

**KOCAELİ ÜNİVERSİTESİ \*FEN BİLİMLERİ ENSTİTÜSÜ**

**ÇOKLU TANIMLAMALI KODLANMIŞ İMGE VE  
VİDEOLARIN EN UYGUN SÜZGEÇLENMESİ**

**YÜKSEK LİSANS TEZİ**

**Elektronik ve Haberleşme Mühendisi Çağlar ATEŞ**

**Anabilim Dalı: Elektronik ve Haberleşme Mühendisliği**

**Danışman: Doç. Dr. Oğuzhan Urhan**

**KOCAELİ, 2009**

**KOCAELİ ÜNİVERSİTESİ\*FEN BİLİMLERİ ENSTİTÜSÜ**

**ÇOKLU TANIMLAMALI KODLANMIŞ İMGE VE  
VİDEOLARIN EN UYGUN SÜZGEÇLENMESİ**

**YÜKSEK LİSANS TEZİ**

**Elektronik ve Haberleşme Mühendisi Çağlar ATEŞ**

**Tezin Enstitüye Verildiği Tarih: 19.05.2009**

**Tezin Savunulduğu Tarih: 30.07.2009**

**Tez Danışmanı**

**Doç. Dr.  
Oğuzhan URHAN**

  
(.....)

**Üye**

**Prof. Dr  
Sarp ERTÜRK**

  
(.....)

**Üye**

**Doç. Dr  
Cabir VURAL**

  
(.....)

**KOCAELİ, 2009**

## ÖNSÖZ ve TEŞEKKÜR

İnternetin yaygınlaşması ile beraber kablolu ve özellikle kablosuz ağların kullanımı artmıştır. Kablosuz ağların artması, iletim kanallarının gürültüye maruz kalmasına neden olmuştur. Kanallardan iletilen veri paketlerinin hatasız ve güvenli bir şekilde iletimini sağlamak için birçok yöntem önerilmiştir. Bu yöntemlerden birisi de çoklu tanımlamalı kodlamadır.

Bu tez çalışmasında, çoklu tanımlamalı kodlama yöntemlerinin en uygun süzgeçleme yaklaşımı kullanılarak imge ve video kodlamaya uygulanması çalışmaları yapılmıştır. Yapılan çalışmaların bu alana emek verenlere ve bu alanda çalışacak araştırmacılara yararlı olmasını dilerim.

Bu konuda çalışma yapmama olanak sağlayan, çalışmanın her aşamasında ilgi ve desteğini eksik etmeyen değerli hocalarım Doç. Dr. Oğuzhan Urhan ve Prof. Dr. Sarp Ertürk olmak üzere, bu uzun süreçte sürekli birlikte çalıştığımız tüm Kulis ekibine ve son olarak, bu güne kadar bana hem maddi hem manevi katkıda bulunan ve bulunmaya devam eden biricik aileme teşekkür ederim.

Bu çalışma, Türkiye Bilimsel ve Teknik Araştırma Kurumu, TÜBİTAK tarafından EEEAG/107E155 numaralı proje kapsamında desteklenmiştir.

Mayıs 2009, KOCAELİ

Çağlar ATEŞ

## İÇİNDEKİLER

ÖNSÖZ ve TEŞEKKÜR.....	i
İÇİNDEKİLER .....	ii
ŞEKİLLER DİZİNİ.....	iii
TABLolar DİZİNİ .....	iv
SİMGELER DİZİNİ VE KISALTMALAR.....	v
ÖZET .....	vi
İNGİLİZCE ÖZET.....	vii
1. GİRİŞ .....	1
2. ÇOKLU TANIMLAMALI KODLAMA [1] .....	4
2.1 Giriş .....	4
2.2 Çoklu Tanımlamalı Kodlama .....	7
2.2.1 Ses Kodlama Uygulamasında Kanal Ayırma .....	7
2.2.1.1 Jayant Modeli .....	7
2.2.1.2 Jayant Modelinin Analizi .....	9
2.2.2 Kanal Ayırma için Skaler Nicemleme .....	10
2.3 Oran Bozunum Bölgesi .....	12
2.4 Çoklu Tanımlamalı Oran Bozunum Bölgesi .....	15
3. ÇOKLU TANIMLAMALI KODLAMADA KULLANILAN YÖNTEMLER ..	18
3.1 Giriş .....	18
3.2 Çok Aşamalı Aşağı Örnekleme Temelli Çoklu Tanımlamalı Kodlama.....	18
3.2.1 Ön-İşlem .....	19
3.2.2 Çoklu Tanımlama Üretimi .....	21
3.2.3 Son-işlem .....	22
3.3 Çoklu Tanımlamalı Skaler Nicemleyici .....	22
3.3.1 Nicemleme ve İndis Haritası Oluşturma.....	24
4. EN UYGUN SÜZGEÇLEME KULLANARAK İMGE İÇİN ÇOKLU TANIMLAMALI KODLAMA.....	30
4.1 EUS Yönteminin Çok Aşamalı Aşağı Örnekleme Temelli ÇT İmge Kodlamaya Uygulanması .....	30
4.1.1 Deneysel Sonuçlar .....	34
4.2 Dalgacık Temelli ÇTSN Yaklaşımları İçin En Uygun Süzgeçleme ile İmge Kodlama .....	40
4.2.1 Deneysel Sonuçlar .....	40
5. EN UYGUN SÜZGEÇLEME KULLANARAK VİDEO İÇİN ÇOKLU TANIMLAMALI KODLAMA.....	43
5.1 EUS Yönteminin Çok Aşamalı Aşağı Örnekleme Temelli ÇT Video Kodlamaya Uygulanması.....	43
5.1.1 Deneysel Sonuçlar .....	48
SONUÇLAR ve ÖNERİLER.....	51
KAYNAKLAR .....	53
KİŞİSEL YAYINLAR ve ESERLER.....	57
ÖZGEÇMİŞ .....	58

## ŞEKİLLER DİZİNİ

Şekil 2.1: Haberleşme kanalından alınan imgelerin paket bazında gösterimi [1].....	6
Şekil 2.2: Jayant tarafından önerilen ses kodlama için kanal ayırma yapısı [4] .....	7
Şekil 2.3: Jayant modelinin oran bozunum grafiği [1].....	8
Şekil 2.4: Farklı nicemleme yapıları [1] .....	11
Şekil 2.5: İki tanımlamalı ÇTK yapısı .....	13
Şekil 2.6: Farklı r değerleri için oran bozunum sınırı [1] .....	17
Şekil 3.1: ÇTK tabanlı aşırı örneklemenin blok gösterimi [4].....	19
Şekil 3.2: Önişlem bloğu içerisindeki işlemler [4].....	20
Şekil 3.3: Frekans uzayında artıklık ekleme gösterimi [4] .....	20
Şekil 3.4: Frekans uzayında örtüşme [4].....	21
Şekil 3.5: Aşağı örnekleme yolu ile ÇTK [27] .....	22
Şekil 3.6: ÇTSN kodlama sisteminin genel yapısı.....	23
Şekil 3.7: İndis değeri üretme [32].....	23
Şekil 3.8: İndis haritası oluşturma [34].....	25
Şekil 3.9: Farklı indis oluşturma [28] .....	26
Şekil 3.10: İç içe indis atama yöntemi kullanılarak oluşturulan indis haritaları [28]	28
Şekil 3.11: İndis matrisi tarama yöntemleri [28] .....	29
Şekil 3.12: Farklı köşegen değerleri için indis matrisi örneği .....	29
Şekil 4.1: Optimize edilmiş IPCG algoritması [38].....	33
Şekil 4.2: İki tanımlamanın alıcıya ulaşması durumu [48] .....	34
Şekil 4.3: Farklı AÖO değerleri için oran bozunum grafikleri .....	35
Şekil 4.4: Farklı AÖO değerleri için oran bozunum grafikleri (devam).....	36
Şekil 4.5: Barbara imgesi için oran bozunum grafikleri .....	37
Şekil 4.6: Peppers imgesi için oran bozunum grafikleri .....	38
Şekil 4.7: Lena ve Barbara imgeleri için dalgacık tabanlı kodlama sonuçları.....	41
Şekil 4.8: Peppers ve Goldhill imgeleri için dalgacık tabanlı kodlama sonuçları .....	42
Şekil 5.1: ÇT video kodlama döngüsünde uyumsuzluğu giderme[27].....	43
Şekil 5.2: ÇTK temelli çok aşamalı video kodlama.....	45
Şekil 5.3: EUS yönteminin H.264/AVC'ye uygulanması.....	46
Şekil 5.4: Farklı test dizileri ve yöntemleri için ÇTK sonuçları .....	49

## **TABLolar DİZİNİ**

Tablo 3.1: Tanımlamaların birleştirilmesi sonucu elde edilen değerler [28] .....	26
Tablo 3.2: Tablo 3.1’de verilen değerler için merkezi ve yan en küçük karesel hata değerleri [28].....	27

## SİMGELER DİZİNİ VE KISALTMALAR

AKD	: Ayrık Kosinüs Dönüşümü
AÖ	: Aşırı Örnekleme
AÖO	: Aşırı Örnekleme Oranı
AR	: Autoregressive
ARQ	: Automatic Interrupt Request
CIF	: Common Intermediate Format
ÇD	: Çift Doğrusal Aradeğerleme
ÇK	: Çift Kübik Aradeğerleme
ÇT	: Çoklu Tanımlama
ÇTK	: Çoklu Tanımlamalı Kodlama
ÇTSN	: Çoklu Tanımlamalı Skaler Nicemleme
EUS	: En Uygun Süzgeçleme
FDKM	: Farksal Darbe Kod Modülasyonu
IP	: Internet Protocol
IPCG	: Iterative Preconditioned Conjugate Gradients Method
JPEG	: Joint Photographic Expert Group
MDC	: Multiple Description Coding
MSE	: Minimum Square Error
OSI	: Open Systems Interconnection
PCT	: Pairwise Correlating Transform
PD	: Polyphase Downsampling
PSNR	: Peak Signal To Noise Ratio
RD	: Rate Distortion
RLC	: Run Length Coding
SNR	: Signal Noise Ratio
SPIHT	: Set Partitioning In Hierarchical Trees
TAKD	: Ters Ayrık Kosinüs Dönüşümü
TCP	: Transmission Control Protocol

# ÇOKLU TANIMLAMALI KODLANMIŞ İMGE VE VİDEOLARIN EN UYGUN SÜZGEÇLENMESİ

Çağlar ATEŞ

**Anahtar Kelimeler:** İmge ve Video Kodlama, Çoklu Tanımlamalı Kodlama, En Uygun Süzgeçleme

**Özet:** Çoklu tanımlamalı kodlama (ÇTK) yaklaşımı günümüzde oldukça yaygınlaşan paket ağları üzerinden veri transferi için kullanılmaktadır. Bu yaklaşım temelde veri kayıplarına açık ağlar üzerinden çokluortam (multimedia) verisinin verimli bir şekilde iletilmesini sağlamaktadır. Bunu orijinal verinin birden fazla farklı bit dizisi (bit-stream) şeklinde kodlanması ve iletilmesi yolu ile olası ağ kayıplarından daha az etkilenmesi sayesinde gerçekleştirmektedir. İletim ortamındaki paket kayıpları durumunda, veri kabul edilebilir kalitede geri elde edilebilmektedir.

Bu tezde standart ÇTK yöntemlerinin başarımını artırabilecek son-işlem olarak en uygun süzgeçleme (EUS) temelli bir yaklaşım önerilmektedir. Bu yaklaşım, temelde her bir tanımlama için kodlanmış veri ile orijinal veri arasındaki farkı en küçükleyecek süzgeç katsayılarını en küçük kareler yöntemi ile elde etmektedir. Böylelikle her bir tanımlama, orijinal tanımlama verisi ve ilgili en uygun süzgeç katsayılarından oluşturulacaktır. Süzgeç katsayıları kullanılarak alıcıda gerçekleştirilecek süzgeçleme ile daha yüksek kalitede veri oluşturulabilecektir.

Bu tez kapsamında standart ÇTK yaklaşımları olan çok aşamalı aşağı örnekleme ve çoklu tanımlamalı skaler niceme yöntemlerine EUS yöntemi tümleştirilmiştir. Önerilen yöntemler imge ve video verilerinin daha etkin şekilde kodlanmasına olanak sağlamıştır.



# OPTIMAL FILTERING OF MULTIPLE DESCRIPTION CODED IMAGES AND VIDEOS

Çağlar ATEŞ

**Keywords:** Image and Video Coding, Multiple Description Coding, Optimal Filtering

**Abstract:** The Multiple Description Coding (MDC) approach is used for data transfer over packet networks which are widespread nowadays. This approach fundamentally provides efficient transmission of multimedia data through this kind of error prone networks. MDC carries out this operation by coding and transmitting the original data using more than one bit-stream and therefore reduces influence of possible packet loss. MDC approach enables reconstruction of data at acceptable quality level in the case of possible packet losses.

An optimal filtering based approach which can improve the performance of standard MDC methods is proposed in this thesis as a post-processing operation. This method basically obtains the filter coefficients by least squares approach minimizing the difference between encoded and original data for each description. Thus, each description are composed description data and appropriate filter coefficients. A higher quality data can be reconstructed performing a filtering operation at the decoder using the filter coefficients that are included to the descriptions.

Standard MDC approaches, i.e. poly-phase down-sampling and multiple description scalar quantization methods, are combined with optimal filtering method within the scope of this thesis. Proposed methods enable more effective coding of image and video data.

## 1. GİRİŞ

Günümüzde kablolu ve özellikle kablosuz ağlar giderek yaygınlaşmaktadır. Kablosuz ortamlar kablolu ortamlara göre gürültüden daha fazla etkilenirler. Kablosuz ortamlarda kapsama alanları giderek genişlemekte, buna paralel olarak çevredeki çeşitli elektriksel kaynaklar sonucu kablosuz iletişim kanallarına binen elektriksel, manyetik veya fiziksel gürültüler artmaktadır. İletilecek verinin karşı tarafta tekrardan kabul edilebilir şekilde geri oluşturulabilmesi gerekmektedir. Bu sebeple hataya eğilimli ağlar üzerinden veri paketlerinin iletimi önemli hale gelmiştir.

