklılık dönüşümünün amacı, bir grup içindeki bant imgeleri arasındaki ışıklılık ve karşıtlık farklılıklarını azaltarak sıkıştırma performansının artırılmasını sağlamaktır. Deneysel sonuçlarda, Cuprite, Moffett Field, Jasper Ridge ve Pavia University hiperspektral görüntüleri üzerinde önerilen yöntem, ışıklılık dönüşümünün farklı versiyonları ile karşılaştırılmıştır. Karşılaştırma sinyal-gürültü oranı ve ortalama spektral açı uzaklığı gibi ölçütler kullanılarak yapılmıştır. Bunun yanında, sıkıştırılan verideki anomali ve hedef tespiti başarımları da karşılaştırılmıştır. Önerilen yöntemin, 3D-DCT sıkıştırma performansını özellikle düşük bit oranlarında ortalama %40 oranına kadar arttırdığı gösterilmiştir.

**Anahtar kelimeler:** Hiperspektral görüntüler, Sıkıştırma, Ayrık kosinüs dönüşümü, Işıklılık dönüşümü.

### Abstract

In recent years, hyperspectral imaging has become a very popular subject with its use in different application areas. Hyperspectral images that require high storage areas need to be compressed with high efficiency and quality. In this study, a novel method that uses automatic adaptive luminance transform and three-dimensional discrete cosine transform (3D-DCT) for lossy compression of hyperspectral images is proposed. Firstly, spectral bands in hyperspectral image are grouped and automatic adaptive luminance transform is performed as a pre-processing stage in the proposed method. Each group is compressed by using DCT and Huffman encoding. The aim of the proposed luminance transform is to increase compression performance by decreasing luminance and contrast differences between band images in a group. In the experimental results, the proposed method and different versions of luminance transform are compared on Cuprite, Moffett Field, Jasper Ridge and Pavia University hyperspectral images. Comparison is carried out using signal-to-noise ratio and average spectral distance metrics. Besides, anomaly and target detection performances are also compared for compressed images. The proposed method has been shown to increase compression performance of 3D-DCT up to an average of 40% rate, especially at low bit rates.

**Keywords:** Hyperspectral images, Compression, Discrete cosine transform, Luminance transform.

## 1 Giriş

Hiperspektral görüntüler, görüntüleme yapılacak olan bölgenin yaydığı veya yansıttığı enerjinin farklı dalga boylarında toplanması ile elde edilmektedir. Bu görüntüler, farklı dalga boylarındaki enerji spektrumlarını içermesi nedeniyle yüzlerce bant görüntüsünden meydana gelmekte, bu nedenle de çok sayıda yeni ve farklı alanlardaki uygulamalara olanak sağlamaktadır. Örneğin; atmosfere yayılan zararlı gazların tespit ve takip edilmesi [1],[2], değişim tespiti uygulamaları [3], toprak tipinin analizi [4], tarım ve orman izleme [5], hedef ve anomali tespiti [6],[7] ve askeri gözetim [8] hiperspektral görüntülerin bazı uygulama alanlarıdır. Ek olarak, hiperspektral görüntülerin sağlık sektöründe yapılan çalışmalarda da kullanıldığı görülmektedir [9],[10].

Hiperspektral görüntüler genellikle yeryüzünden uzak bir platform tarafından üretilerek bir haberleşme kanalı üzerinden yer istasyonuna gönderilmektedir. Bu büyük hacimli görüntülerin iletimi esnasında haberleşme ortamının bant genişliği açısından veya depolanması sırasında hafıza boyutu gereksinimi açısından sorunlar yaşanabilmektedir. Gerek iletim için ihtiyaç duyulan bant genişliğini düşürebilmek gerekse görüntünün yakalandığı ve depolandığı yerdeki hafıza gereksinimini azaltmak amacıyla literatürde hiperspektral görüntü sıkıştırma konusu ortaya çıkmıştır [11]. Bu alanda yapılan çalışmalarda genel olarak düşük bit oranlarında geri elde etme hatasının minimum olması amaçlanmaktadır.

Tek renkli ve RGB imgelerin sıkıştırılması için yüksek sıkıştırma oranlarına ulaşabilen JPEG [12] ve JPEG2000 [13] gibi dönüşüm temelli yaklaşımlar öne çıkmaktadır. Dönüşüm temelli yöntemler ile hiperspektral görüntüler üzerinde

\*Yazışılan yazar/Corresponding author

spektral ve uzamsal bilgiler açısından benzerlik azaltılmakta, dönüşüm sonrasında oluşan veri nicemlenmekte ve son aşamada entropi kodlama yapılarak sıkıştırma işlemi gerçekleştirilmektedir. Sıkıştırma kullanılan dönüşüm yöntemler incelendiğinde ise ayrık kosinüs dönüşümü (DCT) [14] ve ayrık dalgacık dönüşümü (DWT) [15],[16] temelli yaklaşımların oldukça popüler olduğu görülmektedir. Hiperspektral görüntülerin kayıplı bir şekilde sıkıştırılmasında da dönüşüm temelli bu yöntemlerin kullanılabilirliği [17]'de verilen çalışmada gösterilmiştir.

Literatürde yer alan çalışmalarda, hiperspektral görüntü sıkıştırma dalgacık dönüşümü temelli yöntemlerin sıklıkla kullanıldığı görülmektedir. Dalgacık dönüşümü öncesinde spektral eksende uygulanan Temel Bileşenler Analizi (PCA) kullanılarak verideki spektral benzerlik azaltılıp sonrasında sıkıştırma işlemi JPEG2000 algoritması kullanılarak gerçekleştirilmiştir [18]. İlâveten, video sıkıştırma kullanılan 3D-SPIHT [19]'de verilen çalışmada hiperspektral görüntülere uygulanmıştır. Bu yöntemde, dalgacık katsayıları en önemli bitteki bilgiden en düşük önemdeki bite doğru bit düzlemleri şeklinde sıralanarak iletilmektedir. Bu yöntemler literatürde sıfır ağaç temelli kodlama (zero-tree based coding) olarak geçmektedir. Ayrıca, literatürde yer alan sıfır blok temelli kodlama yapısına sahip 3D-SPECK [20] ve bit düzlemleri şeklinde yapılan kodlamayı kullanan TARP filtreleme [21] yöntemleri de hiperspektral görüntü sıkıştırma alanında kullanılmaktadır.

Hiperspektral görüntülerin genellikle bir uydu tarafından sağlandığı düşünüldüğünde, sıkıştırma işleminin bu uydu üzerinde yapılması gerekmektedir. Bu durumda, sıkıştırma sonrasında veri kalitesinin düşmemesinin yanı sıra uydu üzerinde çok güçlü donanım ve kaynaklar olmadığından, yöntemin düşük karmaşıklığa sahip olması da beklenmektedir. Literatürdeki çalışmalarda kullanılan sıkıştırma yapılarının yüksek işlemsel karmaşıklık içermesinden dolayı işlem sürelerinin oldukça uzun olduğu belirtilmektedir [22]-[24]. Örneğin, [22]'de yapılan çalışmada PCA dönüşümü ile DCT işlemsel karmaşıklık açısından karşılaştırılmıştır. Karşılaştırmada, PCA'nin hesabında kovaryans matrisi ve özdeğer belirleme adımlarından dolayı hesapsal karmaşıklığın DCT'ye göre oldukça büyük olduğu belirtilmiştir [22]. Ayrıca, PCA dönüşümü veri bağımlı olduğundan her veri için ayrıca hesaplanması gerekmektedir [22]. DCT dönüşümünde ise bunun aksine sabit ağırlıklar kullanarak DCT hesabı yapılmaktadır. Başka bir çalışmada, bir video kodlama standardı olan H264 yöntemi hiperspektral görüntülere uygulanmıştır. İlgili çalışma, Indian Pines hiperspektral görüntüsünü 1000 saniyenin üstünde sıkıştırılmaktadır [23]. Ek olarak, hiperspektral görüntülerin sıkıştırılmasında Tucker tensör dönüşümü kullanan yöntemler de popülerdir [24]. Buna rağmen, tensör ayrıştırma işlemi yüksek dereceli tekil değer ayrıştırma (HOSVD) yöntemiyle yapılmaktadır. Bu yöntemin hesabının da oldukça yüksek olduğu [24]'teki referansta belirtilmiştir. Ayrıca, bu dönüşümün de yine PCA dönüşümü gibi veri bağımlıdır. DCT dönüşümü ile sıkıştırmanın bir diğer avantajı da DCT katsayılar hesaplanıp nicemlendikten sonra oluşan katsayıların sıralanması için sabit bir tablo kullanmasıdır [25]. Oysa ki, dalgacık dönüşümü kullanan SPIHT yönteminde ağaç formundaki katsayıların büyüklüklerine göre arasındaki bağlantının belirlenerek kodlama yapılması için tekrarlı optimizasyon algoritmaları çalışmaktadır [25]. Son olarak, DCT dönüşümü paralel döngülerle kolayca paralelleştirilebilir. Tüm bu bilgilerin ışığında bu çalışmada

dönüşüm sonrası işlem karmaşıklığı daha yüksek olan diğer dönüşümü temelli yöntemler yerine, görece daha düşük işlem yüküne sahip üç-boyutlu DCT (3D-DCT) temelli bir yöntemin kullanılmasını uygun görülmüştür.

Literatürde 3D-DCT'nin kullanıldığı çalışmalarda da dalgacık dönüşümünün kullanıldığı çalışmalarda olduğu gibi ön işlem veya son işlem uygulanarak performans artışının amaçlandığı görülmektedir. [26]'da yapılan çalışmada hiperspektral görüntüler üzerinde öncelikle 3D-DCT uygulanmaktadır. Bu işlemin devamında, veri üzerinde destek vektör makineleri (DVM) uygulanarak sıkıştırma aşamasında hangi katsayıların daha önemli olduğu incelenmiştir. PCA ve dalgacık dönüşümünü kullanan yaklaşımlara alternatif olarak, [20]'de verilen çalışmada da PCA'nin DCT ile kullanımı ele alınmıştır. Yapılmış olan bir diğer çalışmada ise hiperspektral görüntülerin tensör ayrıştırması ve DCT'nin birlikte kullanıldığı hibrit bir yöntem ile sıkıştırılması önerilmektedir [23]. Literatürde yer alan diğer çalışmalarda yapılan iyileştirmelerin 3D-DCT işlemi sonrasındaki nicemleme veya entropi kodlama işlemleri sırasında gerçekleştirildiği de görülmektedir [25],[28]-[31]. Buradan hareketle, hem ayrık dalgacık dönüşümü hem de ayrık kosinüs dönüşümü temelli sıkıştırma yöntemleri incelendiğinde uygun bir ön işlem veya son işlem ile sıkıştırma performansının artırılabilirliği görülmektedir.

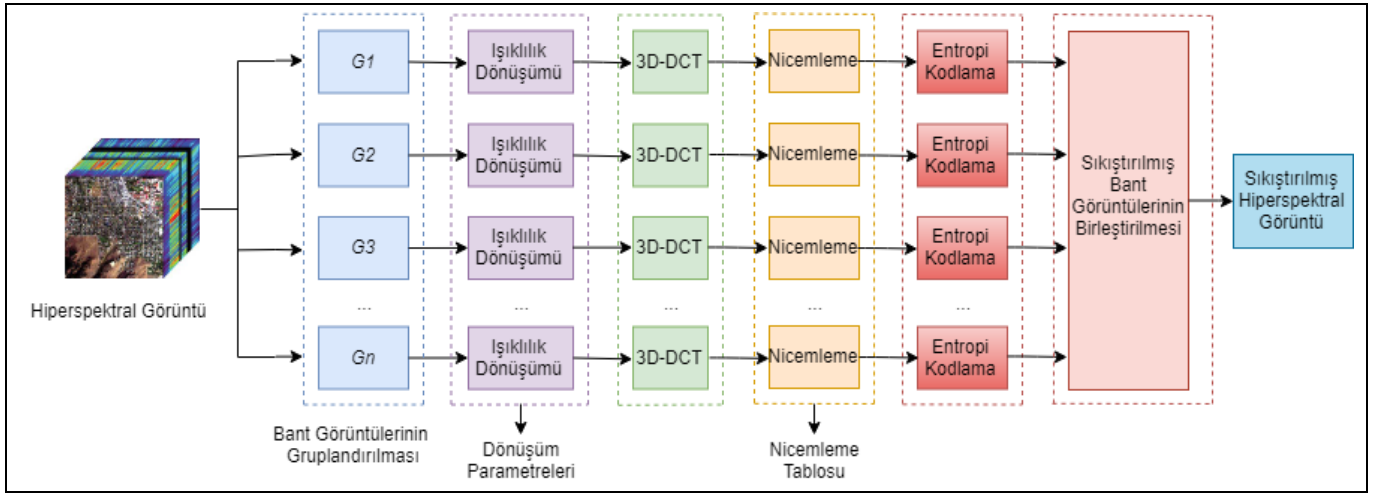
Bu çalışmada, hiperspektral görüntülerin sıkıştırılması öncesinde otomatik ışıklılık dönüşümü ile ön işlenmesi, sonrasında 3D-DCT temelli önerilmektedir. Kullanılan ışıklılık dönüşümü ile bantlar arasındaki parlaklık ve karışıklık farkı azaltılarak sıkıştırma performansının artırılması sağlanmaktadır. Çalışmada, 3D-DCT dönüşümü öncesinde ışıklılık dönüşümünün uyarlamalı ve otomatik olacak şekilde farklı kullanım durumları çeşitli performans ölçütleri ile incelenmektedir. Önerilen otomatik ışıklılık dönüşümü sayesinde uyarlamalı ışıklılık dönüşümdeki grupların dışarıdan gözle belirlenmesi sorunu çözülmektedir. Otomatik ışıklılık dönüşüm önce bant görüntülerini otomatik olarak gruplayıp sonrasında her bir grup içinde ışıklılık dönüşümü uygulanmaktadır.