Çoklu tanımlamalı kodlama (ÇTK) yaklaşımı temelde veri kayıplarına açık, hataya eğilimli ağlar üzerinden ses, imge, video gibi çoklu ortam (multimedia) verilerinin verimli şekilde iletilmesini sağlamaktadır. Bunu orijinal verinin birden fazla farklı bit dizisi (bit-stream) şeklinde kodlanması ve iletilmesi yolu ile olası ağ kayıplarından daha az etkilenmesi sayesinde gerçekleştirmektedir. Ayrıca video konferans, sayısal video yayını gibi birçok çoklu ortam uygulamasının yüksek bant genişliği ve sınırlı gecikme süresi gibi kısıtları vardır. Bu sınırlamalar gerçek zamanlı veri iletimine ihtiyaç duyan birçok uygulamada problem oluşturmaktadır. ÇTK bu problemleri çözmek için geliştirilmiş yöntemlerden biridir. ÇTK yaklaşımı ile bu tip veri ağlarındaki olası kayıp durumlarında, iletilen herhangi bir tanımlama kullanılarak kabul edilebilir ve orijinal veriye olabildiğince yakın bir sonuç verisinin oluşturulması amaçlanmaktadır.

Kablosuz ağlar ve IP ağları (temelde Internet) üzerinden gerçekleştirilen çoklu ortam veri aktarımının birçok uygulaması bulunmaktadır. Özellikle yakın gelecekte oldukça yaygınlaşması beklenen IP Televizyonu (IPTV) uygulamasıyla, birçok televizyon yayınına ilişkin video verisinin internet üzerinden aktarılması amaçlanmaktadır. Ayrıca ülkemizde de kısa bir süre içerisinde uygulanması planlanan üçüncü nesil mobil haberleşme uygulamaları içerisinde görüntülü konuşma önemli bir yer tutmaktadır. Bu tip tüketici elektroniği uygulamalarının yanı sıra mobil gözlem

uygulamalarında da ÇTK yaklaşımı etkin bir şekilde kullanılabilir. Ülkemizde de kullanılan MOBESE adı verilen “Kent Bilgi ve Güvenlik Sistemi” bu tip gözlem uygulamalarına bir örnek oluşturmaktadır. Bu sistemde geniş bir alana yayılmış yüzlerde kameradan alınan bilgiler, gerçek zamanlı olarak IP ağı üzerinden bir kontrol merkezine aktarılmaktadır.

Bahsedilen uygulamalar paket kayıp ve hatalarına açık ağlar üzerinden veri transferi gerçekleştirdiği için gönderilen veriler bazı durumlarda alıcıya ulaşamayabilir. Bu tip durumlarda alıcının verinin tekrar gönderilmesini istemesi genelde mümkündür. Ancak ses ve video gibi gerçek zamanlı veriler için bu durum oldukça uzun gecikmelere neden olacağından mümkün olmamaktadır. ÇTK yaklaşımı bu gibi paket kayıpları durumunda alıcının veriyi kabul edilebilir bir kalitede oluşturmasına olanak sağlamaktadır. Bu amaçla orijinal veri, vericide birden fazla, her biri kendi başına çözülebilir veri paketi (tanımlama) halinde kodlanmakta ve iletilmektedir. Bütün tanımlamaların alıcıya sorunsuz şekilde ulaşması durumunda veri yüksek kalitede geri oluşturulabilirken, daha az tanımlamanın ulaşması durumunda ise kabul edilebilir bir kalitede veri geri elde edilmektedir.

Tez kapsamında ilk olarak, ÇTK konusunun nasıl ortaya çıktığı ve temelleri anlatılmıştır. Jayant modelinden bahsedilmiş ve modelin matematiksel analizi verilmiştir. Skaler nicemleme kullanılarak kanal ayırma işleminden bahsedilip, ÇT probleminin farklı bit oranına bağlı olarak nasıl değiştiğini gösteren oran bozunum bölgesi incelenmiştir. Ardından farklı tanımlamalar kullanıldığında oran bozunum bölgesinin nasıl hesaplanacağı Gauss kaynak ve karesel hata bozunumu ölçüsü için gösterilmiştir. Böylece hem ÇTK konusunun temelleri anlatılmış hem de uygulanacak yöntemlerin hangi ölçütlere göre karşılaştırılacağı belirlenmiştir.

Çalışmanın ikinci kısmında, ÇTK’da kullanılan yöntemlerden olan çok aşamalı aşağı örnekleme temelli ÇTK ve çoklu tanımlamalı skaler nicemleme temelli ÇTK hakkında detaylı bilgi verilmiştir. Bu yöntemlerin nasıl gerçekleştirildiği açıklanmıştır.

Çalışmanın üçüncü kısmında, en uygun süzgeçleme yöntemi (EUS) kullanılarak imgeler için yeni bir ÇTK yöntemi önerilmiştir. İlk olarak EUS yöntemi çok aşamalı

aşağı örnekleme temelli ÇT imge kodlamaya uygulanmıştır. Kullanılan ÇTK yöntemi içerisinde, frekans uzayında aşırı örnekleme yapılarak tanımlamalar arasındaki ilinti artırılmıştır. Böylece bazı tanımlamaların kaybolması durumunda, alıcı tarafından alınan tanımlamalar içerisindeki artıklık kullanılarak kayıp verilerin kestirimi gerçekleştirilmiştir. Tanımlamalar standart JPEG kodlayıcı ile kodlanıp iletim ortamına gönderilmiştir. EUS içerisinde en küçük kareler yöntemi kullanılarak iletim kanalından alınacak işaretlerin görüntü bozunumu en küçüklenmiştir.

Çalışmanın üçüncü kısmının ikinci bölümünde, EUS yöntemi ile dalgacık temelli çoklu tanımlamalı skaler nicemleme yöntemi tümleştirilerek imge kodlamaya uygulanmıştır. Bu yöntemde giriş imgesinin dalgacık dönüşümü sonrası elde edilen dalgacık katsayıları o katsayıya ait alt banda bağlı olacak şekilde elde edilir. Ardından çoklu tanımlamalı skaler nicemleme kullanılarak ilgili indis haritasına göre tanımlama oluşturulur, aritmetik kodlama kullanılarak tanımlamalar sıkıştırılır ve iletim ortamına gönderilir. Farklı test imgelerinde yöntem denenmiş elde edilen sonuçlar gösterilip yorumlanmıştır.

Çalışmanın son bölümünde, çok aşamalı aşağı örnekleme temelli ÇTK yaklaşımına EUS yöntemi eklenerek video kodlamaya uygulanmıştır. Video akışı içerisindeki her bir çerçeve frekans uzayında tek bir yönde sıfır dolgulama işlemi gerçekleştirilmiştir. Böylece aşırı örneklenmiş çerçevelerin ters dönüşümü alınarak tekrardan uzamsal uzaya geri dönülmüş ve aşağı örnekleme yapılmıştır. Böylece elde edilen tüm tanımlamalar birbirlerinden ayrı olarak kodlanmış, sıkıştırılmış ve alıcıya iletilmiştir. Alıcıda elde edilen tanımlamalar ve EUS katsayıları kullanılarak video verisi geri elde edilmiştir.

## 2. ÇOKLU TANIMLAMALI KODLAMA [1]

### 2.1 Giriş

İnternetin yaygınlaşmasıyla insanların günlük hayatta çok kullandığı ve onların hayatını kolaylaştıran birçok uygulama sanal ortamlar üzerinden çalıştırılmaktadır. Örneğin TV, radyo gibi görsel ve yazılı materyaller internet ortamı üzerinden kullanıcılara yayınlanmaktadır. Sıkıştırma yöntemleri çoklu ortam verilerine uygulanmak istendiğinde bazı problemler ile karşılaşılmaktadır. Çoklu tanımlamadan bahsetmeden önce genel olarak kullanılan bir kodlama yapısı olan katmanlı kodlamadan (layered coding) bahsedilecektir.

En yaygın paket ağı olan internet üzerinden yapılan veri iletişimi Açık Sistemler Bağlantısı (Open System Interconnection) tarafından belirlenen 7 temel katman üzerinden gerçekleştirilir. Veriler iletim kontrol protokolü (transmission control protocol-TCP) paketlerine dönüştürülür ve ağ ortamına gönderilir. Kanaldaki gürültü, gecikme gibi etkilerden dolayı paket kayıpları oluyorsa alıcıda veri geri elde edilemez. Bu durumda genellikle paketler alıcıdan yeniden istenir. Bu işleme otomatik tekrar isteği (automatic interrupt request) adı verilir. Ancak gerçek zamanlı iletilmesi gereken ses, görüntü gibi çoklu ortam verilerinde bu tip bir yeniden iletim isteği genelde mümkün olmamaktadır. Çünkü bu işlem gecikmelere yol açmaktadır. Özellikle video uygulamaları için gecikme istenmeyen bir durumdur. Bu nedenle katmanlı kodlama ve çoklu tanımlamalı kodlama gibi paket kayıplarını tolere edebilen yaklaşımlar üzerinde çalışılmaktadır. Aşağıda detayları verilecek olan katmanlı kodlama ve çoklu tanımlamalı kodlama, imgeler üzerinden açıklanacaktır.

Katmanlı kodlama yaklaşımında imgedeki düşük frekanslı veriler taban paket olarak isimlendirilen tanımlama, yüksek frekansları içeren veriler ise ek paket olarak isimlendirilen tanımlama(lar) ile gönderilir. Alıcıya sadece taban paket ulaşırsa, düşük kalitede, görsel olarak kabul edilebilir bir imge elde edilebilir. Fakat sadece ek



















paket ulaşırsa, taban paket olmadığından alıcı ek paketi çözüp vericide sıkıştırılan veriyi geri elde edemez. Bu problemi çözmek için ÇTK yaklaşımı önerilmiştir. ÇTK yaklaşımında her bir paket kendi başına çözülebilir yapıdadır. Böylece üretilen her hangi bir paket alıcıya ulaşırsa veri kabul edilebilir oranda çözülecektir, ne kadar çok tanımlama alıcıda alınırsa buna bağlı olarak kalitede iyileşme görülür.

Şekil 2.1’de farklı yöntemler kullanılarak bir imgenin paket bazında alıcıda gösterimi örneklendirilmiştir. Üç farklı tip kaynak kodlama kullanılmıştır. Bunlar; yinelemeli olmayan kodlama (nonprogressive code), yinelemeli kodlama (progressive code) ve ÇTK’dır. İlk ikisi tekrardan iletim protokolünü (retransmission protocol) kullanır fakat ÇTK kullanmaz. Farz edelim ki paketler geldikçe Şekil 2.1’deki gibi imge ekranımızda gösterilsin. Burada satırlar her yeni paketle oluşturulan imgeyi, sütunlar ise ilgili kaynak kodlama yöntemlerini göstermektedir. Şekil 2.1’de her üç durum içinde altı paketin geldiği durum gösterilmekte üçüncü paketin ise iletim esnasında bozulduğu varsayılmaktadır. İlk iki durum için kayıp paket daha sonra tekrardan iletildiğinden ve yirminci zamanda alındığından dolayı kayıp geçicidir.

İlk paket kaybına kadar, yinelemeli durum açık bir şekilde diğer durumlara göre en iyisidir. Fakat bozuk olan üçüncü paketin tekrardan iletimi başarılı bir şekilde tamamlanana kadar alıcı tarafta yeni bir veri gösterilemez. Sonunda yinelemeli olmayan durum en iyi yöntem olur. Fakat yinelemeli durumdan biraz daha iyidir. ÇTK kullanımı 3. paketin bozuk olmasına rağmen paketler geldikçe imge kalitesini iyileştirir. İlk kayıptan hemen sonra tekrardan iletim tamamlanana kadar ÇTK en iyi imge kalitesini verir. Birçok uygulama için imge kalitesindeki ufak bir artış tekrardan iletim süresi için beklemeye değmez. Sonuçta kanalda paket kaybı yoksa yinelemeli kodlama en iyi yöntemdir. Kayıp varsa, imgenin geri elde edilmesi sağlam bir paket gelene kadar durur.

Geleneksel sistemdeki kaynak kodlama yapısında iletim mekanizmasına güvenilir. İletim mekanizması alıcının tüm paketlere ihtiyacının var olduğunu varsayar, fakat farklı kaynak kodlama yaklaşımlarında bu geleneksel yapı kullanılmayabilir. Tanımlamaların kayıplarının etkisini gidermek için ÇTK yaklaşımında sıkıştırma

verimliliğinden bir miktar ödün verilir. Böylece kanal kayıplarına karşı dayanıklılık sağlanırken sıkıştırma verimliliği nispeten azalır.

t	Tekrardan iletimli yinelemeli olmayan kodlama	Tekrardan iletimli yinelemeli kodlama	Tekrardan iletimli olmayan çoklu tanımlamalı kodlama
1			
2			
...	...	...	...
4			
5			
6			
...	...	...	...
20			

Şekil 2.1: Haberleşme kanalından alınan imgelerin paket bazında gösterimi [1]

Günümüzde anahtar uygulamalar için paket tabanlı haberleşme kullanılmasına rağmen ÇTK modeli diğer haberleşme senaryoları ile bağlanabilir. Aslında ÇTK devre anahtarlama telefon ağları için planlanmıştır. ÇTK hem ses hem de görüntü verilerine uygulanabilen bir yöntemdir. Modelin kaynak kodlama ve nicemleme için gerekli olan altyapısı ile ilgili açıklamalar için [2,3]'deki kaynaklar incelenebilir.

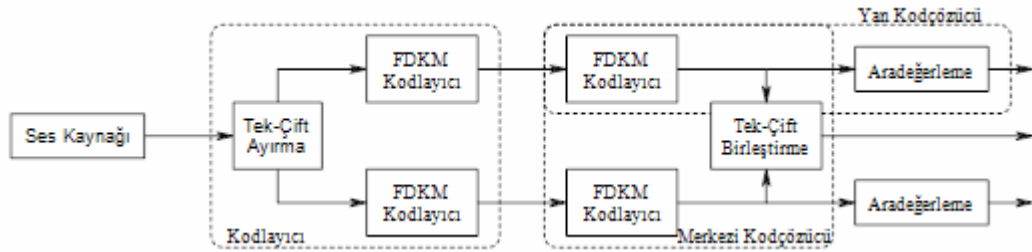
## 2.2 Çoklu Tanımlamalı Kodlama

ÇTK, Bell laboratuvarlarında bir grup araştırmacının ses verisinin telefon ağları üzerinden iletilmesi konusundaki çalışmaları sırasında geliştirilmiştir. ÇTK'nın temel fikri, iletim bağlantılarını meşgul duruma almadan bağlantının kesilmesini önlemek için, tek bir çağrıdan bilginin ayrılabilmesi ve 2 farklı bağlantı üzerinden veya kanaldan iletebileceğidir.

### 2.2.1 Ses Kodlama Uygulamasında Kanal Ayırma

#### 2.2.1.1 Jayant Modeli

Jayant Miller'dan bağımsız olarak yaptığı çalışmada kanal ayırma kodlama yönteminde ses örneklerini tek ve çift şeklinde ayırmayı böylece ses kodlamasının farklı kanallara bölünerek iletilmesini önermiştir [4,5]. Şekil 2.2'de önerdiği yöntem gösterilmiştir.



Şekil 2.2: Jayant tarafından önerilen ses kodlama için kanal ayırma yapısı [4]

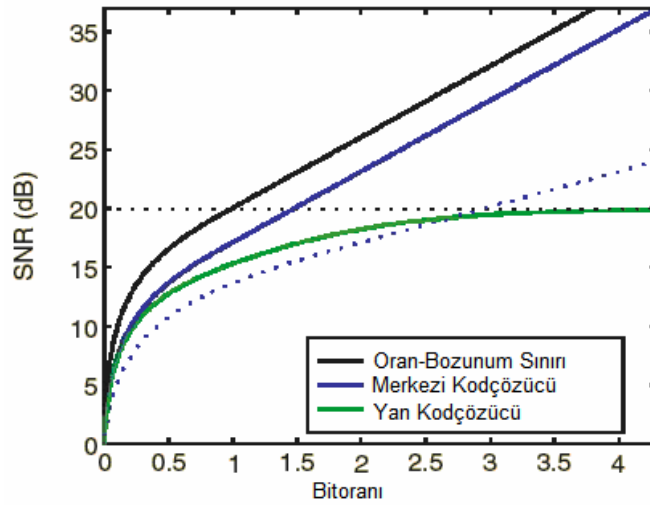
[4]'daki çalışmada ses kaynağından üretilen ses örnekleri vericideki kodlayıcı kısmında tek ve çift örnekler şeklinde gruplandırılır ve tanımlama olarak adlandırılır.



Bu tanımlamaların her biri birbirinden bağımsız farksal darbe kod modülasyonu (FDKM) ile modüle edilerek farklı kanallardan alıcıya iletilir. Ses işaretleri 3.2 kHz'lik bant sınırlı bir işaret olmakta ve genellikle 8 kHz ile örneklenmektedir. Jayant'ın sisteminde örnekleme frekansı altörnekleme yapılırken örtüşmeyi önlemek için 12 kHz alınmıştır.

Kodçözücü kısmında kanaldan alıcıya ulaşan tanımlamalar FDKM kodçözücüsünden geçirilerek kaynak kodçözme işlemi tamamlanır. Kodlayıcıda örneklerin tek çift şeklinde ayrılması sonucu elde edilen tanımlamalar, farklı kanallardan gelen örneklerin alıcıda tek çift şeklinde birleştirilmesi ile istenilen işaret geri oluşturulur. Kanaldaki etkilerden dolayı bazı tanımlamaların bozulduğu veya alıcıya ulaşmadığı durumlarda doğal olarak tek çift örnekler birleştirilemez. Kaybolan tanımlamaların içerdiği veri alıcıda uyarlamalı doğrusal aradeğerleme kullanılarak kısmi olarak elde edilmeye çalışılır.

Jayant'ın elde ettiği sonuçlar Şekil 2.3'de gösterilmektedir. Yan Kodçözücünün yüksek bit oranlarındaki SNR düşüşünün sebebi aradeğerleme hatalarından kaynaklanmaktadır. Merkezi Kodçözücü ile Yan Kodçözücü arasında yüksek bit oranlarına çıkıldığında oluşan asimptot, nicemleme hatalarından dolayı meydana gelmektedir. Ses sinyalleri ile elde edilen sonuçlar grafikteki değerler ile tutarlılık göstermektedir [4].



Şekil 2.3: Jayant modelinin oran bozunum grafiği [1]

### 2.2.1.2 Jayant Modelinin Analizi

Genellikle bir işaretin içerisinde peş peşe örnekler alınır, alınan örnekler arasında yüksek ilinti vardır denilebilir. Elde edilen bu örnekler neredeyse aynı bilgiyi içerir. Bu yöntem kanal ayırmada kullanılır. Jayant'ın modelinin basit matematiksel analizi şu şekildedir [1].

$$x[k] = ax[k-1] + z[k], k \in Z \quad (2.1)$$

Burada  $z[k]$  bağımsız, sıfır ortalamaya sahip Gauss rastlantı değişkenini ifade etmektedir. (2.1) deki ifade birinci dereceden özbağlanımlı (autoregressive-AR) işaret modelidir [6]. Denklemdaki  $a$  ifadesi  $|a| < 1$  olacak şekilde, ardışık örnekler arasındaki ilişkiyi belirtir.  $z[k]$  varyansını  $1 - a^2$  olarak alırsak  $z[k]$  değişimi (variance) değerini  $1 - a^2$  olarak ayarlarsak  $x[k]$  birim güç olur.

Değişintisi  $\sigma^2$  olan bir Gauss kaynağı'nın oran bozunum fonksiyonu  $D(R) = \sigma^2 2^{-2R}$  dir. Gauss AR kaynaklarının oran bozunum fonksiyonu herhangi oran değerleri için kolaylıkla hesaplanabilir [7]. Verilen kaynaklar için oran bozunum fonksiyonu denklem (2.2)'de verilmiştir.

$$D(R) = (1 - a^2)2^{-2R}, R \geq \log_2(1 + a) \text{ için} \quad (2.2)$$

Tanımlamalar tek ve çift örneklerden oluşacağından  $x_1[k] = x[2k+1]$  ve  $x_2[k] = x[2k]$  olur. Bunların her biri ardışık örnekler arasında ilintisi  $a^2$  olan bir AR dizisidir. Merkezi Kodçözücü için oran-bozunum fonksiyonu (2.3)'deki gibi olur.

$$D_{full}(R) = (1 - a^4)2^{-2R}, R \geq \log_2(1 + a^2) \text{ için} \quad (2.3)$$

Buradan sıkıştırma verimliliğindeki kayıp  $\frac{D_{full}(R)}{D(R)} = 1 + a^2$  dir.

Yan kodçözücüde çift numaralı örnekleri içeren tanımlamanın alındığı düşünülürse, kaybolan tanımlamadaki veriler alınan tanımlamadaki verilerin doğrusal aradeğerlemesi yapılarak kestirilir. Karesel ortalama aradeğerleme hatası [8]

$$\sigma^2 = E[(x_1(k) - \hat{x}_1(k))^2] \quad (2.4)$$

$$= E\left[\left(x_1(k) - \frac{x_2(k+1) + x_2(k)}{2}\right)^2\right] \quad (2.5)$$

$$= [R_x(0) - \frac{1}{2}R_x(0) - 2R_x(1) + \frac{1}{2}R_x(2)] \quad (2.6)$$

$$= (1 - a^2) + \frac{1}{2}(1 - a^2) + \omega D_q \quad (2.7)$$

Burada  $D_q$  ortalama karesel nicemleme hatası ve  $\omega \in [0,1]$  nicemleme hatasının ilintisidir. Yan kodçözücü için tek ve çift örnekleri içeren tanımlamaların ortalaması kullanılarak oran bozunum fonksiyonu (2.8) deki gibi elde edilir.

$$D_{half}(R) = \frac{1}{2} \left[ (1 - a^2) + \left(\frac{1}{2}\right)(1 - a^2) \right] + \frac{1 + \omega}{2} D_{full}(R) \quad (2.8)$$

$\omega = 1$  ve  $a = 0.98$  için elde edilen sonuçlar Şekil 2.3'de gösterilmiştir [1]. Düşey eksen bozunum  $D$ , yerine dB cinsinden SNR birimi ile göstermek değerlerin anlaşılması açısından kolaylık sağlar. SNR değeri hesabı  $-10 \log_{10} D$  şeklindedir.

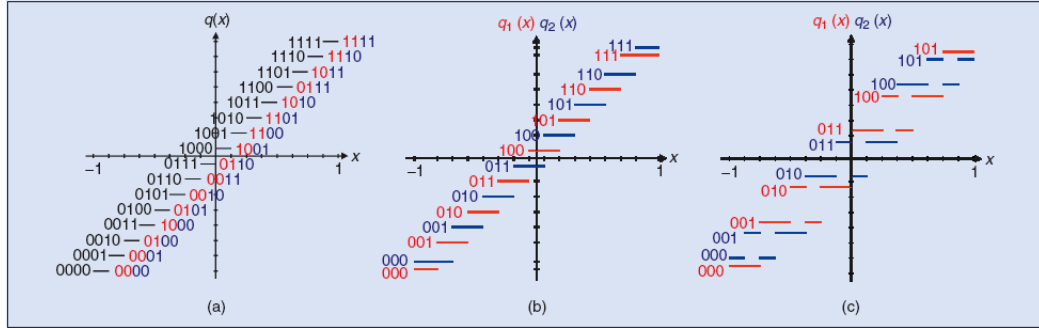
### 2.2.2 Kanal Ayırma için Skaler Nicemleme

Nicemleme, büyük veya sonsuz veri içeren kümenin daha küçük küme ile gösterilmesi işlemidir. Skaler nicemleme ise giriş verisi  $x$ 'in sonlu bir sayı olan çıkış verisi  $y$ 'ye haritalanmasıdır.  $Q: x \rightarrow y$  şeklinde gösterilir. Bir örnek vermek gerekirse tüm gerçel sayıların en yakın tam sayıya yuvarlanması işlemi basit bir skaler nicemlemedir.

Bir önceki bölümde, alınan örneklerin tek çift şeklinde ayrılarak kanala ayrıştırılmasından ve daha sonra bu tanımlamaların ayrı ayrı FDKM kodlayıcılar ile

kodlanmasından bahsedilmiştir. Bu kısımda ise tanımlamaların farklı nicemleyiciler kullanılarak elde edildiğinden bahsedilmiştir.

Şekil 2.4(a)'da veri değerlerinin  $x \in [-1,1]$  arasında olduğu ve bu değerlerin 4 bit ile birörnek (uniform) nicemlenmesi gösterilmektedir. Şekil 2.4 (b) ve (c)'deki grafiklerde farklı nicemleme durumları gösterilmektedir. Dikkat edilirse aynı zamanda (b) ve (c) durumlarında veriler üzerinde örtüşmeler ve bu örtüşmelerden dolayı tanımlamalarda artıklıklar oluşmaktadır. Kırmızı ve mavi renkler farklı birörnek nicemleyicilerdir. Böylece her bir nicemleyici ile elde edilen değerler



Şekil 2.4: Farklı nicemleme yapıları [1]

tanımlama olarak isimlendirilir. Örneğin  $q_1$  ve  $q_2$  nicemleyicileri kullanılarak oluşturulan değerler  $q_1(x) = 110$  1. tanımlama ve  $q_2(x) = 101$  2. tanımlama olarak isimlendirilir. Alıcıda alınan bu tanımlamalar birleştirilerek vericideki değer elde edilmek istendiğinde  $x$  değeri  $[\frac{7}{16}, \frac{9}{16}]$  sınırlarının orta noktası olacaktır. Bu değerde  $\frac{1}{2}$ 'dir.

Şekil 2.4(b)'deki nicemleyicinin tek başına veriyi oluşturmasına rağmen, zayıflığı vardır. Her iki kanaldan alınan tanımlamalar ile oluşturulan verinin kalitesi Şekil 2.4(a)'daki 4-bitlik nicemleyicinin kalitesi kadar olacaktır. Halbuki biz her iki tanımlamanın da gittiğini düşünürsek 6 bitlik veri göndermiş oluyoruz. Bu sorunu çözmek için Reudink toplam bit oranını fazla artırmadan ve kaynak dizideki önceden varolan artıklığa dayanmayan kanal ayırma tekniği önermiştir.

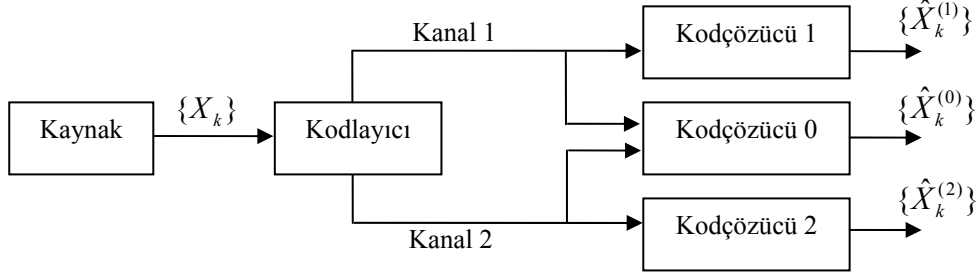
Şekil 2.4(c)'deki nicemleme grafiği Reudink'in tasarladığı nicemleme ailelerinden biridir. Tek başlarına bakıldığında kırmızı ve mavi nicemleyicilerin hücreleri arasında bir bağlantı yoktur. Fakat hep birlikte bakıldığında birbirlerinin tamamlayıcısı konumundadırlar. Örneğin  $q_1(x)=100$  olduğunda  $x$ 'in alabileceği değerler  $[\frac{1}{4}, \frac{3}{8}] \cup [\frac{1}{2}, \frac{3}{4}]$  ayrıca  $q_2(x)=100$  alındığında  $x$ 'in alabileceği değerler  $[\frac{1}{2}, \frac{5}{8}]$ . Şekil 2.4(c)'deki her bir nicemleyicinin 6 çıkış değeri vardır. Nicemleyicinin çıkışlarının hepsi çağrıldığında  $\log_2 6 \approx 2.6$  bit oranına sahip nicemleyici elde edilir. Böylece Şekil 2.4(c)'deki nicemleyiciyi kullanarak Şekil 2.4(b)'deki nicemleyicinin toplam bit oranından daha düşük bit oranına sahip bir nicemleyici gerçekleştirilir.