## 2 Önerilen yöntem

Bu çalışmada, hiperspektral görüntülerin sıkıştırılması için önerilen 3D-DCT temelli yöntemle ait işlem adımları aşağıda sıralanmıştır.

1. Hiperspektral görüntü, bantlar arasındaki ilinti değeri dikkate alınarak benzer bant görüntüleri bir arada olacak şekilde bant gruplarına ayrılmaktadır,
2. Her bir bant grubu üzerinde 3D-DCT işlemi uygulanarak DCT katsayıları elde edilmektedir,
3. Katsayılar nicemlenip zig-zag tarama ile sıralanmaktadır.
4. Elde edilen nicemlenmiş DCT katsayıları entropi kodlanarak (Huffman) sıkıştırma işlemi gerçekleştirilmektedir,
5. Geri elde esnasında, sıkıştırma işlemi sırasında kullanılan entropi kodlama yapısı, nicemleme tabloları ve dönüşüm parametreleri kullanılarak tüm işlemlerin tersi gerçekleştirilmektedir.

Önerilen yöntemle ait sıkıştırma aşamalarını gösteren akış şeması Şekil 1'de verilmiştir.



Şekil 1. Önerilen 3D-DCT temelli yönteme ait akış şeması.  
Figure 1. Flowchart of the proposed 3D-DCT based method.

## 2.1 Bant görüntülerinin gruplandırılması

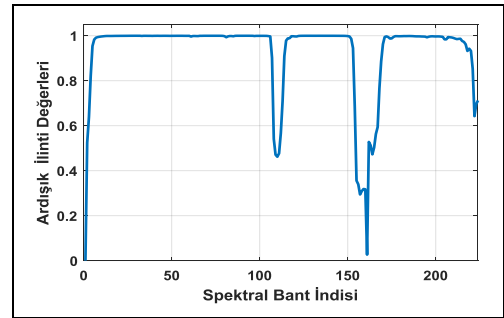
Hiperspektral görüntülerde spektral ve uzamsal ilinti olmak üzere iki farklı ilinti biçimi vardır. Uzamsal ilinti, aynı bant görüntüsü içerisinde birbirine yakın konumda olan piksellerin birbirine benzerliğini; spektral ilinti ise bant görüntülerinin birbirine olan benzerliğini göstermekte olup bantlar arası ilinti olarak da adlandırılmaktadır. Hiperspektral görüntülerde bantlar arası ilinti çok yüksek olduğundan, ilintisizleştirme işlemi ile yüksek sıkıştırma verimliliği sağlanmaktadır. Bu çalışmada ise, birbirileri ile yüksek ilintili komşu bant imgelerinin gruplandırılması yoluna gidilerek her gruba ayrı ön işlem ve sıkıştırmanın uygulanması ve bu sayede daha yüksek sıkıştırma verimliliğine erişilmesi hedeflenmiştir. Bantlar arasındaki ilinti hesabı Denklem (1)'de verilmektedir.

$$r(\mathbf{B}^i, \mathbf{B}^j) = \frac{\sum_{m=0}^{G-1} \sum_{n=0}^{Y-1} (\mathbf{B}_{m,n}^i - \bar{\mathbf{B}}^i)(\mathbf{B}_{m,n}^j - \bar{\mathbf{B}}^j)}{\sqrt{\sum_{m=0}^{G-1} \sum_{n=0}^{Y-1} (\mathbf{B}_{m,n}^i - \bar{\mathbf{B}}^i)^2} \sqrt{\sum_{m=0}^{G-1} \sum_{n=0}^{Y-1} (\mathbf{B}_{m,n}^j - \bar{\mathbf{B}}^j)^2}} \quad (1)$$

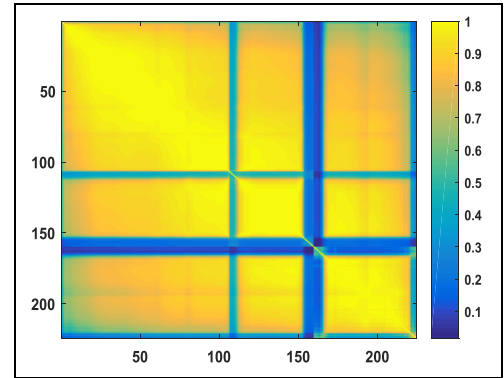
Denklem (1)'de  $\mathbf{B}^i$  terimi hiperspektral veri içerisindeki  $i$ . bant imgesini,  $\bar{\mathbf{B}}^i$  terimi bu imgenin ortalama değerini,  $G$  ve  $Y$  terimleri ise hiperspektral imgenin genişlik ve yükseklik değerlerini göstermektedir.

Bantlar arası ilintiyi bir örnek veri üzerinden yorumlayabilmek amacıyla, Cuprite verisinin ardışık bantlarının ilinti değerleri ve ilinti matrisi sırasıyla Şekil 2(a)'da ve Şekil 2(b)'de verilmektedir. Her iki şekil de incelendiğinde 5-104, 114-150 ve 168-220 indisli spektral bantlar arasında birbirleriyle olan karşılıklı ilintilerinin oldukça yüksek olduğu görülmektedir. Bazı bantlar arasında da ilinti değerlerinde önemli düşüşler olduğu ve diğer gruplar kadar birbirlerine benzemediği fark edilmektedir.

Bant imgelerinin birbirine benzer hale getirilmesi ile 3D-DCT sonrasında elde edilen katsayıların daha yoğun enerjiye sahip olması, dolayısıyla daha yüksek sıkıştırma verimliliği elde edilmesi mümkündür. Yakın zamanda yapılmış olan çalışmada 3D-DCT öncesinde ışıklılık dönüşümünün kullanılmasının sıkıştırma performansını arttırdığı gösterilmiştir [25],[28]-[31].



(a)



(b)

Şekil 2. Cuprite verisi için, (a): Ardışık bantların ilinti değerleri, (b): İlinti matrisi.

Figure 2. For Cuprite data, (a): Correlation values of sequential bands, (b): Correlation matrix.

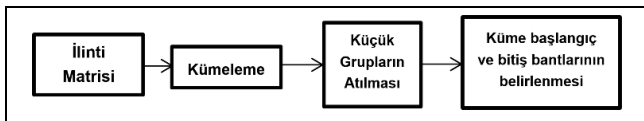
Literatürde 3D-DCT dönüşümü için  $8 \times 8 \times 8$ 'lik bloklar kullanılmaktadır. Işıklılık dönüşümü bu yapıya doğrudan entegre edildiğinde her 8'erli bant grubu içerisinde birbirinden bağımsız bir şekilde uygulanmaktadır [26]. Burada, ışıklılık dönüşümü uygulanacak görüntü bantlarının birbirine benzer olması ışıklılık dönüşümü sonrası elde edilen 3D-DCT katsayılarının daha kompakt olmasına yol açmaktadır. Bu sebeple, ışıklılık dönüşümünün ardışık 8 bant gibi sabit bir değer yerine farklı sayıda bant içerecek şekilde uyarlamalı

olarak gruplanması ile daha verimli bir kodlama şeması olmaktadır [27]. Önerilen yöntemin uyarlamalı ışıklık dönüşümü olarak nitelendirilmesinin nedeni; 3D-DCT ve standart ışıklık dönüşümünde kullanılan  $8 \times 8 \times 8$ 'lik blok yapısı yerine, bant görüntüleri arasındaki benzerlik ölçütü dikkate alınarak blok yapısının  $8 \times 8 \times P$  şeklinde benzer gruplara göre belirlenmesinden kaynaklanmaktadır.

Grupların belirlenmesinde kullanılacak yollardan birisi veriye ait ilinti matrisini hesaplamak ve sonrasında grupları gözle ayırmaktır. Örneğin; Şekil 2(a) incelendiğinde oluşan 7 gruba ait bant indisleri grupları "1-3, 4-107, 108-112, 113-153, 154-167, 168-221, 222-224" şeklinde olmaktadır. Bu şekilde ayrılan her gruba birbirinden bağımsız olarak ışıklık dönüşümü ve 3D-DCT dönüşümü uygulanabilmekte ve böylelikle veri daha verimli şekilde sıkıştırılabilmektedir. Uyarlamalı ışıklık dönüşümü uygulanırken, bir uzman tarafından ilinti matrisleri üzerinden belirlenen grupların başlangıç ve bitiş bant indislerine ihtiyaç duyulmaktadır.

Bu çalışmada, bant gruplandırma işleminin bir kümeleme algoritmasıyla otomatik hale getirilmesi ve performans değerlendirmesi üzerine yoğunlaşmıştır. Bant gruplandırma işleminde öncelikle her bir veriye ait ilinti matrisleri hesaplanmakta ve bantların birbiriyle olan benzerlik durumu üzerinden kümeleme yapılmaktadır. Burada, K-Ortalama gibi geleneksel yaklaşımlarda kümeleme sonrasında elde edilmesi istenilen küme sayısının daha önceden algoritmaya girdi olarak verilmesi gerekmektedir. Buna rağmen farklı hiperspektral görüntülerde oluşan ilinti matrisleri birbirinden tamamen farklı olmakta ve sabit bir küme sayısı girilmesi durumunda bant görüntüleri verimli bir şekilde gruplanmamaktadır. Burada verimsiz gruplama, herhangi bir spektral bantın birbiriyle daha ilintili olduğu halde daha ilintisiz olan başka bir küme etiketine atanması anlamına gelmektedir. Bu şekilde yapılan kümeleme sonrasında sıkıştırma işlemleri uygulandığından bu durum sıkıştırma performansını doğrudan etkilenmektedir. Örneğin, [32]'de aynı veri için farklı gruplama sonuçlarının etkisi incelendiğinde sıkıştırma performansındaki farklılıkların sıkıştırma sonuçlarında da farklılık yarattığı paylaşılmıştır. Bundan dolayı, bu çalışmada, bantların gruplanması için küme sayısını kendi kendine belirleyen bir hiyerarşik kümeleme yaklaşımı kullanılmaktadır.

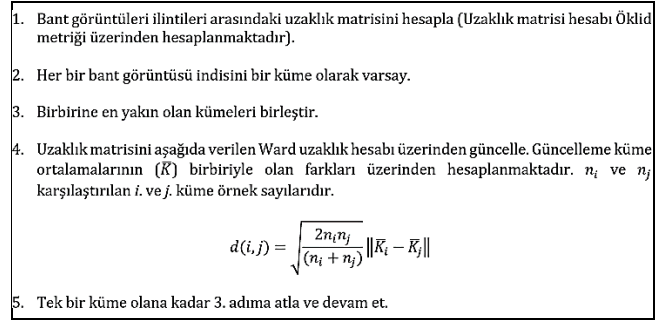
Bant görüntülerinin gruplanması için önerilen yöntem adımları ve çalışma akışı Şekil 3'te verilmiştir.



Şekil 3. Bantların gruplanması sürecine ait şema.

Figure 3. A scheme of band grouping process.

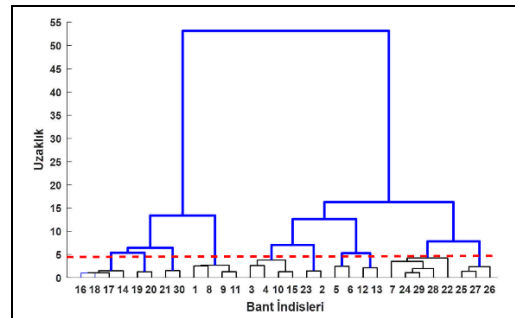
Yöntemin ilk adımında verideki her bant görüntüsünün birbiriyle olan ilinti değerleri hesaplanarak ilinti matrisi oluşturulur. Ardından ilinti matrisi toplayıcı hiyerarşik kümeleme (agglomerative hierarchical clustering) yöntemi [33],[34] ile kümelenebilir. Bu kümeleme yönteminde tüm örneklerin birbiriyle olan ilişkisi hiyerarşik bir ağaç yapısıyla karakterize edilmektedir. Bu sayede, veri içerisindeki örneklerin birbiriyle olan ilişkisi grafiksel olarak görülebilmekte ve örneklerin arasında uzaklık değeri belirlenerek küme sayısı bulunabilmektedir. Toplayıcı hiyerarşik kümelemeye ait akış Şekil 4'te verilmiştir.



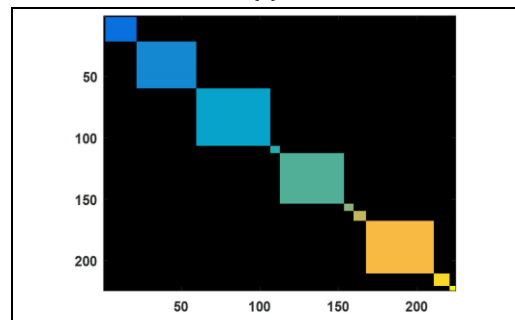
Şekil 4. Toplayıcı hiyerarşik kümeleme ile bant görüntülerinin kümelenebilmesi.

Figure 4. Segmentation of band images with agglomerative hierarchical clustering.

Hiyerarşik kümeleme sonrasında bir ağaç yapısı oluşmaktadır. Ağaç yapısında belirli uzaklığın altındaki kümeler birleştirilerek nihai kümeler belirlenmektedir. Bazen ilinti değeri çok düşük tek bir bant veya iki bant görüntüsü bir kümeye atanabilmektedir. Üçüncü adımda belirlenen tüm kümeler arasında toplam bant sayısı 3'ten az olan küme iptal edilerek ilintisi daha büyük olan komşu gruplara kaydırılmaktadır. Son olarak her bir kümenin başlangıç ve bitiş bant indisleri belirlenerek gruplama işlemi sonlandırılmaktadır. Bant gruplandırma işlemi Cuprite verisi için uygulandığında elde edilen ağaç yapısı (dendrogram) ve nihai kümeleme sonucu Şekil 5'te verilmiştir.



(a)



(b)

Şekil 5. Cuprite verisi için, (a): Ağaç çizit, (b): Kümeleme sonucu.

Figure 5. For Cuprite data, (a): Dendrogram, (b): Clustering result.

Önerilen yöntemin ilinti matrisini başarılı bir şekilde kümelediği görülmektedir. Otomatik gruplandırmanın başarımının sıkıştırmaya etkisi ve elle seçilen gruplar üzerinden yapılan sıkıştırma ile karşılaştırılması Deneysel Sonuçlar başlığı altında verilmektedir.

## 2.2 Işıklılık dönüşümü

Hiperspektral görüntüdeki spektral bantlar gruplandırıldıktan sonra her grup içerisinde ışıklılık dönüşümü gerçekleştirilmektedir. Grup içerisindeki diğer bant imgelerinin ortalama parlaklık ve karşıtlık değeri, ışıklılık dönüşümü ile referans imgenin ortalama parlaklık ve karşıtlık değerine çekilmektedir. Bu sayede, grup içindeki bant imgeleri birbirine daha benzer hale getirilmektedir. Bu durumda, 3D-DCT katsayıları daha kompakt hale getirilerek az sayıda katsayıyla korunmak istenen bilgi ifade edilebilmektedir. Bunun sonucu olarak sıkıştırma verimliliği artırılmaktadır [35],[36]. Burada grup görüntüleri için öncelikle referans bant görüntüsü seçilmektedir. Bu çalışmada referans bant görüntüsü seçimi, görüntü gruplarının ilk bant görüntüsü olacak şekilde yapılmıştır. Hiperspektral görüntülerin bant grupları üzerinde kullanılan ışıklılık dönüşümü ifadesi Denklem (2)'deki eşitliklerle verilmiştir. Bant görüntülerinin gruplandırılması ile elde edilen grup uzunluğu ( $P$ ) boyunca, bantlar arasında ışıklılık dönüşümü yapılmıştır. Denklem (2)'de  $\mathbf{B}^{ref}$  grubun referans bant görüntüsünü;  $\mathbf{B}^i$  grup içindeki  $i$ . bant görüntüsünü  $\alpha$  ve  $\beta$  sırası ile görüntüler arasındaki karşıtlık ve parlaklık farkını göstermektedir.  $\mathbf{E}$  terimi ise dönüşüm sonrasında oluşacak hata matrisini göstermektedir.

$$\mathbf{B}^i = \alpha \mathbf{B}^{ref} + \beta + \mathbf{E} \quad i = 2,3,4, \dots, P \quad (2)$$

En uygun ışıklılık dönüşümü parametreleri ( $\alpha^*, \beta^*$ ) Denklem (4) ve Denklem (5)'teki gibi dönüşüm sonrasında elde edilecek olan toplam karesel hatanın ( $ER$ ) minimize edilmesiyle bulunmaktadır.

$$ER = \sum \sum [\mathbf{B}^i - (\alpha \mathbf{B}^{ref} + \beta)]^2 \quad (3)$$

$$\arg \min_{\alpha, \beta} ER = (\alpha^*, \beta^*) \quad (4)$$

Toplam karesel hatayı minimum yapan parametreleri Denklem (5)'teki gibi toplam hatanın parametrelere göre ayrı ayrı kısmi türevlerinin alınıp sıfır değerine eşitlenmesiyle Denklem (6) ve Denklem (7)'deki gibi bulmaktadır.

$$\frac{\partial ER}{\partial \alpha} = 0 \text{ ve } \frac{\partial ER}{\partial \beta} = 0 \quad (5)$$

$$\alpha^* = \frac{(G \times Y)(\mathbf{B}^i, \mathbf{B}^{ref}) - \|\mathbf{B}^i\|_1 \|\mathbf{B}^{ref}\|_1}{(G \times Y)\|\mathbf{B}^i\|_2 \|\mathbf{B}^{ref}\|_2 - \|\mathbf{B}^{ref}\|_1^2} \quad (6)$$

$$\beta^* = \left( \|\mathbf{B}^i\|_1 - \alpha \|\mathbf{B}^{ref}\|_1 \right) / (G \times Y) \quad (7)$$

Bulunan parametreler kullanılarak bant görüntüleri referans bant görüntüsüne yaklaştırılmaktadır. Bu sayede, 3D-DCT katsayılarındaki enerji bütünlüğü artmakta ve daha az katsayı ile temsil edilebilmektedir. Bu da verinin daha verimli bir sıkıştırmasının yolunu açmaktadır.

## 2.3 Üç Boyutlu ayırık kosinüs dönüşümü ve sıkıştırma

3D-DCT öncesinde ışıklılık dönüşümü ile elde edilen bant grupları  $8 \times 8 \times P$ 'lik veri küplerine ayrılmaktadır. Burada,  $P$  terimi ilgili bant grubunun uzunluğunu göstermektedir. Bu veri küpleri  $\mathbf{I}$  sembolüyle ifade edildiği takdirde,  $\mathbf{I}(m, n, p)$  verinin ( $m, n, p$ ) konumundaki pikselin ışıklılık değerini belirtmektedir. Bu bilgiler üzerinden 3D-DCT dönüşümü Denklem (8)'deki eşitlikle tanımlanmaktadır.

$$\mathbf{S}(u, v, w) = \kappa(u, v, w) \sum_{m=0}^{G-1} \sum_{n=0}^{Y-1} \sum_{p=0}^{P-1} \mathbf{I}(m, n, p) C_{2M}^{(2m+1)u} C_{2N}^{(2n+1)v} C_{2P}^{(2p+1)w} \quad (8)$$

Burada, dönüşüm uygulanacak verinin genişlik ( $G$ ) ve yükseklik ( $Y$ ) değeri 8 olarak seçilmektedir. Denklem (8)'te belirtilen  $C_j^i$  ve  $\kappa(u, v, w)$  ifadesi aşağıdaki gibidir.

$$C_j^i = \cos\left(\frac{i\pi}{j}\right) \quad (9)$$

$$\kappa(u, v, w) = \sqrt{\frac{8}{G \times Y \times P}} E(k) \quad (10)$$

$E(k)$  ifadesi ise Denklem (11)'deki gibi tanımlanmaktadır.

$$E(k) = \begin{cases} \frac{1}{\sqrt{2}}; & u, v, w = 0 \\ 1; & \text{diğer} \end{cases} \quad (11)$$

3D-DCT tersinebilir bir dönüşümdür ve Denklem (12)'de verilen eşitlikle DCT dönüşüm katsayılarından veri geri elde edilebilir [28].

$$\mathbf{I}(m, n, p) = \sum_{u=0}^{G-1} \sum_{v=0}^{Y-1} \sum_{w=0}^{P-1} \kappa(u, v, w) \mathbf{S}(u, v, w) C_{2M}^{(2m+1)u} C_{2N}^{(2n+1)v} C_{2P}^{(2p+1)w} \quad (12)$$

Sıkıştırma işleminde öncelikle 3D-DCT katsayıları Denklem (8)'deki eşitlik kullanılarak elde edilir. Sonrasında elde edilen bu katsayıların nicemleme işlemiyle belirli seviyelere indirgenmesi gerekmektedir. Bu işlem için gerekli nicemleme tabloları [28]'deki çalışma referans alınarak elde edilmiştir. Nicemleme işlemi, Denklem (13)'teki gibi uygulanmakta olup, nicemleme matrisinin ( $\mathbf{Q}$ ) üretilmesi için Denklem (14)'deki eşitlik kullanılmaktadır.

$$\mathbf{S}_q(u, v, w) = \text{round}\left(\frac{\mathbf{S}(u, v, w)}{\mathbf{Q}(u, v, w)}\right) \quad (13)$$

$$\mathbf{Q}(u, v, w) = \text{round}(u + v + kw + 3) \quad (14)$$

$$u, v = 0, 1, 2, 3, \dots, 7; \quad w = 0, 1, 2, 3, \dots, P - 1 \quad (15)$$

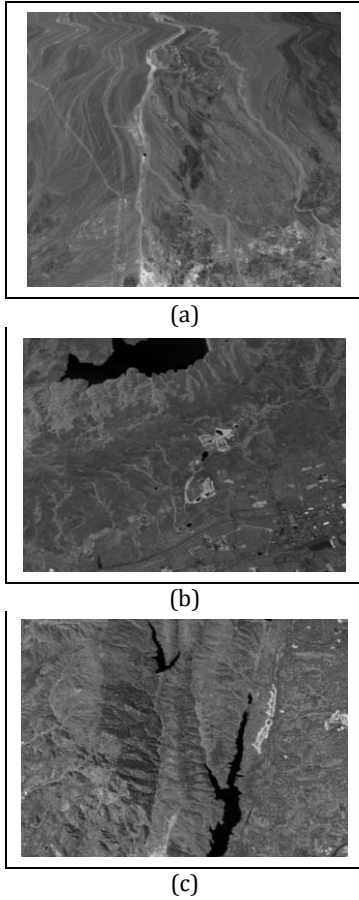
Burada,  $\mathbf{S}_q$  terimi nicemlenmiş görüntü bloğunu göstermektedir. Nicemleme işlemi esnasında kullanılan ve Denklem (14) ile elde edilen nicemleme matrisi için  $k$  nicemleme parametresi, bantlar arasındaki benzerliğin yüksek olmasından dolayı  $k=8$  olarak belirlenmiştir [28]. Nicemleme sonrasında 3D-DCT katsayıları küpün ilk elemanından itibaren dışa doğru zig-zag bir biçimde taranmaktadır. Zig-zag sıralama adımıyla yapılan işlemler [28]'de verilen tablolar referans alınarak gerçekleştirilmiştir. Son olarak, sembol kodlama ile nicemlenmiş DCT katsayıları sıkıştırılmıştır.

Sembol kodlama adımı DC ve AC bileşenler JPEG algoritmasında olduğu gibi Huffman kodlamayla gerçekleştirilmektedir. Tarama sonrasında elde edilen AC ve DC bileşenler kodlanırken farklı prosedürler işletilmektedir. DC bileşenlerin kodlanmasında DC değerler arka arkaya eklenerek farksal kodlama (DPCM) uygulanmaktadır. AC bileşenlerde ise AC katsayılar arka arkaya eklenerek sıfırdan küçük değerler için pozitif tek sayı, sıfıra eşit ve ondan büyük değerler için ise pozitif çift sayıya olacak şekilde öteleme yapılmaktadır. Ardından elde edilen bileşen vektörleri [37]'daki Huffman kodlayıcıyla sıkıştırılmaktadır. Huffman kodlama esnasında öncelikle kod kitabı üretilmektedir. Sonrasında üretilen kod kitabına göre her bir sembol, ilgili kod sözcüğüyle kodlanarak kaydedilmektedir.