### 2.3 Oran Bozunum Bölgesi

Gersho, Jayant, Quirk ve Reudink ses için kanal ayırma üzerine çalıştılar. Witsenhausen kanal ayırmanın bilgi teorisinde problemlere yol açtığı farkına vardı. Yapılan bu pratik çalışmalar bilgi teorisi literatüründe yeni bir probleme esin kaynağı oldu. 1979 Eylül'ünde bilgi teorisi çalıştayında Wyner, Witsenhausen, Wolf ve Ziv ile birlikte elde ettiği sonuçları yorumlayarak problemi ortaya attı. Ortaya matematiği sonradan bulunacak olan şu problem çıkmıştı. Bir bilgi kaynağının 2 ayrı tanımlama ile tanımlandığında ayrı ve ortak olarak bu tanımlamalar alındığında asıl veriye göre kalite ne olacak ve sınırları nasıl belirlenecek? İşte bu problem Çoklu Tanımlama (ÇT) problemi olarak anılmaktadır.

2 tanımlamalı ÇT durumu şematik olarak Şekil 2.5'de gösterilmiştir. Kodlayıcı,  $\{X_k\}_{k=1}^N$  sembolleri ile gösterilen kaynak dizisini 2 gürültüsüz kanal üzerinden 3 alıcıya gönderir. Kodçözücü (merkezi kodçözücü) her iki kanaldan gönderilen bilgileri alırken kalan 2 kodçözücü (yan kodçözücüler) sadece kendi kanalları üzerinden gönderilen tanımlamaları alır. Kanal  $i$  üzerinden iletim oranı  $R_i$ ,  $i=1,2$  kaynak örnek başına bit olarak ifade edilir. Kodçözücü  $i$  tarafından üretilen geri elde

edilmiş veri dizisi  $\{\hat{X}_k^{(i)}\}_{k=1}^N$  şeklindedir. Geri elde edilen verilerin asılları ile olan bozunum  $D_i$ ,  $i = 0,1,2$  sembolleri ile gösterilir.



Şekil 2.5: İki tanımlamalı ÇTK yapısı

Kalite ve bit oranı arasında daima bir ödünleşim (tradeoff) vardır. Bilgi teorisinin bir dalı olan oran bozunum teorisi ulaşılabilir oran ve bozunumların sınırlarını belirler. Oran-bozunum sınırları bit oranına bağlı olarak kalitenin nasıl değiştiği ile ilgili açıklayıcı bilgiler verir. Burada verilecek olan bilgiler [9]'den alınmıştır. Daha detaylı incelemeler için [7,10] referanslarına bakılabilir.

Kaynağın bağımsız ve eş dağılıma (independent and identically distributed) sahip, gerçek rastlantı değişkenleri  $X_1, X_2, \dots, X_n$  dizisi ürettiği varsayalım. Bozunum ölçüsü  $d$ , negatif olmayan değer verir. Kaynak veriyi  $x$ , alıcıda geri elde edilmiş  $x$  değeri ise  $\hat{x}$  ile gösterilirse bozunum  $d(x, \hat{x})$  ile ifade edilir.  $x^{(n)} = (x_1, x_2, \dots, x_n)$  ve  $\hat{x}^{(n)} = (\hat{x}_1, \hat{x}_2, \dots, \hat{x}_n)$  dizileri arasındaki bozunum (2.9) ile hesaplanır.

$$d(x^{(n)}, \hat{x}^{(n)}) = \frac{1}{n} \sum_{i=1}^n d(x_i, \hat{x}_i) \quad (2.9)$$

$$d(x, \hat{x}) = (x - \hat{x})^2 \quad (2.10)$$

En yaygın bozunum ölçüsü (2.10) bağıntısı ile verilen karesel hatadır.  $n$  uzunluğuna sahip diziler arasındaki karesel hata bozunumu, dizilerin her bir elemanın farklarının karelerinin toplamının dizi uzunluğuna bölümüyle hesaplanır.

Kaynak kod, kodlanacak veri dizisindeki verileri yaklaşık olarak temsil eden bir yapıdır. Elimizde  $n$  uzunluğunda  $x^{(n)}$  dizisi olsun. Bu diziyi kaynak kodlamak istiyoruz. Kaynak kodlama oranı  $R$  olsun. Kodlayıcı  $\alpha$ , kodçözücü  $\beta$  ile gösterilsin. Kod ile ilişkili olan bozunum kaynak dizisi ile kaynağın kodlanıp kodçözücüde çözülmesi sonucu elde edilen dizi arasındaki bozunum ölçüsünün beklendik değeri (expected value) (2.11)'deki gibi tanımlanır.

$$D = E[d(X^{(n)}, \beta(\alpha(X^{(n)})))] \quad (2.11)$$

Elimizde oran  $R$ , bozunum  $D$  ve kaynak kod uzunluğu mevcut ise oran bozunum çifti oluşturulabilir. Ulaşılabilir oran bozunum çiftleri kümesinin düzlemde ikiye ayırdığı bölgelerden üst kısım, oran bozunum bölgesi ( $RD$ ) olarak isimlendirilir. Oran bozunum fonksiyonu, oran bozunum bölgesindeki en küçük oranı veren değerleri ifade eder. Bu sebeple oran bozunum fonksiyonu teorik sınırı belirler ve bu sınır değerler altında sonuç elde edilemez. Bu değerler kaynak çeşidine ve bozunum ölçüsüne göre değişmektedir. İstenilen oran değeri  $R$  kullanılarak, oluşacak bozunum  $D$ 'yi en küçültecek kaynak kod uzunluğu  $n$ 'yi belirlemek zordur. Çünkü kaynak kodlar genellikle sabit-oran vektör nicemleyicilerdir. Vektör nicemleyicileri en iyileştirmenin zorluğu [11,12,13]'de tartışılmıştır.

Oran bozunum fonksiyonu problemi skaler koşullu yoğunluk (scalar conditional density) için kısıtlı en küçültme problemine dönüşür [7,14]. Bu en küçültme problemi bazı kaynaklar ve bozunum ölçüleri için çözülmüştür. Örneğin, değişintisi  $\sigma^2$  olan Gauss kaynaklarının karesel hata bozunumu için oran bozunum fonksiyonu  $D(R) = \sigma^2 2^{-2R}$  şeklindedir. Sadece bir örnek olmasına rağmen tüm sürekli değerli kaynakların oran-bozunum fonksiyonlarının sınırlarını gösterir: olasılık yoğunluğu  $f(x)$  ve değişintisi  $\sigma^2$  olan bir kaynak için karesel hata (2.12) denklemi ile belirlenir. Burada  $h = -\int f(x) \log_2 f(x) dx$  farksal entropidir.

$$\frac{1}{2\pi e} 2^{2h} 2^{-2R} \leq D(R) \leq \sigma^2 2^{-2R} \quad (2.12)$$

## 2.4 Çoklu Tanımlamalı Oran Bozunum Bölgesi

Kanal 1 üzerinden  $R_1$  ve kanal 2 üzerinden  $R_2$  bit oranına sahip tanımlamaların gönderileceği bir sistem tasarlandığında  $R_1$  ve  $R_2$  bit oranlarında üretilen tanımlamalar birbirlerine çok benzerdir, bu yüzden merkezi kodçözücüde birleştirildiklerinde sadece bir tanesinin kullanılmasıyla elde edildenden fazla bir avantajı olmayacaktır. Tanımlamaların sadece kendilerinin geri elde edilmeleri sonucu kalite kabul edilebilir olacaktır, birbirine çok benzer olmayacaktır. Bu da ÇTK'nın tanımlamalar arasındaki ödünleşiminden kaynaklanmaktadır.

ÇTK modeli, oran bozunum teorisinde ve pratik sıkıştırma durumlarında birçok probleme sebep olur. Temel problem, verilen bir kaynak ve bozunum ölçüsü için  $(R_1, R_2, D_0, D_1, D_2)$  sınırlarının ne olacağına karar vermektir.

Çoklu tanımlamalı oran bozunum bölgesi veya ÇT bölgesi, belirli bir kaynak ve bozunum ölçüsü için ÇT kodlamada aynı zamanda ulaşılabilecek oran ve bozunumlar kümesinin iç kısmıdır. Yukarıda bahsedilen oran bozunum bölgesi tek tanımlama ile elde edilen bölgedir. Sadece  $R$  ve  $D$  değerlerinin bulunması yeterlidir. Fakat iki tanımlamalı durum için  $(R_1, R_2, D_0, D_1, D_2)$  sınırlarının ne olacağı bilinmelidir.

El Gamal ve Cover, kaynağı ve geri üretileni temsil eden rastlantı değişkenlerinin ortak dağılımından  $(R_1, R_2, D_0, D_1, D_2)$  değerlerine nasıl karar verileceğini gösterdiler [15]. Bu yöntemle elde edilen noktalar tüm  $(R_1, R_2, D_0, D_1, D_2)$  değerler için elde edilemiyordu. Bu yüzden ulaşılabilen noktaları kapsayan iç sınır bölgesine ÇT bölge ismi verildi. Bilgi teorisi açısından ulaşılabilen noktaların diğer karakteristikleri [16,17]'deki kaynaklarda verilmiştir. [17]'deki kaynakta, [15,16]'deki 2 tanımlama için elde edilen bölgelerin daha fazla tanımlamaya genelleştirilmesi sonucu oluşturulan bölge gösterilmiştir.



ÇT bölgesinde olmayan noktalar dışarı sınır olarak adlandırılır. Dışarı sınır oran bozunum fonksiyonu kullanılarak elde edilir.  $D_1$  bozunumuna sahip kodçözücü 1 en azından sembol başına (2.14)' de gösterildiği gibi,  $R(D_1)$  bit almak zorundadır.

$$R_1 + R_2 \geq R(D_0) \quad (2.13)$$

$$R_i \geq R(D_i) \quad i = 1, 2 \quad (2.14)$$

ÇT bölgesi yalnızca belleksiz Gauss kaynak ve karesel hata bozunumu ölçüsü için tanımlanmıştır. Bu kaynak ve ölçü için Ozarow, ÇT bölgesinin büyük bir küme olduğunu göstermiştir [18]. Bu durum ÇT kodlamada sınırların belirlenmesinde yeterlidir. Böylece herhangi bir sürekli-değerli belleksiz kaynak ve karesel hata bozunum ölçüsü baz alınarak oluşturulacak ÇT bölgesi için Gauss kaynağı kullanılarak bölge sınırlandırılabilir [19].

Değişintisi  $\sigma^2$  olan belleksiz Gauss kaynağı için  $(R_1, R_2, D_0, D_1, D_2)$  'den oluşan ÇT bölge (2.15) ve (2.16) 'daki eşitlikler kullanılarak elde edilir.

$$D_i \geq \sigma^2 2^{-2R_i} \quad i = 1, 2 \quad (2.15)$$

$$D_0 \geq \sigma^2 2^{-2(R_1+R_2)} \cdot \gamma_D(R_1, R_2, D_1, D_2) \quad (2.16)$$

Burada eğer (2.17) ve (2.18) bağıntıları kullanıldığında  $\gamma_D = 1$  olur.

$$D_1 + D_2 > \sigma^2 + D_0 \quad (2.17)$$

$$\gamma_D = \frac{1}{1 - \left( \sqrt{(1-D_1)(1-D_2)} - \sqrt{D_1 D_2 - 2^{-2(R_1+R_2)}} \right)^2} \quad (2.18)$$

Dengeli durumda yani  $R_1 = R_2$  ve  $D_1 = D_2$ , Ozarow'un sonuçları birim değışintili kaynak için kenar bozunum sınırını kanıtlar [2].

$$D_1 \geq \min \left\{ \frac{1}{2} \left[ 1 + D_0 - (1 - D_0) \sqrt{\frac{1 - 2^{-2(R_1 + R_2)}}{D_0}} \right], 1 - \sqrt{\frac{1 - 2^{-2(R_1 + R_2)}}{D_0}} \right\} \quad (2.19)$$

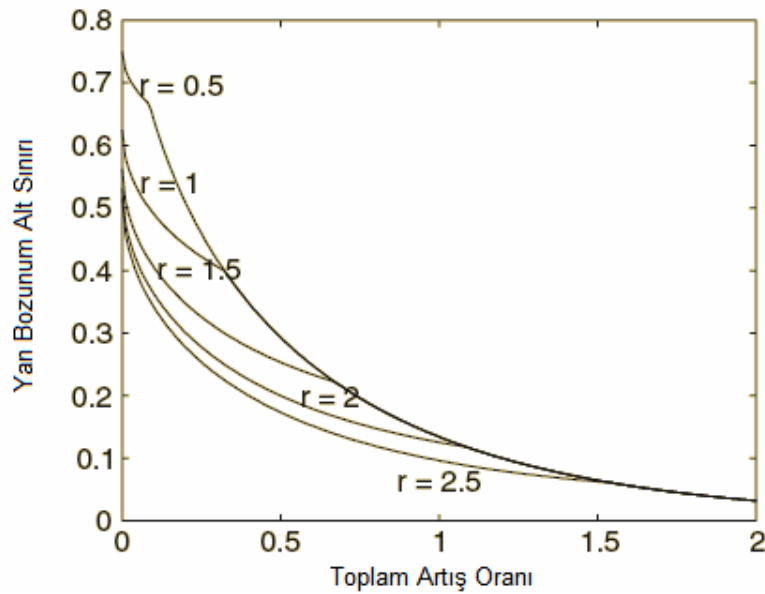
Denklem (2.19)  $r = R(D_0)$  ve  $\rho = R_1 + R_2 - R(D_0)$  cinsinden yazıldığında,

$$D_1 \geq \begin{cases} \frac{1}{2} \left[ 1 + 2^{-2r} - (1 - 2^{-2r}) \sqrt{1 - 2^{-2\rho}} \right] & \rho \leq r - 1 + \log_2(1 + 2^{-2r}) \\ 1 - \sqrt{1 - 2^{-2\rho}} & \rho > r - 1 + \log_2(1 + 2^{-2r}) \end{cases} \quad (2.20)$$

(2.20) deki ifadenin farklı  $r$  değerleri için gösterimi Şekil 2.6'da verilmiştir. Düşük artıklığa sahip  $D_1$ 'in eğimi,

$$\frac{\partial D_1}{\partial \rho} = -\frac{1 - 2^{-2r}}{2} \frac{2^{-2\rho} \ln 2}{\sqrt{1 - 2^{-2\rho}}} \quad (2.21)$$

(2.21)'de gösterildiği gibidir.  $\rho = 0^+$ 'da eğim sonsuzdur. Sistemin merkezi kodçözücü için oran bozunum sınırına ulaşmaya başladığını ve oranın küçük küçük arttığını düşünürsek, sonsuz eğimin anlamı, küçük ek oranların kenar bozunum azaltımını, merkezi bozunum azaltımından daha fazla etkilemesidir.