Sıkıştırılan görüntünün geri elde edilmesi için sembol kodlama, nicemleme ve ışıklılık dönüşümü adımlarında yapılanların ters işlemleri gerçekleştirilmiştir; elde edilen bant grupları birleştirilmiştir.

### 3 Deneysel sonuçlar

Bu çalışmada, yapılan deneylerde Cuprite, Moffett Field, Jasper Ridge ve Pavia University hiperspektral görüntüleri kullanılmıştır. Kullanılan görüntülerden Cuprite, Moffett Field ve Jasper Ridge 512×512 piksel uzamsal çözünürlüğünde olup 224 spektral banttan oluşmaktadır. Pavia University verisi ise 610×340 uzamsal çözünürlüğündedir ve 103 spektral banttan oluşmaktadır. Bu görüntüler genellikle literatürde spektral ayrıştırma ile katışım analizi problemlerinde materyallere ait katışım oranlarının bulunmasında ve sahne sınıflandırmasında kullanılmaktadır. Ayrıca, sıkıştırma yöntemlerin karşılaştırılmasında da sıklıkla kullanılmaktadır. Çalışmada kullanılan Cuprite, Moffett Field ve Jasper Ridge isimli hiperspektral görüntülerin 50. bant görselleri Şekil 6'da verilmektedir.



Şekil 6. Hiperspektral görüntülerin 50. bant görüntüleri, (a): Cuprite, (b): Moffett field, (c): Jasper ridge.

Figure 6. 50<sup>th</sup> band images of hyperspectral images, (a): Cuprite, (b): Moffett field, (c): Jasper ridge.

#### 3.1 Sıkıştırma performanslarının değerlendirilmesi

Önerilen yöntemde 3D-DCT işlemi öncesinde ışıklılık dönüşümü, farklı veriler üzerinde benzer bant görüntülerinin gruplandırılması (uyarlamalı ve otomatik olmak üzere) ile birlikte kullanılmıştır. Önerilen yöntemin sıkıştırma performansı, işaret-gürültü oranı (Signal-to-Noise Ratio, SNR), ortalama spektral açı benzerliği (Mean Spectral Angle, MSA), ortalama spektral ilinti (Mean Spectral Correlation, MSC) ve karşılıklı bilgi (Mutual Information, MI) gibi kalite ve benzerlik metrikleri kullanılarak değerlendirilmiştir. Kullanılan metriklerin tanımları sırasıyla Denklem (16), Denklem (17),

Denklem (18) ve Denklem (19)'da verilmiştir. Denklem (18)'de *Corr* terimi ilinti katsayısını, Denklem (19)'da ise  $p(x, y)$  terimi  $x$  ve  $y$  sembollerinin birleşik olasılıklarını (joint probability) göstermektedir.

$$SNR = 10 \log_{10} \frac{\sum_{m=0}^{G-1} \sum_{n=0}^{Y-1} \|\mathbf{I}(m, n)\|^2}{\sum_{m=0}^{G-1} \sum_{n=0}^{Y-1} \|\mathbf{I}(m, n) - \mathbf{I}^R(m, n)\|^2} \quad (16)$$

$$MSA = \frac{1}{G \times Y} \sum_{m=0}^{G-1} \sum_{n=0}^{Y-1} SA_{m,n} \quad (17)$$

$$SA_{m,n} = \cos^{-1} \left( \frac{\langle \mathbf{I}(m, n), \mathbf{I}^R(m, n) \rangle}{\|\mathbf{I}(m, n)\| \|\mathbf{I}^R(m, n)\|} \right)$$

$$MSC = \frac{1}{G \times Y} \sum_{m=0}^{G-1} \sum_{n=0}^{Y-1} Corr(\mathbf{I}(m, n), \mathbf{I}^R(m, n)) \quad (18)$$

$$MI = \sum_{x \in \mathbf{I}, y \in \mathbf{I}^R} p(x, y) \log \left( \frac{p(x, y)}{p(x)p(y)} \right) \quad (19)$$

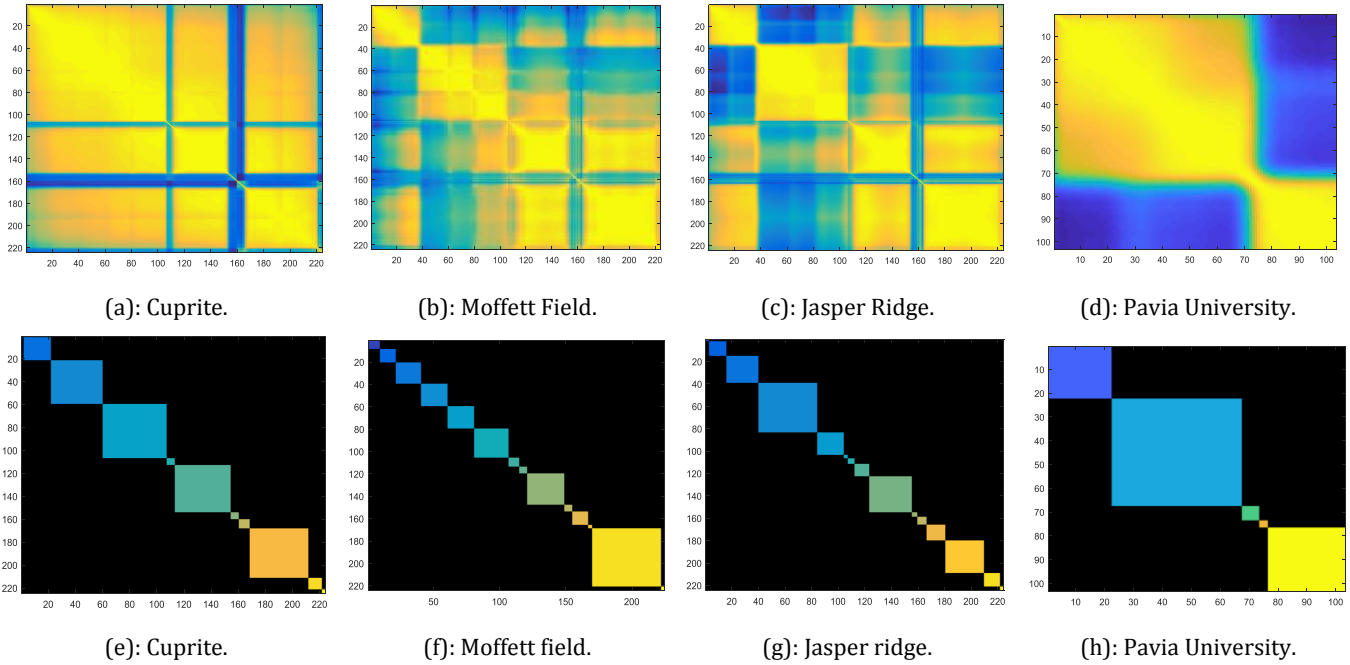
Karşılaştırma sırasında dört farklı durum değerlendirilmiştir. Bunlar, sadece 3D-DCT yöntemi, standart ışıklılık dönüşümü ile 3D-DCT (ID+3D-DCT), uyarlamalı ışıklılık dönüşümü ile 3D-DCT (UID+3D-DCT) ve otomatik uyarlamalı ışıklılık dönüşümü ile 3D-DCT'nin (OID+3D-DCT) kullanıldığı durumlardır. Otomatik nitelmesi ile yöntemin kullanıcıdan herhangi bir giriş gerek duymadan gruplama yapabilme özelliği vurgulanmaktadır. Benzer bant görüntülerinin, bantlar arasındaki ilintiye göre gruplanmasında kullanılan ilinti matrislerinin görseli ve önerilen bant gruplama yaklaşımı sonuçları Şekil 7'de gösterilmektedir. Hiperspektral görüntüleri ait ilinti matrislerinin görsellerinde sarı ile belirtilen bölgeler bantlar arası ilintinin yüksek; mavi ile gösterilen bölgeler ise bantlar arasındaki ilintinin düşük olduğunu göstermektedir. Buna göre yapılan gruplandırma işleminde Cuprite verisi için "1-3, 4-107, 108-112, 113-154, 155-168, 169-221, 222-224", Jasper Ridge ve Moffett Field verileri için "1-3, 4-38, 39-104, 105-112, 113-153, 154-165, 166-221, 222-224" bant aralıkları görseller üzerinden elle seçilmiştir. Elle belirlenen aralıklar ile otomatik gruplama sonuçları karşılaştırıldığında küme sayılarının aynı olmadığı görülmektedir. Aslında elle aynı grup olarak görülen bölgeler, otomatik gruplama yöntemi tarafından daha fazla gruba bölünmüştür. Bu durum sıkıştırma performanslarında bazı farklılıklara sebep olmaktadır. Bant gruplama için farklı kümeleme yaklaşımları kullanılmış olmasına karşın bu yöntemlerin sonuçları, elle seçmeye göre daha düşük sonuçlar verdiğinden, bu makalede paylaşılmamıştır.

Tablo 1'de tüm verilerin farklı yöntemlerle 0.1 ve 0.5 bit oranında (bpp) sıkıştırılması sonrasında ölçülen SNR değerleri sayısal değerler halinde paylaşılmaktadır. Yöntem karşılaştırılmasına literatürde en sık kullanılan JPEG2000 [13] ve PCA+JPEG2000 [38] de dahil edilmiştir. Bu tabloya göre önerilen yöntemde ait SNR değerlerinin 3D-DCT, ID+3D-DCT ve JPEG2000 yöntemlerinden daha yüksek olduğu görülmektedir. Bunun yanında, PCA+JPEG2000 yöntemine ait sonuçlar tüm yöntemlerin üstüne çıkmaktadır. Tablo 2'de ise sinyal-gürültü oranı sonuçları, bit oranları ve sıkıştırma oranları (CR) tüm verilerde ortalamaları alınarak paylaşılmıştır. Burada, genel performansta da OID+3D-DCT yöntemine ait SNR değerlerinin düşük bit oranlarında PCA+JPEG2000 haricindeki yöntemlere göre oldukça yüksek olduğu görülmektedir.

Örneğin, 0.1 bpp bit oranında Cuprite verisi için elde edilen sonuçlar incelendiğinde otomatik uyarlamalı ışıklık dönüşümünün uygulanması sadece 3D-DCT'ye göre 10 dB bir fark yaratmıştır. Ayrıca, Tablo 1'deki sonuçlara paralel olarak Tablo 2'de de PCA+JPEG2000 yöntemi en yüksek SNR değerlerine sahiptir. Buna rağmen, işlemsel karmaşıklığı oldukça yüksek olan bu yöntemin uydu platformları üzerindeki gömülü sistemlerde koşturulması gerek işlem süresi gerekse kullanılan kaynaklar bakımından verimsiz olabilmektedir. Bu çalışmanın temel amacı, PCA+JPEG2000 yöntemine göre işlemsel karmaşıklığı oldukça düşük 3D-DCT yapılarının performansının artırılmasıdır. Bir sonraki bölümde konu ayrıntılı bir şekilde ele alınmıştır. Bu sebeple, bu bölümde paylaşılan diğer sonuçlarda sadece düşük işlem yüklü yöntemlere ait bilgiler verilmiştir.

Farklı hiperspektral görüntüler üzerinde uygulanan uyarlamalı ve otomatik ışıklık dönüşümü sonucu elde edilen SNR, MSA, MSC ve MI sonuçları sırasıyla Şekil 8, Şekil 9, Şekil 10 ve

Şekil 11'de ayrıntılarıyla gösterilmektedir. Karşılaştırmalı değerlendirme amacıyla literatürde sık kullanılan JPEG2000 yöntemine ait sonuçlar da grafiklere eklenmiştir. Elde edilen sonuçlar incelendiğinde, JPEG2000 yöntemine ait metriklerin tüm 3D-DCT versiyonlarından daha düşük sıkıştırma kalitesine sahip olduğu gözlemlenmiştir. Ayrıca, hiperspektral görüntülerin sıkıştırılmasında ön işlem olarak uygulanan ışıklık dönüşümünün bantlar arasındaki ilintiyi dikkate alarak otomatik bir şekilde yapılması ile üç metrik açısından da mevcut yaklaşımlara kıyasla iyileştirme sağlandığı görülmektedir. UID+3D-DCT ve OID+3D-DCT'nin özellikle düşük bit oranlarında sıkıştırma performansını önemli derecede arttırdığı gözlemlenmiştir. SNR, MI ve MSC sonuçları incelendiğinde önerilen yöntem ile elde edilen sonuçların en yüksek değerleri verdiği görülmüştür. Aynı şekilde MSA sonuçları incelendiğinde de önerilen yöntemin en düşük (en iyi) sonuçları verdiği görülmektedir.



Şekil 7. Hiperspektral görüntülerin bantlar arası ilinti matrisleri görseli - önerilen bant gruplama yöntemi sonuçları.

Figure 7. Images of interband correlation matrices of hyperspectral images – the result of proposed band grouping method.

Tablo 1. Farklı yöntemlere ve veri kümelerine ait sinyal-gürültü oranı sonuçları (SNR-dB).

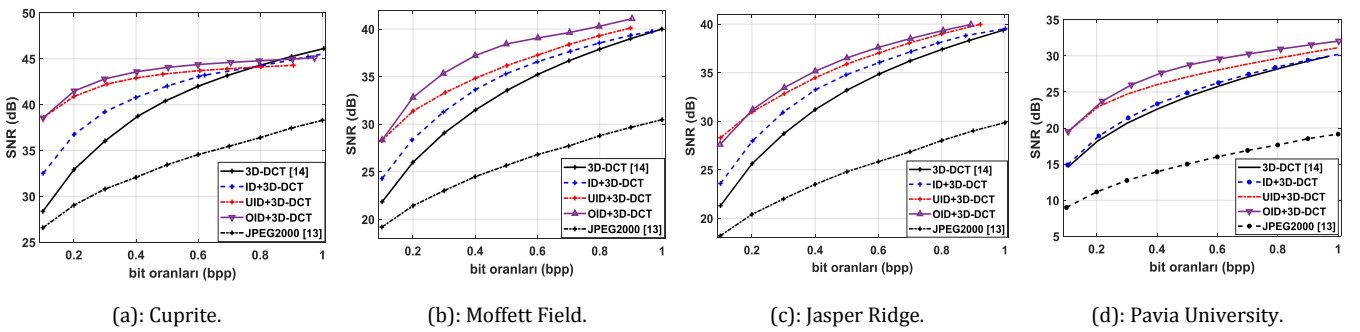
Table 1. Signal-noise ratio results for different methods and data sets (SNR-dB).

Yöntemler / bit ve sıkıştırma oranları	Cuprite		Moffet Field		Jasper Ridge		Pavia University	
	0.1 bpp	0.5 bpp	0.1 bpp	0.5 bpp	0.1 bpp	0.5 bpp	0.1 bpp	0.5 bpp
JPEG2000 [13]	26.57	33.44	19.17	25.66	18.20	24.80	9.00	15.01
PCA+JPEG2000 [38]	40.04	50.64	33.76	46.18	31.02	44.28	23.41	31.92
3D-DCT	28.36	40.40	21.84	33.56	21.30	33.19	14.91	24.30
ID+3D-DCT	32.51	42.02	24.28	35.28	23.59	34.77	14.91	24.88
UID+3D-DCT	38.61	43.32	28.28	36.16	28.30	35.89	19.37	27.10
OID+3D-DCT	38.55	44.07	28.33	38.42	27.61	36.51	19.51	28.76

Tablo 2. Farklı yöntemlere ait sonuçların ortalama sinyal-gürültü oranı açısından değerlendirilmesi (SNR-dB).

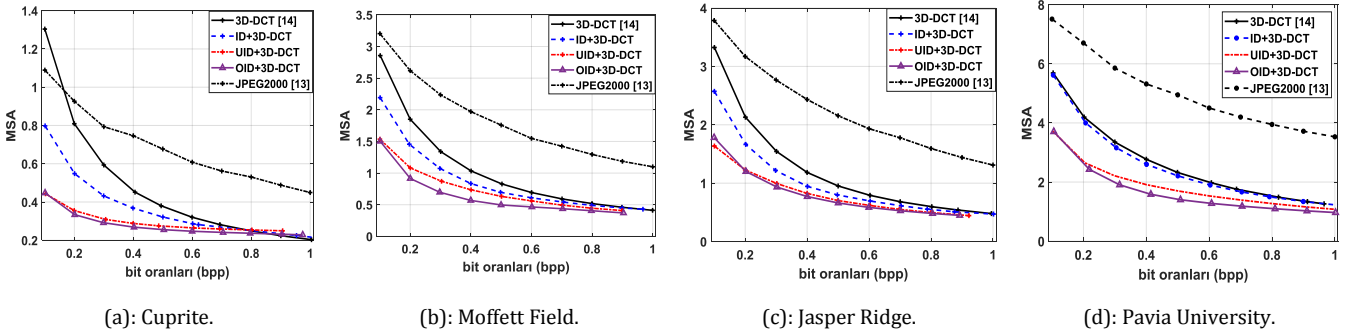
Table 2. Evaluation of various methods results in terms of average signal-noise ratio (SNR-dB).

Yöntemler / bit ve sıkıştırma oranları	0.1 bpp (CR=160)	0.3 bpp (CR=53)	0.5 bpp (CR=32)	0.7 bpp (CR=23)	1.0 bpp (CR=16)
JPEG2000 [13]	21.83	26.09	28.61	32.94	33.89
PCA+JPEG2000 [38]	35.43	43.14	46.66	48.35	49.78
3D-DCT	23.99	31.52	35.90	38.76	41.79
ID+3D-DCT	27.61	34.41	37.52	39.10	41.48
UID+3D-DCT	33.37	37.23	38.75	39.84	41.37
OID+3D-DCT	33.27	37.97	39.74	40.63	41.12



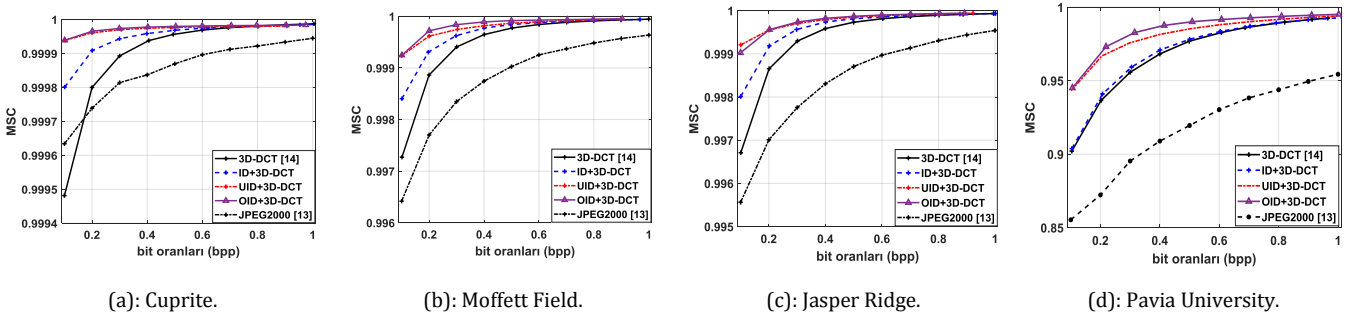
Şekil 8. Farklı bit oranları için hiperspektral sıkıştırma yöntemlerinin SNR sonuçları.

Figure 8. SNR results of hyperspectral compression methods for different bit-rates.



Şekil 9. Farklı bit oranları için hiperspektral sıkıştırma yöntemlerinin MSA sonuçları.

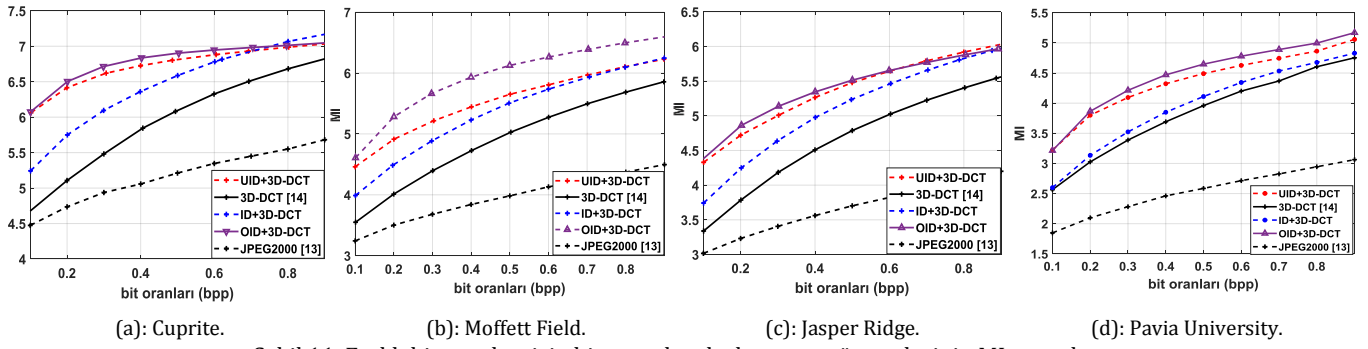
Şekil 9. MSA results of hyperspectral compression methods for different bit-rates.



Şekil 10. Farklı bit oranları için hiperspektral sıkıştırma yöntemlerinin MSC sonuçları.

Şekil 10. MSC results of hyperspectral compression methods for different bit-rates.





Şekil 11. Farklı bit oranları için hiperspektral sıkıştırma yöntemlerinin MI sonuçları.

Figure 11. MI results of hyperspectral compression methods for different bit-rates.

### 3.2 Yöntemlerin çalışma sürelerinin incelenmesi

Yöntemlerin çalışma süreleri, Cuprite verisi üzerinde 0.1 bpp bit oranında gerçekleştirilen sıkıştırma ile elde edilen süreler üzerinden karşılaştırılmıştır. Bu bit oranında yapılan sıkıştırma mevcut veriler için 160 katlık sıkıştırma oranına denk gelmektedir. 3D-DCT temelli tüm yöntemlerin kodlama işlemleri MATLAB® programı üzerinde benzer optimizasyon seviyesinde ve MEX kodu halinde kodlanarak yapılmıştır. JPEG2000 ve PCA+JPEG2000 yöntemlerinin kodlanması için QccPack ve ticari bir sıkıştırma yazılımı olan Kakadu enkoderi kullanılmıştır. Ölçülen süreler 2.2 GHz hızındaki Intel Core i5-5200U işlemcili bir bilgisayar üzerinden elde edilmiştir. Elde edilen çalışma süreleri Tablo 3'te gösterilmektedir.

Tablo 3. Farklı yöntemlerin işlem süresi açısından değerlendirilmesi (sn).

Table 3. Evaluation of different methods in terms of computation time (s).

	JPEG2K [13]	PCA+ JPEG2K [38]	3D-DCT	ID+ 3DDCT	UID+ 3DDCT	OID+ 3DDCT	OID+ 3DDCT (AF=2)	OID+ 3DDCT (AF=3)
Çalışma Süresi	14.2	44.4	12.7	16.2	92.3	93.5	34.1	29.9

Yöntemlerin çalışma süreleri incelendiğinde 3D-DCT ile ID+3D-DCT arasında 3.5 s'lik ışıklılık dönüşümü farkı görülmektedir. Uyarlamalı ışıklılık dönüşümünde ve otomatik ışıklılık dönüşümünde ise çalışma sürelerinin diğerlerine oranla daha yüksek çıkmasının nedeni, bantlar arasındaki ilinti değerlerinin hesaplanmasındaki geçen süreden kaynaklanmaktadır. Bu işlem her iki yöntem için de yaklaşık 69 s sürmektedir. Bu sebeple, sadece ilinti hesabındaki sürenin düşürülmesi yöntemin genel çalışma süresini doğrudan azaltacaktır. Buradan yola çıkılarak, bant görüntülerinin karşılıklı ilintilerinin hesaplanması öncesinde bant görüntüleri her iki uzamsal ekseninde de AF oranında alt örneklenmiştir. Yani oluşacak ilinti matrisi verinin alt örneklenmiş hali üzerinden hesaplanmaktadır. AF=2 olduğunda her iki ekseninde de boyutlar yarıya düştüğünden hızlanmanın etkisi karesel olacaktır. Bununla birlikte, AF değeri 4 ve üstünde bir değer olarak seçildiğinde gruplama sırasında farklılıklar oluşmaya başlamaktadır. Bu sebeple, çalışmada AF=2 ve AF=3 iken elde edilen süreler paylaşılmıştır. Sürelere bakıldığında alt örnekleme OID+3D-DCT yöntemini en az 2.5 kat hızlandırmaktadır. Bunun yanında, önerilen yöntemin çalışma süresi PCA+JPEG2000 ile kıyaslandığında AF=3 için daha kısa sürede sonuçların elde edilebildiği görülmektedir. Burada, PCA+JPEG2000 yöntemi ticari bir yazılım olan Kakadu

enkoderini kullandığından karşılaştırma aynı optimizasyon seviyesinde olmamaktadır. Adil bir karşılaştırma yapabilmek amacıyla değerlendirmeler Tablo 4'teki gibi işlemsel karmaşıklıklar üzerinden yapılmıştır.

Tablo 4. İşlemsel karmaşıklıklarının karşılaştırılması.

Table 4. Comparison of computational complexities.