Şekil 2.6: Farklı  $r$  değerleri için oran bozunum sınırı [1]

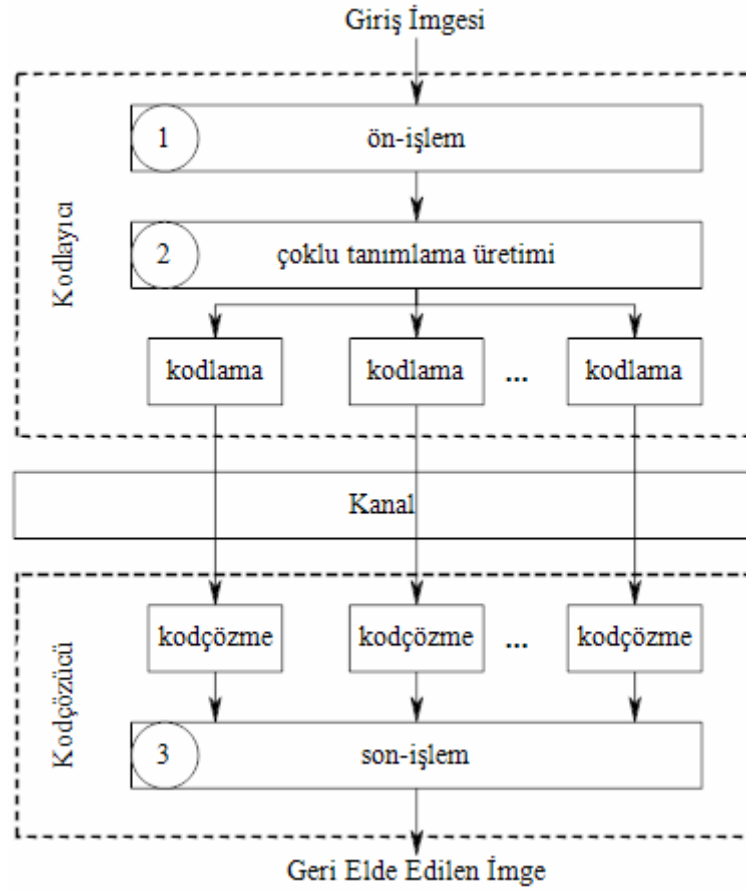
### **3. ÇOKLU TANIMLAMALI KODLAMADA KULLANILAN YÖNTEMLER**

#### **3.1 Giriş**

Güvenilir olmayan ağlar üzerinden ÇTK tekniği kullanılarak iletilecek veri farklı yöntemler kullanılarak tanımlamalara ayrılabilir. Öngörü dengeleme (prediction compensation), öznelik yönlü (Feature-oriented), hareket dengelemeli zamansal kestirim gibi yöntemlerin ÇTK ile birleştirilmiş durumları mevcuttur [20,21,22]. Bu tez kapsamında çok aşamalı aşağı örnekleme (polyphase downsampling based multiple description), çoklu tanımlamalı skaler nicemleme (multiple description scaler quantization) olmak üzere iki yöntem ele alınmıştır. Bu sebepten dolayı sadece bu yöntemlerden bahsedilecektir.

#### **3.2 Çok Aşamalı Aşağı Örnekleme Temelli Çoklu Tanımlamalı Kodlama**

Çoklu tanımlamalı kodlamada kullanılan yöntemlerden birisi iki farklı tipi olan “çok aşamalı aşağı örnekleme temelli ÇTK”dır [23]. İlk yöntem giriş verilerinin aşağı örneklemeden sonra farklı nicemleyiciler ile nicemlenerek tanımlamaların oluşturulmasıdır [24,10]. Diğer kullanılan yöntemde ise veri dizisi, Şekil 3.1’de görüldüğü gibi üç aşamalı işlemde geçerek tanımlamalar oluşturulmaktadır. Bu aşamalar, ön-işlem, çoklu tanımlama üretimi ve son-işlemdir. Ön-işlem kısmında artıklık eklendikten sonra çoklu tanımlama üretimi kısmında tanımlamalar oluşturulur. Kodlama kısmında farklı kodlama teknikleri kullanılabilir. Kanaldan alıcıya ulaşan tanımlamalar kodçözme işleminden geçirilerek birleştirilir. Ön-işlem kısmında eklenen artıklık son-işlem kısmında çıkartılır. Sonuçta giriş verisine yakın bir veri geri elde edilmiş olur. Yöntemin avantajlarından birincisi, oluşturulan tanımlamaların kanaldaki hatalara olan dayanıklılığını artırmaktır. İkincisi ilk ve son işlem şeklinde olduğundan ÇTK yapısına uygulanması kolaydır.

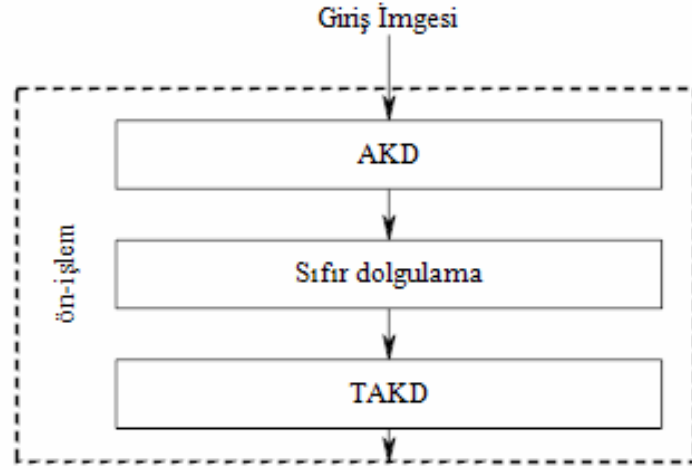


Şekil 3.1: ÇTK tabanlı aşırı örneklemenin blok gösterimi [4]

### 3.2.1 Ön-İşlem

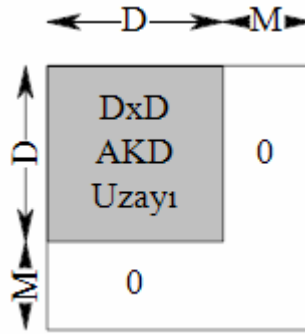
Bu aşamada, giriş imgesine artıklık eklenir [25]. [26]'daki çalışma da genel yapı itibari ile [25]'dekine benzerdir. Önerilen yöntem temel olarak yönlü sıfır dolgulamadır. Bu işlem her iki yöne değil sadece seçilen yöne yapılmaktadır.

Şekil 3.2'de ön-işlem kısmında yapılan dönüşüm alma, artıklık ekleme, ters dönüşüm alma işlemleri gösterilmiştir. İlk olarak giriş verileri ayrık kosinüs dönüşümünden (AKD) geçirilerek uzamsal uzaydan frekans uzayına dönüştürülür. Artıklık ekleme işlemi frekans uzayında veriye sıfır eklemektir. Böylece oluşturulacak tanımlamalar arasındaki ilişki artırılır. Artıklık eklendikten sonra ters ayrık kosinüs dönüşümü alınarak tekrardan uzamsal uzaya geçilir.



Şekil 3.2: Ön-işlem bloğu içerisindeki işlemler [4]

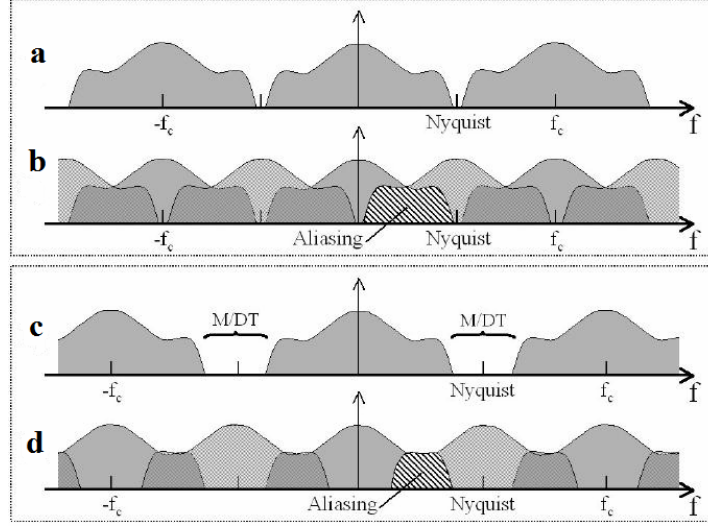
Artıklık ekleme işleminin ayrıntılı gösterimi Şekil 3.3’de verilmiştir.  $D \times D$  boyutundaki giriş imgesine sıfır ekleme parametresi  $M$  değerine bağlı olarak frekans uzayında her iki yönde sıfır eklenir. Burada  $M$  değeri aşırı örnekleme oranı (AÖO) sonucu elde edilen sıfır dolgulama değeridir. Aşırı örnekleme oranı giriş imgesi ile sıfır dolgulanan imge boyutları arasındaki oran değeridir. Böylece ön-işlemden geçirilen ve aşırı örneklenmiş duruma getirilen imgenin son boyutu  $(D+M) \times (D+M)$  olacaktır.



Şekil 3.3: Frekans uzayında artıklık ekleme gösterimi [4]

Aşırı örnekleme işlemi frekans uzayında ele alınırsa, imgeyi geri çatarken gerekli olan aşağı örnekleme işleminin neden olacağı örtüşme etkisini azaltacağı açıktır. Bu durum Şekil 3.4’te tek boyutlu işaretler için gösterilmiştir. İşarete  $M = 0$  dolgulama yapıldığında a’da işaretin spektrumu görülmektedir.  $M > 0$  uygulandığında c işareti

görülmektedir. b ve d ise a ve b'deki işaretlerin aşağı örneklenmiş hallerini göstermektedir. Görüldüğü üzere  $M$  değeri arttıkça örtüşme etkisi azalmaktadır.



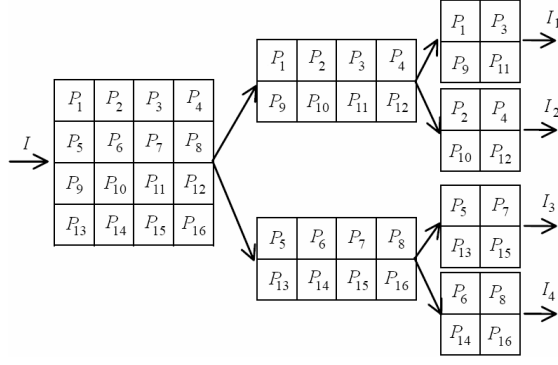
Şekil 3.4: Frekans uzayında örtüşme [4]

### 3.2.2 Çoklu Tanımlama Üretimi

Giriş imgesine artıklık eklendikten sonra uzamsal uzaya tekrar geçilir. Daha sonra tanımlamalar üretilmelidir. Literatürde genel olarak sonuçlar 2 veya 4 tanımlama üzerinden verilmiştir. Bu nedenden dolayı yöntemin açıklanması için 4 tanımlama kullanılmıştır. Tanımlamalar (3.1)'de görüldüğü üzere aşağı alt örnekleme kullanılarak elde edilir.

$$\begin{aligned}
 I_1(i, j) &= I(2i-1, 2j-1) \\
 I_2(i, j) &= I(2i, 2j-1) \\
 I_3(i, j) &= I(2i-1, 2j) \\
 I_4(i, j) &= I(2i, 2j)
 \end{aligned}
 \tag{3.1}$$

Komşu pikseller arasında artıklık bulunmasından dolayı her bir piksel giriş imgesinin temel özelliğini kendi içinde barındırır. Elde edilen tanımlamalar istenilen sıkıştırma algoritmaları kullanılarak sıkıştırılır ve kanal ortamına iletilir.



Şekil 3.5: Aşağı örnekleme yolu ile ÇTK [27]

Sıkıştırma algoritmaları olarak SPIHT, JPEG, JPEG2000 kullanılabilir. Eğer bu yöntem videoya uygulanacaksa kodlama kısmında H.263 veya H.264/AVC gibi standartları ile kodlama yapılabilir.

### 3.2.3 Son-işlem

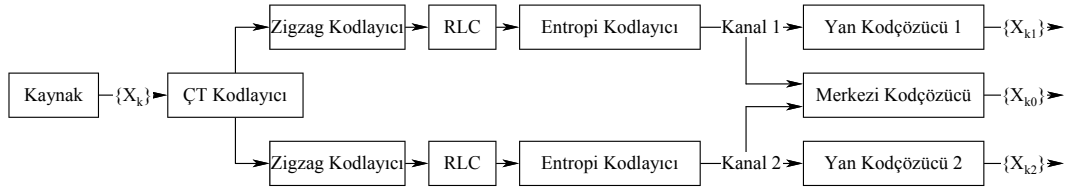
Son-işlem kısmı alıcıda gerçekleşir. Kanaldan gelerek alıcıya ulaşan tanımlamalar kodçözücüde çözülerek birleştirilirler. Birleştirilme işleminde eğer tüm tanımlamalar geldiyse bir sorun olmaz. Fakat kanaldaki gürültüler veya ağ üzerindeki bazı aygıtların tampon belleklerinin taşmasından kaynaklı paket kayıpları durumunda kayıp paketler üst örnekleme kullanılarak alıcıya ulaşan tanımlamalardan kestirilir. Üst örnekleme işleminde aradeğerleme yöntemi kullanılmıştır. Elde edilen tanımlamalar birleştirilerek elde edilen imge, ön-işlem kısmındaki artıklık eklenmiş imgedir. Bu sebeple ön-işlemdeki işlemlere benzer işlemler uygulanır. İlk olarak AKD alınarak frekans uzayında eklenen artıklık çıkarılır. TAKD alınarak uzamsal uzaya geçilir. Böylece giriş verisi elde edilmiş olur.