	İşlemsel Karmaşıklık
PCA	$O(GYL^2) + O(L^3) + O(GYL^2)$
3DDCT	$O(3GYL \log_2 8)$
ID+ 3DDCT	$O(GY(L/8) \times 7) + O(3GYL \log_2 8)$
OID+ 3DDCT	$O(GYL^2/2) + \sum_{i=1}^{KS} O(GY(P_i - 1)/2) + O(GYP_i \log_2 8^2 P_i)$
OID+ 3DDCT (AF)	$O(GYL^2/(2AF^2)) + \sum_{i=1}^{KS} (O(GY(P_i - 1)/2) + O(GYP_i \log_2 8^2 P_i))$

Tablo 4'te  $L$  toplam bant sayısını ve  $KS$  toplam küme sayısını göstermektedir. 1 boyutlu DCT işlemi  $O(n \log_2 n)$  işlemsel karmaşıklığa sahiptir. Tüm hiperspektral görüntüye 3D-DCT dönüşümü uygulandığında toplam işlemsel karmaşıklık Tablo 4'teki gibi olmaktadır. Işıklılık dönüşümü her 8 bant grubu için parametre hesabı yaptığından oldukça az bir işlemsel karmaşıklık eklemektedir. Bu durum, Tablo 3'ten de görülebilmektedir. Önerilen yöntem için önemli işlem yükü, ilinti matrisinin eldesidir. Bu işlem için toplam  $O(GYL^2/2)$  karmaşıklık gerekmektedir. Ancak bu durum alt-örnekleme faktörünün de kullanılmasıyla  $O(GYL^2/(2AF^2))$  değerlerine kadar düşürülmektedir. Ancak, PCA dönüşümü oldukça yüksek bir işlemsel karmaşıklığa sahip olmaktadır [18].

Tablo 4'teki değerlerin daha iyi yorumlanabilmesi için işlem sırasında kullanılan toplam çarpma sayısı ve sayıların 3D-DCT çarpma sayısına oranları Tablo 5'te paylaşılmıştır.

Tablo 5. Toplam çarpma sayısı ve 3D-DCT'ye göre oranları.

Table 5. Total number of multiplication and ratios by 3D-DCT.

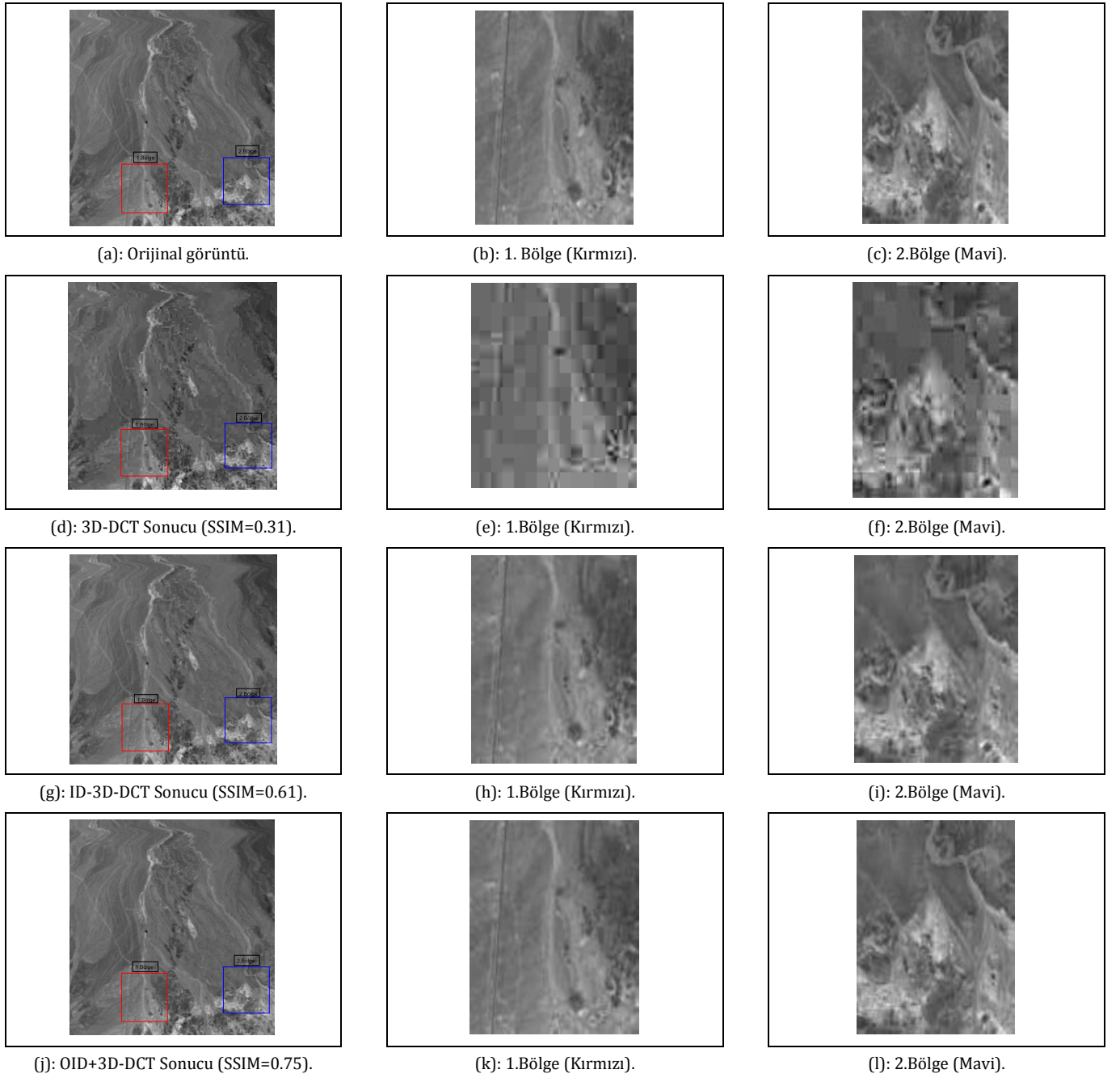
	Toplam Çarpma Sayısı	Toplam Çarpma Sayısı/ (3DDCT Çarpma Sayısı)
PCA	$2.63 \times 10^{10}$	49.80
3D-DCT	$5.28 \times 10^8$	1.00
ID+ 3D-DCT	$7.34 \times 10^8$	1.10
OID+ 3D-DCT	$7.27 \times 10^9$	13.76
OID+ 3D-DCT (AF=3)	$1.43 \times 10^9$	2.71

Bu hesap yapılırken Cuprite verisindeki bilgiler dikkate alınmıştır. Buna göre, ID uygulamak 3D-DCT yaklaşımını sadece %10 daha fazla işlem gerektirmektedir. PCA işlemi ise 3D-DCT ye göre 50 kat daha az çarpma işlemine sahip olduğu görülmektedir. Önerilen yöntem ise AF=3 olduğu durumda PCA'ya göre yaklaşık 20 kat daha hızlı olmaktadır. PCA+JPEG2000 yöntemi dönüşümden sonra JPEG2000 işlemi de uyguladığından bunun da karmaşıklığını göz önünde bulundurarak 3D-DCT temelli yöntemlerin oldukça hızlı çalıştığı söylenmektedir [18],[25]. Tüm bu bilgiler ışığında, dönüşüm işlemi için karmaşıklığı daha yüksek olan PCA gibi yöntemler yerine, daha düşük işlem yüküne sahip 3D-DCT

temelli bir yöntem üzerinde çalışmalar gerçekleştirilmiştir. Toplamda yaklaşık 2.4 katlık bir işlem süresi artışı ile 3D-DCT performansı önemli ölçüde artırılmıştır.

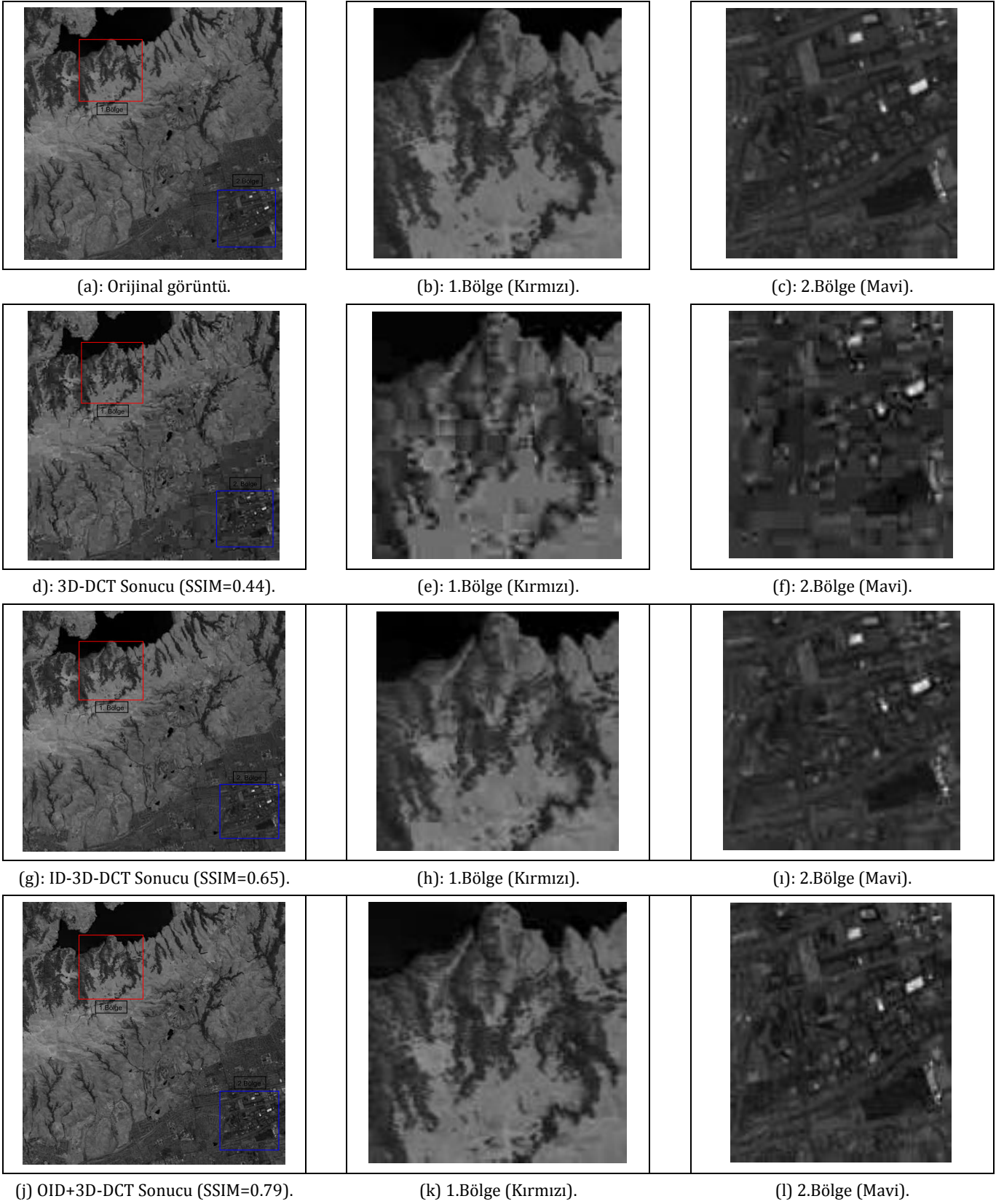
### 3.3 Görüntü görsel kalitesinde meydana gelen değişimlerin incelenmesi

Sıkıştırma işlemi sonrasında hiperspektral imgelerdeki bozunumlar görsel olarak incelenmiştir. Görüntülerde meydana gelen bozunumlar hiperspektral imgelerin 0.1 bpp elde edilen sıkıştırılmış görsellerinin 150. bantları üzerinden Şekil 12, Şekil 13 ve Şekil 14'te gösterilmektedir.



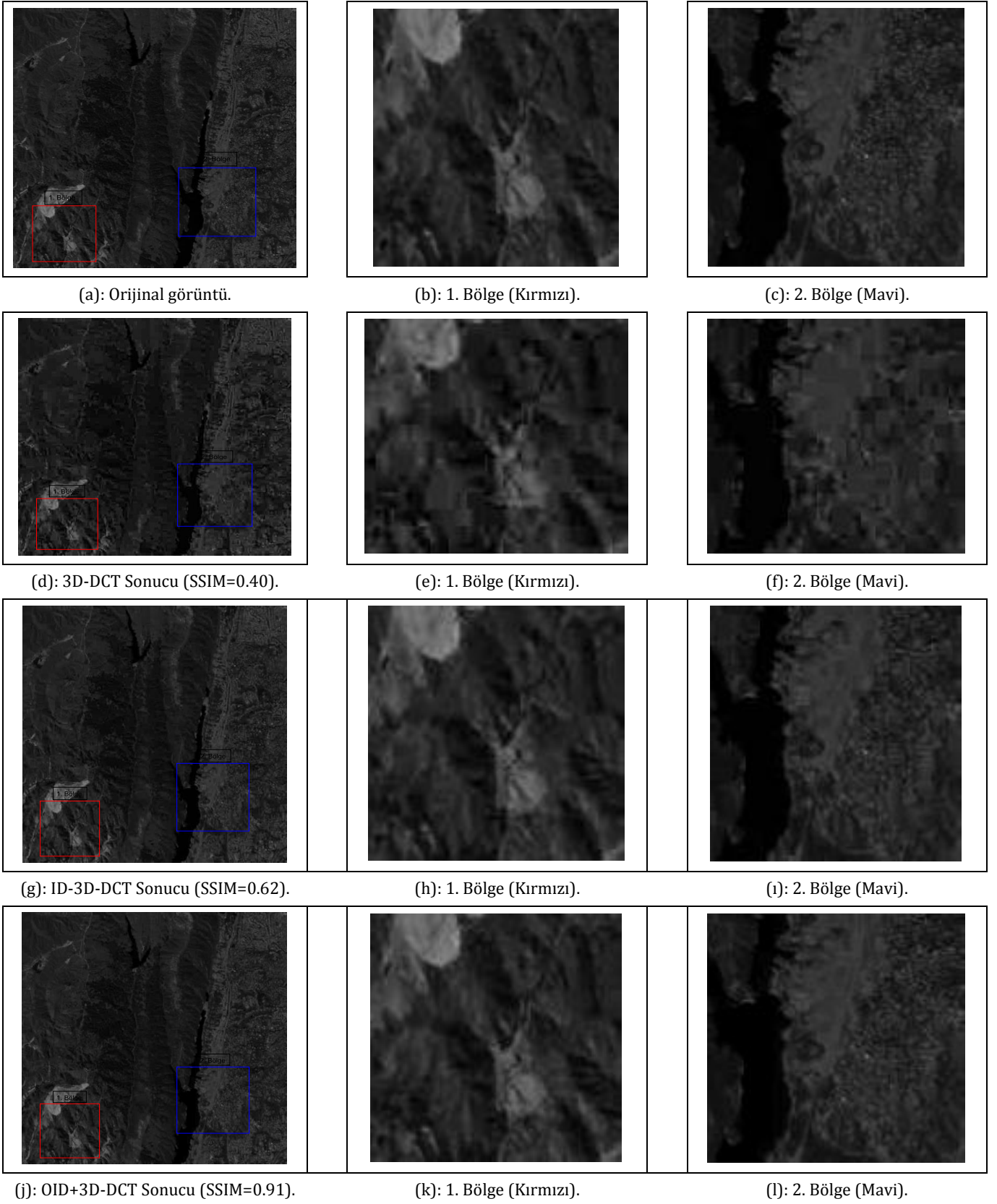
Şekil 12. Cuprite hiperspektral görüntüsünün farklı yöntemler ile sıkıştırılmasının görsel sonuçları.

Figure 12. Visual results of compression by different methods for Cuprite hyperspectral image.



Şekil 13. Moffett Field hiperspektral görüntüsünün farklı yöntemler ile sıkıştırılmasının görsel sonuçları.

Figure 13. Visual results of compression by different methods for Moffett Field hyperspectral image.



Şekil 14. Jasper Ridge hiperspektral görüntüsünün farklı yöntemler ile sıkıştırılmasının görsel sonuçları.  
Figure 14. Visual results of compression by different methods for Jasper Ridge hyperspectral image.

Görüntü kalitesinde meydana gelen bozunumları daha rahat görebilmek adına aynı görseller üzerinden seçilen farklı iki bölgenin yakınlaştırılmış görselleri de incelenmiş ve aynı şekiller üzerinde gösterilmiştir. Ayrıca, görsellerdeki bozunumun ölçülmesi için yapısal benzerlik indisi (SSIM) [39] değerleri her bir imge için hesaplanmış ve görsellerin altında parantez içinde paylaşılmıştır. Denklem (20)'de eşitliği verilen SSIM metriği, iltinti ( $c$ ), ışıklılık ( $L$ ) ve kontrast ( $k$ ) fonksiyonlarının üstel ağırlıklı çarpımlarına eşittir. SSIM değeri yüksek olan görüntüler orijinal görüntüye daha yakın olarak yorumlanmaktadır.

$$SSIM(x, y) = [L(x, y)]^a [c(x, y)]^b [k(x, y)]^d \quad (20)$$

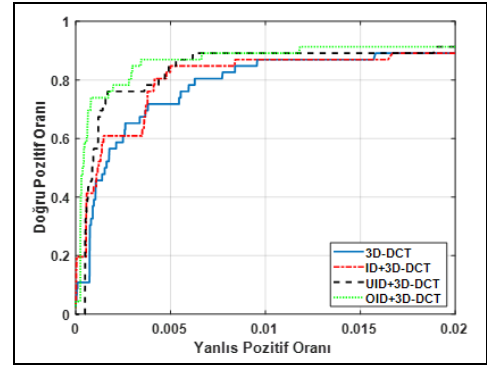
Hiperspektral görüntülerin 0.1 bpp bit oranında farklı yöntemler ile sıkıştırılması sonucunda elde edilen görsel sonuçlar incelendiğinde, 3D-DCT işlemi sonucunda meydana gelen görsel kalite bozunumları net bir şekilde gözükmemektedir. Aynı şekilde yöntem sonucunda bloklaşma etkileri de göze çarpmaktadır. Görseldeki bu etkiler SSIM metriğindeki değerler üzerinden de doğrulanmaktadır. Örneğin, Cuprite verisinde 3D-DCT sonucunda 150. bant görüntüsünün SSIM değeri 0.31 iken önerilen yöntemde bu değer 0.75'e yükseldiği görülmektedir. Benzer sonuçlar diğer verilerde de görülmektedir. Sonuç olarak, otomatik ışıklılık dönüşümü sonrasında elde edilen görsellerde bu bozunumların azaldığı fark edilmekte ve SSIM değerlerini 150. bant için en az 0.35 kadar yükseldiği görülmektedir. Elde edilen sonuçlar, kullanılan yöntemler açısından birbirleriyle karşılaştırıldığında üç veri için de otomatik ışıklılık dönüşümünün önemli ölçüde iyileştirmeler yaptığını göstermektedir [34].

### 3.4 Sıkıştırma sonrası anomali ve hedef tespiti sonuçları

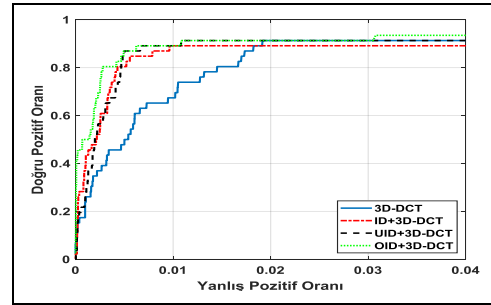
Anomali tespit başarımının ölçülebilmesi amacıyla önce Cuprite ve Moffett Field verilerindeki anomali bölgeleri işaretlenerek yer doğrusu haritası oluşturulmuştur. Haritanın oluşturulmasında [40]'daki çalışma referans alınmıştır. Anomalilerin tespiti için RX (Reed Xiaoli) tespit yöntemi [41] kullanılmış ve bunun sonucunda en yüksek değere sahip belirli sayıda piksel anomali olarak işaretlenmiştir. Sonrasında sıkıştırılmış olan verilere RX uygulanarak anomaliler için tespit başarımı incelenmiştir. Anomali tespit başarımları ROC eğrileri üzerinden karşılaştırılmıştır.

Anomali tespiti için yöntemlerin Cuprite ve Moffett Field verilerinde 0.1 ve 0.3 bpp bit oranlarındaki performansları Şekil 15'te verilmiştir. ROC eğrileri incelendiğinde özellikle 0.1 bpp bit oranında UID+3D-DCT yaklaşımının 3D-DCT ve ID+3D-DCT yaklaşımlarına göre aynı yanlış pozitif oranlarında daha yüksek doğru pozitif oranına sahip olduğu; dolayısıyla anomali bölgelerini daha iyi koruduğu görülmektedir. OID+3D-DCT yaklaşımının ise genel olarak diğer tüm yöntemlerden daha iyi anomali tespit başarımı sağladığı görülmektedir.

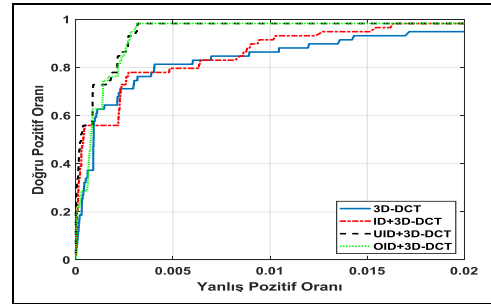
Hedef tespit başarımlarının ölçülmesi için Cuprite verisi içerisinde iki farklı bölge belirlenmiştir. Hedef bölgeleri [33]'teki yayına benzer olarak anomali bölgeleri içerisinde belirlenmiştir. Hedef imzaları için orijinal verideki hedef bölgesinin ortasından alınan tek bir piksel spektrumu kullanılmıştır. Orijinal veri üzerinde ACE (Adaptive Cosine Estimator) hedef tespiti yöntemleriyle elde edilen sonuçlar üzerinden hedef bölgeleri işaretlenmiştir. Sıkıştırılmış veride ACE yöntemi koşurulduğunda hedef bulma performansı yine ROC eğrileri üzerinden değerlendirilmiştir.



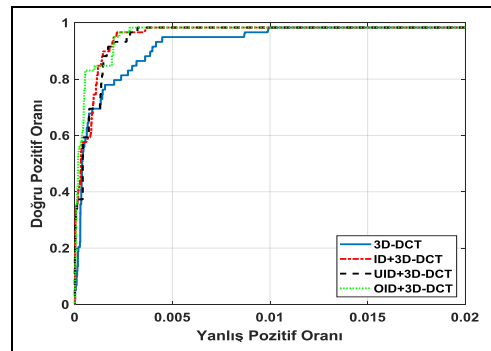
(a): Cuprite-0.1 bpp.



(b): Cuprite-0.3 bpp.



(c): Moffett-0.1 bpp.

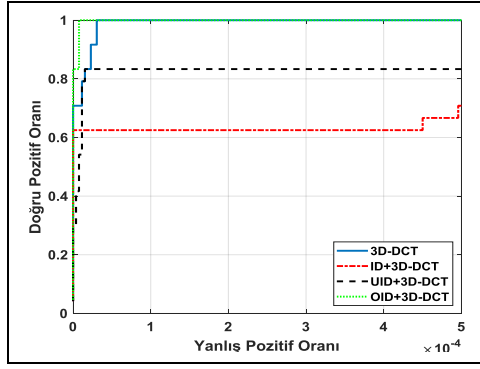


(d): Moffett-0.3 bpp.

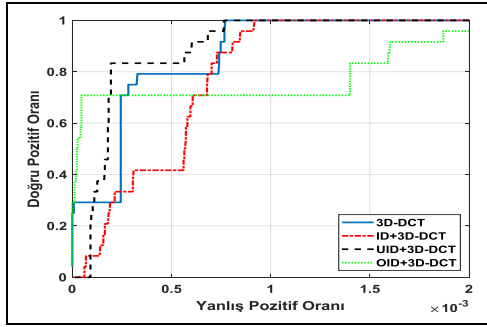
Şekil 15. Hiperspektral görüntülerin anomali tespiti sonuçları.

Figure 15. Anomaly detection results of hyperspectral images.

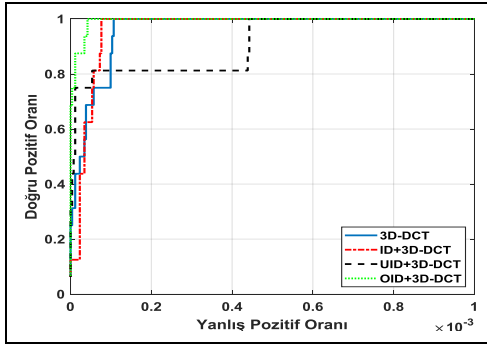
Hedef tespiti için yöntemlerin Cuprite verisinde iki farklı hedefe ait 0.1 ve 0.3 bpp bit oranlarındaki performansları Şekil 16'da paylaşılmıştır. OID+3D-DCT yaklaşımının özellikle düşük yanlış pozitif oranlarında çok daha yüksek doğru pozitif oranı elde ettiği görülmektedir.



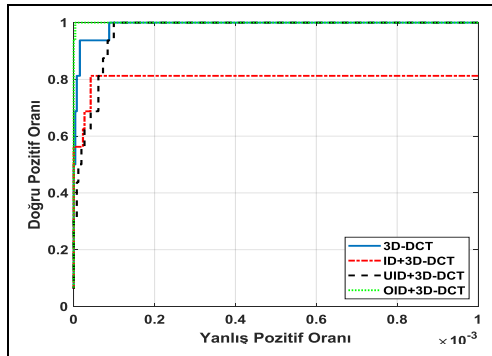
(a): Cuprite-Hedef 1-0.1 bpp.