### 3.3 Çoklu Tanımlamalı Skaler Nicemleyici

Çoklu Tanımlamalı Skaler Nicemleme (ÇTSN), Vaishampayan tarafından önerilen ve genellikle dönüşüm katsayılarına uygulanan bir ÇTK yaklaşımıdır [29]. ÇTSN kullanılarak tanımlamalar arasındaki ilişki istenildiği gibi ayarlanabilir. Bu ilişki merkezi ve yan kodçözücülerde tanımlamalar kullanılırken ortaya çıkar. Bu

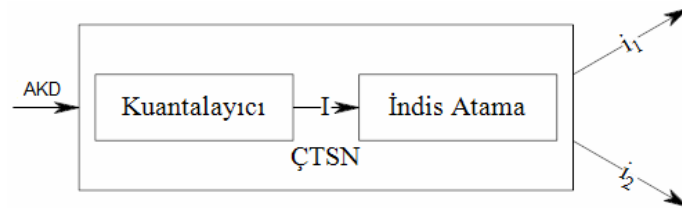
yaklaşımında da merkezi ve yan kodçözücüler arasında daima bir ödünleşim vardır [28,29].

ÇTSN kodlama yaklaşımının AKD için uygulanması durumunda oluşan kodlama sisteminin genel yapısı Şekil 3.6’da verilmiştir. ÇT oluşturma işlemi ÇT kodlayıcı içerisinde gerçekleştirilmektedir. Kaynaktan alınan veri dizilerine kodlayıcı blok içerisinde AKD ve nicemleme işlemleri uygulanır. Bu sistem yardımıyla iki ayrı skaler nicemleme kullanılarak iki farklı tanımlama dizileri elde edilir. Kodçözücüye gelen tanımlamaları çözmek için yan kodçözücülerin yanı sıra tanımlamaların her ikisi geldiğinde kodçözme işleminin birlikte yapılmasını sağlayan merkezi kodçözücü de eklenmiştir.



Şekil 3.6: ÇTSN kodlama sisteminin genel yapısı

ÇTSN kodlayıcısı bir skaler nicemleyici ve indis atama işleminden oluşur. ÇTSN kodlayıcı, her bir alınan AKD örneğinden nicemleme sonucu, indis matrisini de kullanarak Şekil 3.7’de gösterildiği gibi  $i_1$  ve  $i_2$  olmak üzere iki adet indis değeri üretir.



Şekil 3.7: İndis değeri üretme [32]

Üretilen indis değerlerinin kod vektörleri doğal bir şekilde dizilmezler ve kodlama karmaşıklığına sebep olurlar. Bu tür zorluklardan sakınan bir yöntem Servetto tarafından önerilen MD kafes vektör nicemleyicisidir (MDLVQ) [30]. Bu kaynakta



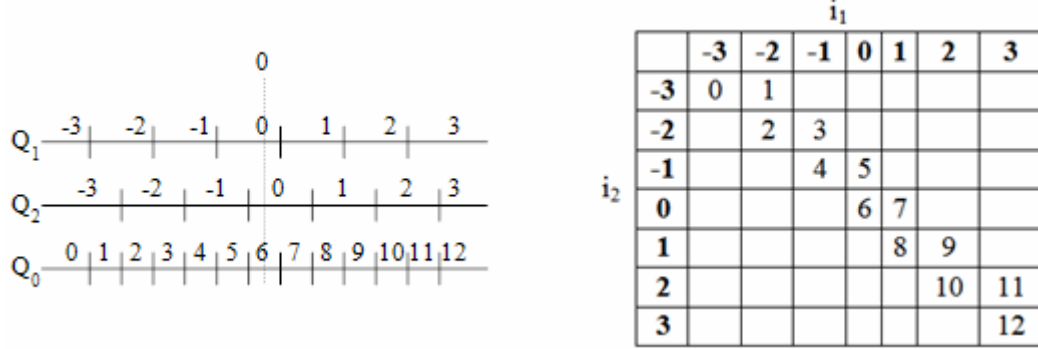
indis atama problemi kafes simetrisi kullanılarak basitleştirilmiştir. Ayrıca kafes yapısı kodlama karmaşıklığını azaltmıştır [31].

Nicemleyici, her bir AKD değerinden nicemlenmiş veri  $l$ 'yi ( $R \rightarrow l$ ) üretir. İndis atama bloğu,  $l$  değerini alarak içerisindeki indis harita matrisine göre  $(i_1, i_2)$  indis çiftini elde eder. Bu elde edilen değerler Zigzag, RLC (Run-Length Coding-RLC) ve entropi kodlama bloklarından geçerler. İki farklı kanal üzerinden alıcıya gönderilirler. ÇTSN  $i_1$  ve  $i_2$  indislerini seçecek  $f_1 : l \rightarrow i_1$  ve  $f_2 : l \rightarrow i_2$  iki yan kodlayıcı, alıcıda alınan tanımlamaları geri elde etmek için kullanılan  $g_0 : (i_1, i_2) \rightarrow l$ ,  $g_1 : i_1 \rightarrow l$  ve  $g_2 : i_2 \rightarrow l$  üç adet kodçözücü içerir. Merkezi kodçözücü  $g$ ,  $l$  değerini  $g_1$  ve  $g_2$ 'den daha iyi kestirir.

### 3.3.1 Nicemleme ve İndis Haritası Oluşturma

İndis atama işlemi ÇTSN yapısının önemli bir kısmını oluşturur. En uygun indis atama işlemi kesin bir çözüme ulaşmayıp, karmaşık işlemler gerektirdiğinden zordur [1]. Vaishampayan makalesinde en uygun başarıma yakın sonuçların üretildiği birçok yeni yöntem önermiştir [28]. Önerdiği temel fikir, indis haritasındaki hücre (cell) değerlerini Şekil 3.8'de görüldüğü gibi indis haritasına sol üstten sağ aşağıya doğru bir köşegen şeklinde dizmektir. Burada  $Q_1$  ve  $Q_2$  iki ayrı nicemleyicidir. Örneğin  $-1.8$  değerinin iletileceğini varsayalım.  $-1.8$  değeri  $Q_0$  daki hücre değerlerinden 3 numaralı hücre içerisine denk gelecektir. İndis haritasından 3 numaralı hücre değerine bakıldığında 1. tanımlamada  $-1$  ikinci tanımlamada  $-2$  gönderileceği görülmektedir. Yapılan işlem  $-1.8$  değerini 1. nicemleyici  $-1$  olarak kodlamakta 2. nicemleyici  $-2$  olarak kodlamaktan ibarettir. Alıcıda her iki tanımlamada alındığı durumda  $Q_0$  daki 3 numaralı hücrenin tam orta değeri alınır. Elde edilen değer  $-1.75$  dir. Çünkü  $Q_0$  hücre boyutu  $0.5$  birimdir. Bundan dolayı 3 numaralı hücrenin orta değeri  $-1.75$  yapmaktadır. Sadece 1. tanımlamanın geldiği düşünülürse, indis haritasından bakıldığında  $-1$  numaralı sütunda 3 ve 4 numaralı hücreler bulunmaktadır ama hangisi olduğu 2. tanımlama gelmediği için bilinmemektedir. Bu sebeple sonuç 3 ve 4 numaralı hücre değerlerinin ortası olan  $-1.5$  olacaktır. Aynı şekilde 2. tanımlamanın geldiği düşünülürse,  $-2$  numaralı satıra

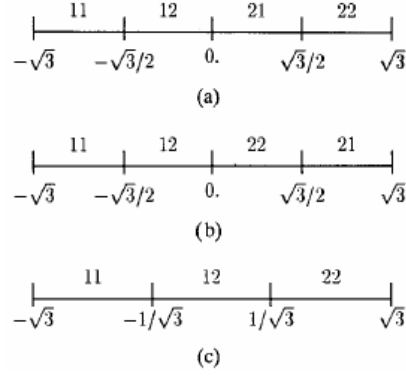
bakılarak 2 ve 3 numaralı hücrelerin ortası geri oluşturulacak -2 değerini vermektedir. Hata değerlerine bakıldığında her iki tanımlama alındığında, sadece 1. tanımlama ve sadece 2. tanımlama alındığında oluşacak hata değerleri sırasıyla 0.05, 0.3 ve 0.2 olacaktır.



Şekil 3.8: İndis haritası oluşturma [34]

Nicemleme ve indis haritası arasındaki ilişkiyi daha iyi görebilmek açısından nicemleme işleminden sonra bu işlem sonucu elde edilen hücrelerin, indis haritasına dizilişi Şekil 3.8’de gösterilmiştir [33].

İndis oluşturma işlemi Vaishampayan’ın makalesinde [28] aşağıda verilen daha genel bir örnek üzerinden anlatılmıştır. Elimizdeki kaynak verinin  $(-\sqrt{3}, \sqrt{3})$  aralığında düzgün dağılımlı ve bit oranı değerleri her iki kanal için de  $R_1 = R_2 = 1$  olduğu varsayalım. Eşit aralıklı nicemleme yapan nicemleyici seçilirse hücre uzunluklarının eşit olması gerektiğinden kaynak verideki veri miktarının en küçük ve en büyük değerleri arası, kaç adet indis çifti var ise ona bölünür. Böylece her bir hücrenin ne kadarlık bir alanı kapsadığı bulunur. Şekil 3.9’da görüldüğü gibi her bir durumda (1,1), (1,2), (2,1), (2,2) olan 4 farklı kodsözcüğü (codeword) iletim kanalından gönderilir.



Şekil 3.9: Farklı indis oluşturma [28]

Tablo 3.1: Tanımlamaların birleştirilmesi sonucu elde edilen değerler [28]

	$\hat{x}_{11}^0$	$\hat{x}_{12}^0$	$\hat{x}_{21}^0$	$\hat{x}_{22}^0$	$\hat{x}_1^1$	$\hat{x}_2^1$	$\hat{x}_1^2$	$\hat{x}_2^2$
(a)	$-\frac{3\sqrt{3}}{4}$	$-\frac{\sqrt{3}}{4}$	$\frac{\sqrt{3}}{4}$	$\frac{3\sqrt{3}}{4}$	$-\frac{\sqrt{3}}{2}$	$\frac{\sqrt{3}}{2}$	$-\frac{\sqrt{3}}{4}$	$\frac{\sqrt{3}}{4}$
(b)	$-\frac{3\sqrt{3}}{4}$	$-\frac{\sqrt{3}}{4}$	$\frac{\sqrt{3}}{4}$	$\frac{3\sqrt{3}}{4}$	$-\frac{\sqrt{3}}{2}$	$\frac{\sqrt{3}}{2}$	0.0	0.0
(c)	$-\frac{2}{\sqrt{3}}$	0.0	---	$\frac{2}{\sqrt{3}}$	$-\frac{1}{\sqrt{3}}$	$\frac{2}{\sqrt{3}}$	$-\frac{2}{\sqrt{3}}$	$\frac{1}{\sqrt{3}}$

Alıcıda alınan tanımlamaların içerisindeki indis değerleri birleştirilerek Tablo 3.1’de görülen değerler elde edilir. Burada  $i = 1,2$  olmak üzere  $\hat{x}_{ii}^0$  merkezi kodçözücüde geri elde edilen verileri,  $\hat{x}_i^1$  yan kodçözücü 1 ve  $\hat{x}_i^2$  yan kodçözücü 2’yi göstermektedir. Örneğin nicemleme sonrası elde edilen değer  $(-\frac{\sqrt{3}}{2}, 0)$  aralığında ise, indis haritasında birinci tanımlamada 1 ikinci tanımlamada 2 değeri olacaktır. Alıcıda tüm tanımlamalar alındığında merkezi kodçözücüde (1,2) için o hücrenin orta değeri, eğer sadece birinci kanaldan sadece 1 geldi ise ikinci kanaldan ne geldiğini bilmediğimizden bir de olabilir 2 de olabilir. Bu sebeple hücre sınırı (1,1) ve (1,2) ’nin kesişimi  $(-\sqrt{3}, 0)$  olacaktır. Üretilecek değerde bu hücrenin ortası olan  $-\frac{\sqrt{3}}{2}$  dir.

Tablo 3.2: Tablo 3.1’de verilen değerler için merkezi ve yan en küçük karesel hata değerleri [28]

	$E(d_0)$	$E(d_1)$	$E(d_2)$
(a)	$\frac{1}{16}$	$\frac{1}{4}$	$\frac{13}{16}$
(b)	$\frac{1}{16}$	$\frac{1}{4}$	1
(c)	$\frac{1}{9}$	$\frac{1}{3}$	$\frac{1}{3}$

Tablo 3.2’de  $d_0$ ,  $d_1$  ve  $d_2$  sırasıyla merkezi ve yan tanımlamalara ait en küçük karesel hata değerini göstermektedir. Görüldüğü üzere (b) durumunun (a) durumundan daha başarılı olduğu görülmektedir. Fakat tanımlamalar arasında bir denge yoktur. Şöyle ki bir tanımlama sonucu en küçük karesel hata (MSE) değeri 1 iken diğerinin değeri 0.25 olmaktadır. Bu da alıcıda alınan tanımlamanın değişmesine göre bir başarımla elde edileceği anlamına gelir. Oysa tanımlamalar arasında bir tutarlılık bulunmalıdır. Değerleri birbirine yakın veyahut aynı olmalıdır. İşte bu tip tanımlamalar oluşturulursa, bu tanımlamalara dengeli (balanced) tanımlama denir. (c) durumunda dengeli tanımlamalar sonucu MSE değerleri görülmektedir. (b) değerinde hata değeri 0.25 olmasına karşın diğer tanımlama alındığında hata 1 olmaktadır. Oysa dengeli tanımlama durumunda hangi tanımlama alınır alınsın hata 0.33 şeklinde olacaktır. Fakat Tablo 3.1’de görüldüğü üzere dengeli tanımlamalar oluşturulması için (2,1) çifti kullanılmamıştır. Bu anlatılanlardan görüldüğü üzere yan ve merkezi kodçözücü sonucu elde edilen hata değerlerini belirleyen indis atama işlevidir.

Vaishampayan tarafından doğrusal indis ataması (linear index assignment) ve iç içe indis ataması (nested index assignment) isimlerinde iki farklı indis haritası oluşturma yöntemi önerilmiştir [28]. Her iki yöntemde de oluşturulacak hücreler,  $k$  köşegen sayısı olmak üzere, indis matrisindeki ana köşegene ve bu köşegene  $2k$  uzaklıktaki bölgeye, seçilen tarama yöntemine göre ardışık olarak dizilir.

- İç içe indis atama: Tarama işlemi ana köşegenden başlayarak doğu ve güney yönünde olur. Başlanacak ilk hücre  $(i, i)$  alındığında, doğu yönündeki tarama  $(i, i), (i, i+1), (i+1, i), (i, i+2), (i+2, i), \dots, (i, i+k), (i+k, i)$  ve güney yönündeki tarama  $(i, i), (i+1, i), (i, i+1), (i+2, i), (i, i+2), \dots, (i+k, i), (i, i+k)$  şeklindedir. Şekil 3.10'da bu tarama yöntemine uygun üretilmiş indis matrisleri görülmektedir.

	1	2	3	4	5	6	7	8
1	1	2	4					
2	3	6	7	9				
3	5	8	11	12	14			
4		10	13	16	17	19		
5			15	18	21	22	24	
6				20	23	26	27	29
7					25	28	31	32
8						30	33	34

(a) İç içe indis atama  $k=2$  doğu

	1	2	3	4	5	6	7	8
1	1	3	5					
2	2	6	8	10				
3	4	7	11	13	15			
4		9	12	16	18	20		
5			14	17	21	23	25	
6				19	22	26	28	30
7					24	27	31	33
8						29	32	34

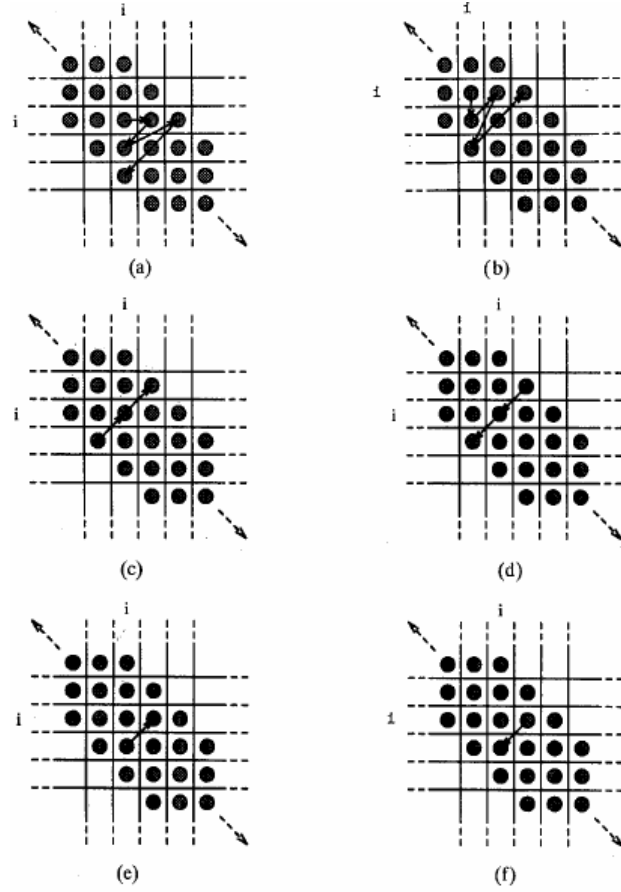
(b) İç içe indis atama  $k=2$  güney

Şekil 3.10: İç içe indis atama yöntemi kullanılarak oluşturulan indis haritaları [28]

- Doğrusal indis atama: Aynı şekilde iç içe indis atamaya seçenек olarak, yönü üst ve aşağı olan iki farklı tarama durumu vardır. Çift olan hücre değerlerinin yukarı ve aşağı sıralaması sırasıyla,  $(i + \lfloor \frac{k}{2} \rfloor, i - \lfloor \frac{k}{2} \rfloor), (i + \lfloor \frac{k}{2} \rfloor - 1, i - \lfloor \frac{k}{2} \rfloor + 1), \dots,$   
 $(i - \lfloor \frac{k}{2} \rfloor, i + \lfloor \frac{k}{2} \rfloor)$  ve  $(i - \lfloor \frac{k}{2} \rfloor, i + \lfloor \frac{k}{2} \rfloor), (i - \lfloor \frac{k}{2} \rfloor + 1, i + \lfloor \frac{k}{2} \rfloor - 1), \dots,$   
 $(i + \lfloor \frac{k}{2} \rfloor, i - \lfloor \frac{k}{2} \rfloor)$  şeklindedir. Tek olan hücre değerlerinin sıralaması ise yukarı için,  
 $(i + 1 + \lfloor \frac{k-1}{2} \rfloor, i - \lfloor \frac{k-1}{2} \rfloor), (i + \lfloor \frac{k-1}{2} \rfloor, i - \lfloor \frac{k-1}{2} \rfloor + 1), \dots,$   
 $(i - \lfloor \frac{k-1}{2} \rfloor, i + 1 + \lfloor \frac{k-1}{2} \rfloor)$  ve aşağı için  $(i - \lfloor \frac{k-1}{2} \rfloor, i + 1 + \lfloor \frac{k-1}{2} \rfloor),$   
 $(i - \lfloor \frac{k-1}{2} \rfloor + 1, i + \lfloor \frac{k-1}{2} \rfloor), \dots, (i + 1 + \lfloor \frac{k-1}{2} \rfloor, i - \lfloor \frac{k-1}{2} \rfloor)$  dir.

Burada  $\lfloor \cdot \rfloor$  gösterimi içerisindeki sayıya yakın en küçük tamsayıyı ifade etmektedir.

Önerilen doğu, güney, alt, üst tarama yöntemleri Şekil 3.11'de gösterilmiştir.



Şekil 3.11: İndis matrisi tarama yöntemleri [28]

Farklı köşegen değerleri için birkaç indis matrisi örneği Şekil 3.12’de verilmiştir. Burada (a) durumunda  $k = 1$ , diğerinde ise  $k = 3$  alınmıştır.

1	3						
2	4	5					
	6	7	9				
		8	10	11			
			12	13	15		
				14	16	17	
					18	19	21
						20	22

(a)

1	3	5					
2	6	8	10				
4	7	11	12	14			
	9	13	16	17	19		
		15	18	21	23	25	
			20	22	26	28	30
				24	27	31	32
					29	33	34

(b)

Şekil 3.12: Farklı köşegen değerleri için indis matrisi örneği

#### **4. EN UYGUN SÜZGEÇLEME KULLANARAK İMGE İÇİN ÇOKLU TANIMLAMALI KODLAMA**

Bu bölümde tez kapsamında yapılan çalışmalardan bahsedilecektir. Tezin özgünlüğü Bölüm 3’de gösterilen ÇTK’lı yöntemlere EUS yönteminin tümleştirilmesidir. Yapılan çalışmalarda ilk olarak çok aşamalı aşağı örnekleme temelli ÇT imge kodlama yapısı ile birlikte EUS yaklaşımı kullanılarak önerilen yöntemden bahsedilecek ve elde edilen sonuçlar verilecektir. Daha sonra Bölüm 4.2’de farklı bir ÇTK yöntemi olan dalgacık temelli ÇTSN yaklaşım için EUS ile imge kodlama işlemi anlatılacaktır.

##### **4.1 EUS Yönteminin Çok Aşamalı Aşağı Örnekleme Temelli ÇT İmge Kodlamaya Uygulanması**

Bu çalışmada hataya açık ağlar üzerinden görüntü iletimini kabul edilebilir seviyede gerçekleştiren gürbüz çoklu tanımlamalı imge kodlama yapısı önerilmiştir. Amaç en küçük kareler yöntemini kullanarak iletim kanalından alınan işaretlerin görüntü bozunumunu en küçükmektir. Önerilen yöntem ÇTK yapısında tanımlamalar arasındaki ilintiyi artıracak artıklık ekleme tabanlıdır.

Kodlayıcıda kodlama işleminden önce aşağı-örnekleme ve kodçözücü kısmında kodçözme işleminden önce üst-örnekleme işlemlerini kullanmak, düşük bit oranlarında geleneksel imge kodlama yaklaşımlarından daha iyi sonuçlar vermektedir [34]. Giriş imgesi ve geri elde edilen imge arasında kodlama ve arkasından örnek seyreltme ile birlikte en uygun süzgeçleme yaklaşımı kullanarak standart JPEG kodlama performansı artırılabilir. [35]’de en uygun süzgeçleme kullanılarak JPEG kodlanmış imgelerin sıkıştırma başarımını iyileştirdiği gösterilmiştir.

Burada önerilen yöntem çok aşamalı aşağı örnekleme temelli sıfır dolgulama [23] yöntemi ile en uygun süzgeçleme yaklaşımının birleştirilmesidir. Bu fikirden yola

çıkarak önerdiğimiz yöntemin [23] ve [36]'deki yöntemler ile karşılaştırıldığında başarımı artıracacağı düşünülmektedir.

Bölüm 3.2'de anlatılan yöntem ile [36]'de önerilen en uygun süzgeçleme (EUS) yaklaşımı tümleştirilmiştir.

Burada  $X$  giriş imgesini gösterirken  $X_n, n = 1,2,3,4$  olmak üzere aşağı örneklenmiş ve tanımlama olarak adlandırılan alt-imgeleri göstermektedir. Alt-imgelerin Bölüm 3.2.2'de detaylı anlatıldığı şekilde, [23]'deki yöntemin kullanılması ile elde edilmesinden sonra her bir alt imge standart JPEG kodlayıcı ile kodlanıp çözülerek, geri çatılmış  $\tilde{X}_n$  imgesi elde edilir. Elde edilen  $\tilde{X}_n$  alt-imgeleri,  $l \times l$  boyutlarında olan en iyi geriçatma süzgeci  $G$  kullanılarak asıl alt-örneklenmiş imgelerin iyileştirilmiş halleri olan  $\hat{X}_n$  elde edilir. (4.1)'de sadece  $\tilde{X}_1$  tanımlaması alındığı durum gösterilmiştir. Alıcıya ulaşamayan diğer tanımlamalar,  $\tilde{X}_1$  tanımlaması ve ilgili en iyi geriçatma süzgeçleri kullanılarak kestirilir. Burada  $*$  işareti evrişim (convolution) işlemini göstermektedir.

$$\begin{aligned}
 \hat{X}_1 &= \tilde{X}_1 * \mathbf{G}_{1,1} \\
 \hat{X}_2 &= \tilde{X}_1 * \mathbf{G}_{1,2} \\
 \hat{X}_3 &= \tilde{X}_1 * \mathbf{G}_{1,3} \\
 \hat{X}_4 &= \tilde{X}_1 * \mathbf{G}_{1,4}
 \end{aligned} \tag{4.1}$$

(4.1)'deki  $\mathbf{G}$  değerleri asıl alt-örneklenmiş imge ile kodlama ve kodçözme sonucu geri elde edilen tanımlamalar arasındaki bozunumu en küçükleyecek şekilde (4.2)'deki ifade ile bulunur. (4.2)'deki eşitliğini çözmek için büyük denklem sistemlerini hızlı bir şekilde çözebilen özyineli olarak önkoşullanmış eşlenik gradyanlar yöntemi (The Iterative Preconditioned Conjugate Gradients method-IPCG) [38,39] kullanılmıştır. Çeşitli denemeler sonucunda işlem yükü performans dengesi dikkate alınarak en uygun süzgecin boyutu  $5 \times 5$  olarak seçilmiştir ve her bir



katsayı değeri 16 bit olarak tutulmaktadır. Süzgeç katsayılarının getirdiği ek yük  $5 \times 5 \times 16 = 400 \text{ bit}$  olmaktadır.

$$\min_{\mathbf{g}_{1,n}} \left\| \mathbf{X}_n - \hat{\mathbf{X}}_1 \right\|_2^2 = \min_{\mathbf{g}_{1,n}} \left\| \mathbf{X}_n - \tilde{\mathbf{X}}_1 * \mathbf{G}_{1,n} \right\|_2^2 \quad (4.2)$$

İşlem kolaylığı açısından  $\mathbf{G}$  matrisi,  $g(m+nl) = G(m,n), 0 \leq m, n \leq l-1$  ile gösterilip,  $b(i+jw) = X_n(i,j), 0 \leq i \leq w, 0 \leq j \leq h$  biçiminde yazılabilir.  $w$  ve  $h$  değerleri kestirilecek imgenin boyutunu gösterir.  $g$  ve  $b$ 'nin yeni biçimi dikkate alınarak (4.3) deki gibi bir  $A$  matrisi oluşturulabilir.

$$A = \begin{bmatrix} \mathbf{a}_{0,0}^T \\ \mathbf{a}_{0,1}^T \\ \vdots \\ \mathbf{a}_{w,h}^T \end{bmatrix} \quad (4.3)$$

Bu durumda çözülecek denklem sistemi,

$$Ag = b \quad (4.4)$$

olacaktır. Bu ifadenin IPCG ile çözülebilmesi için ifadenin kare matris haline getirilmesi şarttır. (4.4)'de her iki taraf  $A^T$  ile çarpılarak ifade kare matris durumuna getirilir.  $\bar{A} = A^T A$  ve  $\bar{b} = A^T b$  olacağından, yeni ifade (4.5)'deki gibi olur.

$$\bar{A}g = \bar{b} \quad (4.5)$$

$\bar{A}$  matrisinin köşegen elemanları kullanılarak oluşturulan pozitif simetrik ön koşullayıcı  $M$  matrisi elde edilir.  $0 \leq m, n \leq l-1$  kısıtı altında  $G_0(m,n) = \delta\left(m - \frac{l-1}{2}, n - \frac{l-1}{2}\right)$  başlangıç değeri olarak alınıp, Şekil 4.1'de algoritması verilen optimize edilmiş IPCG uygulamasına [38] girdi olarak verildiğinde (4.5)'in çözümü öz yineli olarak gerçekleştirilir. Burada  $r^{(i)}$  i.

özyineleme aşaması için artıklığı,  $p_i$  aynı özyineleme aşaması için sonraki adımda izlenecek yönü göstermektedir.