(b): Cuprite-Hedef 1-0.3 bpp.



(c): Cuprite-Hedef 2-0.1 bpp.



(d): Cuprite-Hedef 2-0.3 bpp.

Şekil 16. Hiperspektral görüntülerin hedef tespiti sonuçları.

Figure 16. Target detection results of hyperspectral images.

Bu sonuçlardan ayrıca hedefleri daha az gözetken yapının ID+3D-DCT yaklaşımı olduğu da açıktır. Sonuç olarak, gerek anomali tespiti gerekse hedef tespiti üzerinden elde edilen sonuçlar yorumlandığında genel olarak önerilen OID+3D-DCT

yönteminin karşılaştırmada kullanılan diğer yöntemlerden daha başarılı bir performans gösterdiği gözlemlenmiştir.

#### 4 Sonuçlar

Bu çalışmada, hiperspektral görüntülerin dönüşüm temelli sıkıştırılmasında sıkça kullanılan 3D-DCT'ye ön işlem olarak otomatik uyarlamalı ışıklılık dönüşümünün uygulanması önerilmiştir. Önerilen yöntem sayesinde işlemsel karmaşıklığı düşük 3D-DCT dönüşümü ile elde edilen sıkıştırma başarımı önemli ölçüde artırılmıştır. Çalışmada birkaç önemli nokta öne çıkmaktadır. Birincisi, uyarlamalı ışıklılık dönüşümü ile hiperspektral görüntülerin standart hale gelmiş  $8 \times 8 \times 8$ 'lik blok boyutları dışında bantlar arasındaki ilinti değerleri dikkate alınarak gruplandırılması ve sonrasında sıkıştırılması, veriyi orijinale daha benzer şekilde geri elde etmeyi sağlamaktadır. İkinci olarak bant gruplarının otomatik bir şekilde belirlenmesi gerçekleştirilmiştir. Geliştirilen yöntem tamamen otomatik çalışmakta olup, kullanıcıdan grup sayısı istememektedir. Önerilen bu yöntem sayesinde daha yüksek işaret-gürültü oranı ve ortalama spektral ilinti değerine ulaşabilmektedir. Ayrıca, önerilen yöntemin en düşük ortalama spektral açı uzaklığına sahip olduğu da görülmektedir. Sıkıştırılan görüntülerin kalite ölçütleri kadar farklı uygulamalardaki başarımları da önemlidir. Bu açıdan, farklı yöntemlerle sıkıştırılan hiperspektral görüntüler üzerinde anomali ve hedef tespit performansı irdelenmiştir. Önerilen yöntemin 3D-DCT temelli diğer yöntemlere göre daha iyi biçimde anomali ve hedefleri koruduğu görülmüştür.

#### 5 Conclusions

In this work, it is proposed to apply automatic adaptive luminance transform as a pretreatment to 3D-DCT, which is frequently used in transform based compression of hyperspectral images. Thanks to the proposed method, the compression performance obtained by the 3D-DCT transform with low computational complexity has been significantly increased. Several important points stand out in the paper. First, using adaptive luminance transform, hyperspectral images are grouped and then compressed, taking into account the correlation values between the bands, except for the standardized  $8 \times 8 \times 8$  block sizes, resulting to restore the data in more similar to the original. Secondly, automatic segmentation of the band groups is realized. The developed method works completely automatically and does not require the number of groups from the user. Thanks to this proposed method, higher signal-to-noise ratio and average spectral correlation value can be achieved. It is also seen that the proposed method has the lowest average spectral angle distance. The performance of compressed images in different applications is as important as the quality criteria. In this respect, anomaly and target detection performances on hyperspectral images compressed by different methods are examined. It has been observed that the proposed method preserves the anomalies and targets better than other 3D-DCT based methods.

#### 6 Kaynaklar

- [1] Öztürk Ş, Esin Y, Artan Y, Özdiil Ö, Demirel B. "Importance of band selection for ethene and methanol gas detection in hyperspectral imagery". *9th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, Amsterdam, Netherlands, 23-26 September 2018.

- [2] Xu Y, Wu Z, Wei Z, Dalla Mura M, Chanussot J, Bertozzi A. "Gas plume detection in hyperspectral video sequence using low rank representation". *IEEE 2016 International Conference on Image Processing (ICIP)*, Phoenix, USA, 25-28 September 2016.
- [3] Çeşmeci D, Karaca AC, Ertürk A, Güllü MK, Ertürk S. "Hyperspectral change detection by multi-band census transform". *IEEE 2018 Geoscience and Remote Sensing Symposium*, Quebec City, Canada, 13-18 July 2018.
- [4] Kumar JP, Deshpande S, Inamdar A. "Detection of fertilizer quantity in soil using hyperspectral data". *9th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, Amsterdam, Netherlands, 23-26 September 2018.
- [5] Zhao Y, Wang Y, Wei D, Mu H, Ning T. "Application of hyperspectral imaging in measurement real-time of seeds". *IEEE 2016 International Conference on Smart Cloud*, New York, NY, 18-20 November 2016.
- [6] Liang Y, Markopoulos PP, Saber ES. "Subpixel target detection in hyperspectral images with local matched filtering in SLIC superpixels". *8th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, Los Angeles, USA, 21-24 August 2016.
- [7] Ben Salem M, Ettabaï KS, Bouhlel MS. "Anomaly detection in hyperspectral images based spatial spectral classification". *7th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT)*, Hammamet, Tunisia, 18-26 December 2016.
- [8] Xu H, Wang XJ. "Applications of multispectral hyperspectral imaging technologies in military". *Infrared And Laser Engineering*, 36(1), 13-18, 2007.
- [9] Weijtmans PJC, Shan C, Tan T, Brouwer de Koning SG, Ruers TJM. "A dual stream network for tumor detection in hyperspectral images". *IEEE 16th International Symposium on Biomedical Imaging*, Venice, Italy, 8-11 April 2019.
- [10] Pike R, Lu G, Wang D, Chen ZG, Fei B. "A minimum spanning forest-based method for noninvasive cancer detection with hyperspectral imaging". *IEEE Transactions on Biomedical Engineering*, 63(3), 653-663, 2016.
- [11] Christophe E, "Hyperspectral Data Compression Tradeoff, In: Prasad S, Bruce L, Chanussot J, Optical Remote Sensing". *Augmented Vision and Reality*, 3(1), 9-29, 2011.
- [12] Pennebaker WB, Mitchell JL. *JPEG: Still image data compression standard*. 2nd ed. New York, USA, Springer Verlag, 2006.
- [13] Schelkens, Peter, Athanassios Skodras, and Touradj Ebrahimi. *The JPEG 2000 Suite*. 1st ed. West Sussex, United Kingdom, John Wiley & Sons, 2009.
- [14] Abousleman GP, Marcellin MW, Hunt BR. "Compression of hyperspectral imagery using the 3-D DCT and hybrid DPCM/DCT". *IEEE Transactions on Geoscience and Remote Sensing*, 33(1), 26-34, 1995.
- [15] Thyagarajan K. S. *Still Image Video Compression with MATLAB*, 1st ed. New Jersey, USA, John Wiley & Sons, 2011.
- [16] Yıldız K, Buldu A. "Wavelet transform and principal component analysis in fabric defect detection and classification". *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, 23(5), 622-627, 2016.
- [17] Penna B, Tillo T, Magli E, Olmo G. "Transform coding techniques for lossy hyperspectral data compression". *IEEE Transactions on Geoscience and Remote Sensing*, 45(5), 1408-1421, 2007.
- [18] Mei S, Khan BM, Zhang Y, Du Q. "Low-complexity hyperspectral image compression using folded PCA and JPEG2000". *IEEE 2018 International Geoscience and Remote Sensing Symposium (IGARSS)*, Valencia, Spain, 22-27 July 2018.
- [19] Tang X, Pearlman WA. *Three-dimensional wavelet-based compression of hyperspectral images*. Editors: Motta G, Rizzo F, Storer JA. *Hyperspectral Data Compression*, 273-308, Boston, USA, Springer US, 2006.
- [20] Lim, S, Sohn K, Lee C. "Compression for hyperspectral images using three dimensional wavelet transform". *IEEE 2001 International Geoscience and Remote Sensing Symposium (IGARSS)*, Sydney, Australia, 9-13 July 2001.
- [21] Wang, Y, Rucker JT, Fowler JE. "Three-dimensional tarp coding for the compression of hyperspectral images". *IEEE Geoscience and Remote Sensing Letters*, 1(2), 136-140, 2004.
- [22] Hassanzadeh S, Karami A. "Compression and noise reduction of hyperspectral images using non-negative tensor decomposition and compressed sensing". *European Journal of Remote Sensing*, 49(1), 587-598, 2016.
- [23] Huber-Lerner M, Hadar O, Rotman SR, Huber-Shalem R. "Compression of hyperspectral images containing a subpixel target". *IEEE Journal Of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(6), 2246-2255, 2014.
- [24] Santos L, López S, Callico GM, Lopez JF, Sarmiento R. "Performance evaluation of the H. 264/AVC video coding standard for lossy hyperspectral image compression". *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 5(2), 451-461, 2011.
- [25] Qiao T, Ren J, Sun M, Zheng J, Marshall S. "Effective compression of hyperspectral imagery using an improved 3D DCT approach for land-cover analysis in remote-sensing applications". *International Journal of Remote Sensing*, 35(20), 7316-7337, 2014.
- [26] Karami A, Behesti S, Yazdi M. "Hyperspectral image compression using 3D discrete cosine transform and support vector machine learning". *11th International Conference on Information Science, Signal Processing and their Applications (ISSPA)*, Canada, Montreal, 3-5 July 2012.
- [27] Karami A, Yazdi M, Asli AZ. "Hyperspectral image compression based on tucker decomposition and discrete cosine transform". *2nd International Conference on Image Processing Theory, Tools and Applications*, Paris, France, 7-10 July 2010.
- [28] Haiyan T, Wenbang S, Bingzhe G, Fengjing Z. "Research on quantization and scanning order for 3-D DCT video coding". *International Conference on Computer Science and Electronics Engineering*, Hangzhou, China, 23-25 March 2012.
- [29] Engin MA, Cavusoglu B. "New approach in image compression: 3D spiral JPEG". *IEEE Communications Letters*, 15(11), 1234-1236, 2011.
- [30] Can E, Karaca AC, Danişman M, Urhan O, Güllü MK. "Compression of hyperspectral images using luminance transform and 3D-DCT". *IEEE 2018 International Geoscience and Remote Sensing Symposium*, Valencia, Spain, 22-27 July 2018.

- [31] Can E, Karaca AC, Danişman M, Urhan O, Güllü MK. "Compression of hyperspectral images using adaptive luminance transform". *26th IEEE Signal Processing and Communications Applications Conference (SIU2018)*, Izmir, Turkey, 2-5 Mayıs 2018.
- [32] Du Q, Zhu W, Yang H, Fowler JE. "Segmented principal component analysis for parallel compression of hyperspectral imagery". *IEEE Geoscience and Remote Sensing Letters*, 6(4), 713-717, 2009.
- [33] Zhou S, Xu Z, Liu F, "Method for determining the optimal number of clusters based on agglomerative hierarchical clustering". *IEEE Transactions on Neural Networks and Learning Systems*, 28(12), 3007-3017, 2017.
- [34] Murtagh F, Contreras P. "Algorithms for hierarchical clustering: an overview". *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(1), 86-97, 2012.
- [35] Gerçek D, Çeşmeci D, Güllü MK, Ertürk A, Ertürk S, "Automated co-registration of satellite images through luminance transform". *The Photogrammetric Record*, 31(156), 407-427, 2016.
- [36] Kwon HJ, Lee SH, Lee GY, Sohng KL, "Luminance adaptation transform based on brightness functions for LDR image reproduction". *Digital Signal Processing*, 30(1), 74-85, 2014.
- [37] Skretting K. "MATLAB'da Huffman ve Aritmetik Kodlama". <http://www.ux.uis.no/~karlsk/proj99/index.html> (01.07.2019).
- [38] Du Q, Ly N, Fowler JE. "An operational approach to PCA+JPEG2000 compression of hyperspectral imagery". *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(6), 2237-2245, 2014.
- [39] Wang Z, Bovik A, Sheikh H, Simoncelli E. "Image quality assessment: from error visibility to structural similarity". *IEEE Transaction on Image Processing*, 13(4), 600-612, 2004.
- [40] Li W, Wu G, Du Q. "Transferred deep learning for anomaly detection in hyperspectral imagery". *IEEE Geoscience and Remote Sensing Letters*, 14(5), 597-601, 2017.
- [41] Reed I, Yu X. "Adaptive multiple-band cfar detection of an optical pattern with unknown spectral distribution". *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 38(1), 1760-1770, 1990.