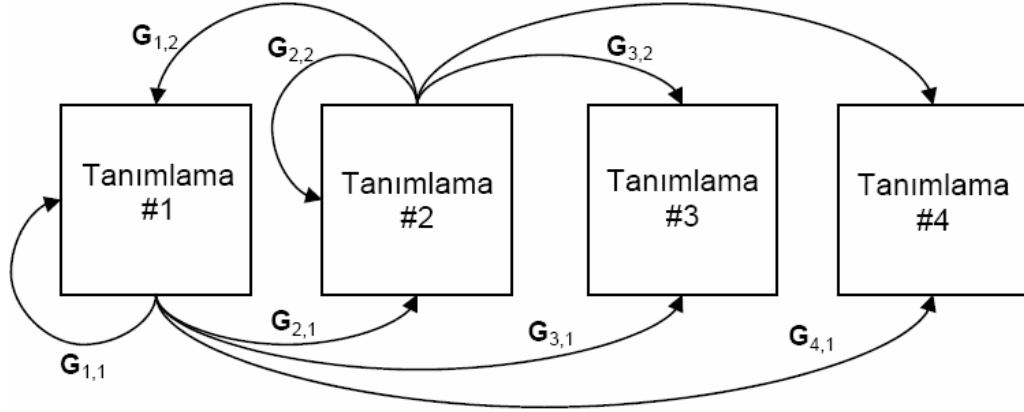
```

for i = 1,2,...
    çöz  $Mz^{(i-1)} = r^{(i-1)}$ 
     $\rho^{(i-1)} = r^{(i-1)T} z^{(i-1)}$ 
    if i = 1
         $p^{(1)} = z^{(0)}$ 
    else
         $\beta_{i-1} = \frac{\rho_{(i-1)}}{\rho_{(i-2)}}$ 
         $\rho^{(i)} = z(i-1) + \beta_{i-1}\rho^{(i-1)}$ 
    endif
     $q_i = A\rho_i$ 
     $\alpha_i = \frac{\rho_{(i-1)}}{\rho^{(i)T} q^{(i)}}$ 
     $x^{(i)} = x^{(i-1)} + \alpha_i \rho^{(i)}$ 
     $r^{(i)} = r^{(i-1)} - \alpha_i q^{(i)}$ 
    yakınlığı kontrol et; gerekliyse devam
end

```

Şekil 4.1: Optimize edilmiş IPCG algoritması [38]

[37]'deki yöntemde her bir tanımlamanın diğer tanımlamaları oluşturmak için 4 adet en uygun süzgeç matrisine ihtiyacı olduğundan toplamda 16 adet süzgeç matrisi kullanılır. Kayıp tanımlamalar, alınan tanımlamalar ve en uygun süzgeçler kullanılarak kestirilir. 3. ve 4. tanımlamaların kaybolması durumunda, 1. , 2. tanımlamalar ve ilgili en uygun süzgeç matrisleri kullanılarak kaybolan tanımlamaların elde edilmesi Şekil 4.2'de gösterilmektedir. Son kestirim, alınan tanımlamalardan kestirilen tanımlamalar oluşturulduktan sonra bu tanımlamaların ortalamaları alınarak elde edilir.

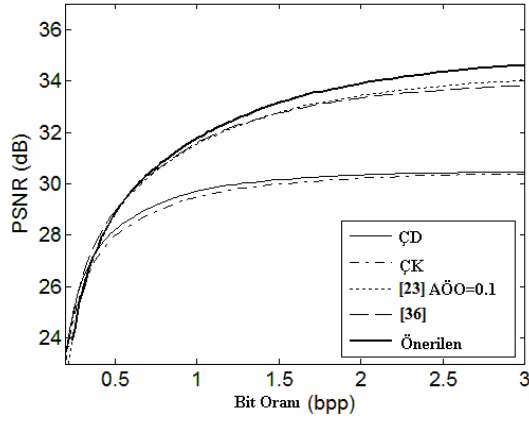


Şekil 4.2: İki tanımlamanın alıcıya ulaşması durumu [48]

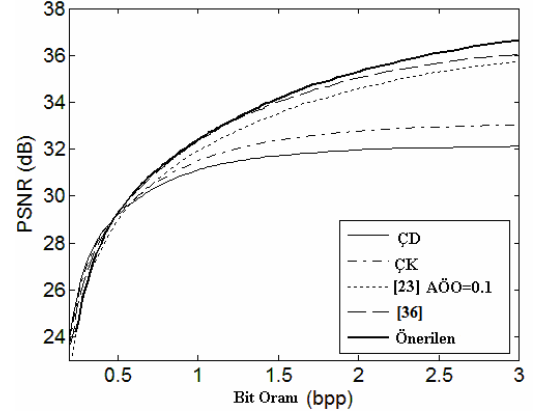
Daha sonra Bölüm 3.2.1’de anlatıldığı gibi kodlayıcı tarafında eklenen artıklık, kodçözücüde AKD kullanılarak frekans uzayına geçilerek kaldırılır. Tekrardan TAKD alınarak sonuç imgesi elde edilir.

#### 4.1.1 Deneysel Sonuçlar

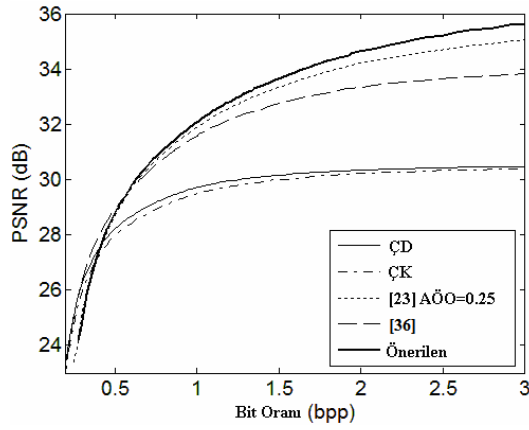
Bu çalışmada önerilen yöntemin başarımını göstermek için  $512 \times 512$  piksel boyutlarındaki Lena, Barbara ve Peppers test imgeleri kullanılmıştır. ÇTK yöntemlerinin başarımını gösterirken genellikle oran bozunum grafikleri verilmektedir. Bozunum değeri olarak PSNR ölçütü kullanılmıştır. Şekil 4.3 ve Şekil 4.4, Lena imgesi için farklı AÖO değerleri ve farklı tanımlamaların gelmesi durumunda elde edilen oran bozunum değerlerini göstermektedir.



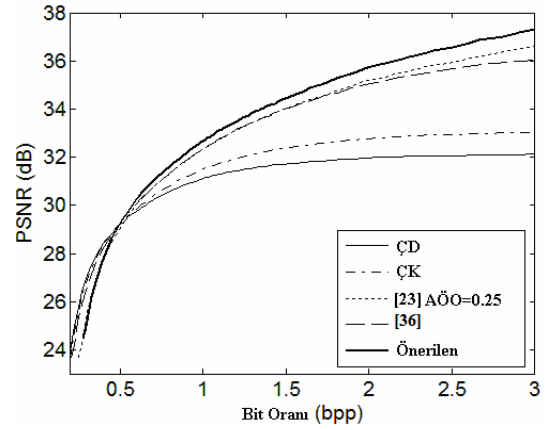
(a) 1 tanımlama alınması durumu



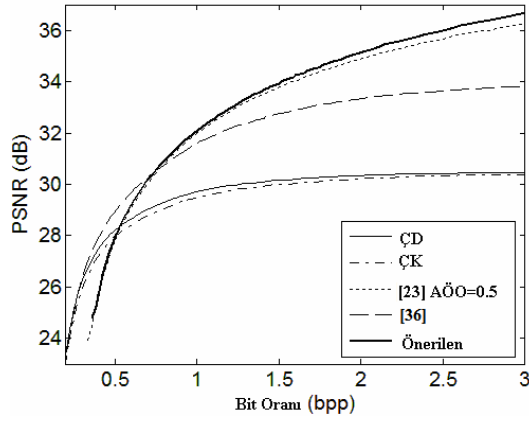
(b) 2 tanımlama alınması durumu



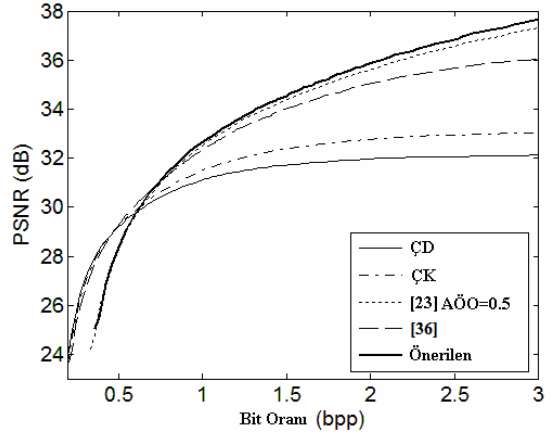
(c) 1 tanımlama alınması durumu



(d) 2 tanımlama alınması durumu

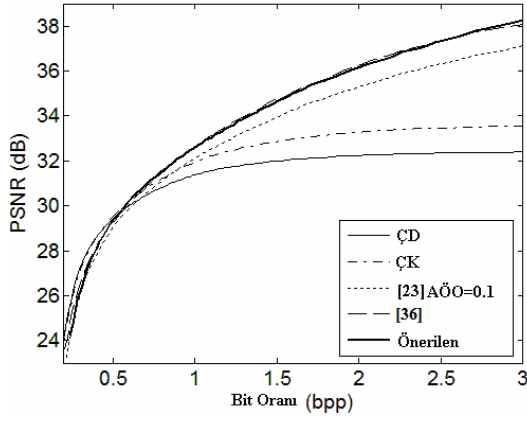


(e) 1 tanımlama alınması durumu

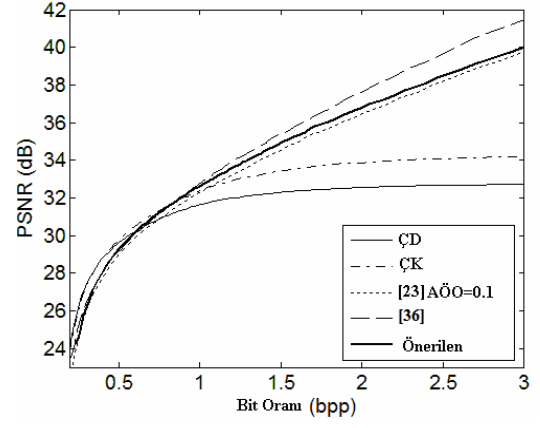


(f) 2 tanımlama alınması durumu

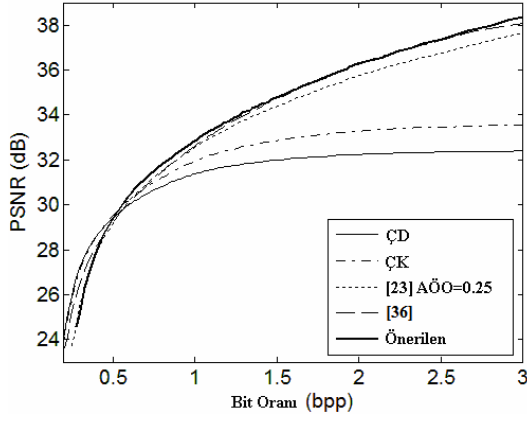
Şekil 4.3: Farklı AÖO değerleri için oran bozunum grafikleri



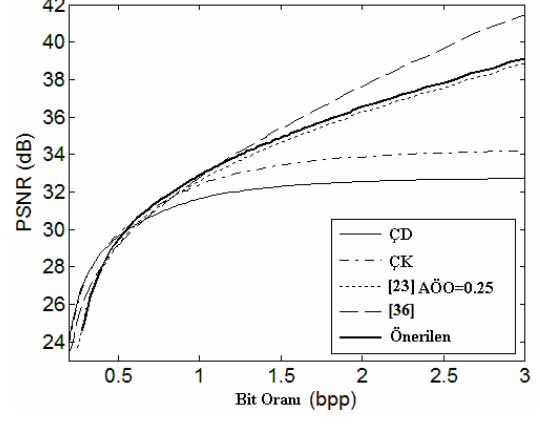
(a) 3 tanımlama alınması durumu



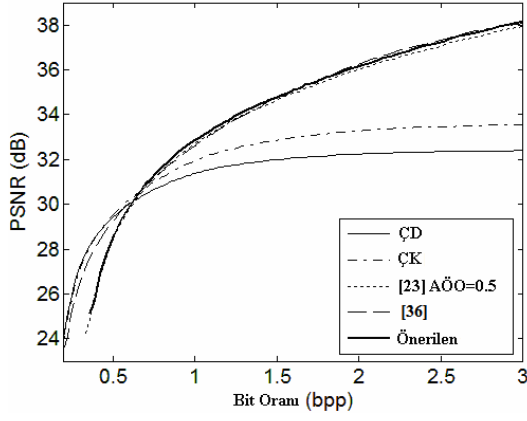
(b) 4 tanımlama alınması durumu



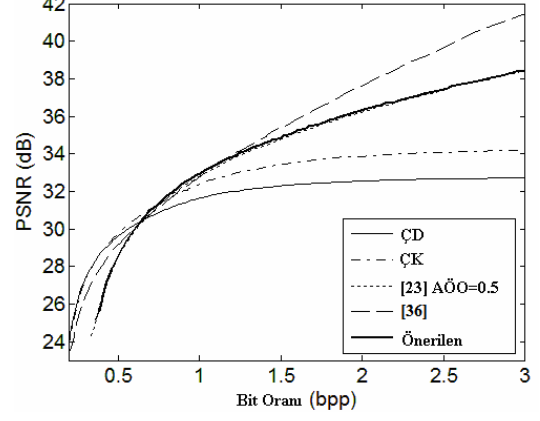
(c) 3 tanımlama alınması durumu



(d) 4 tanımlama alınması durumu

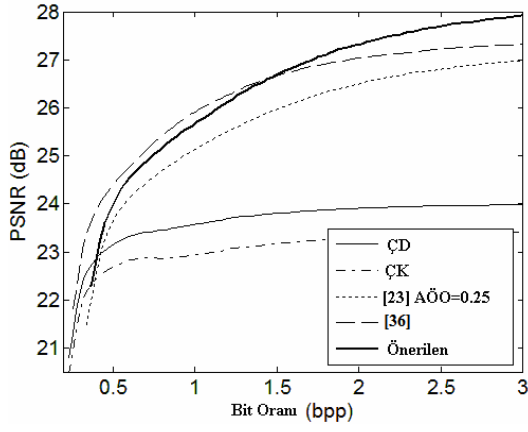


(e) 3 tanımlama alınması durumu

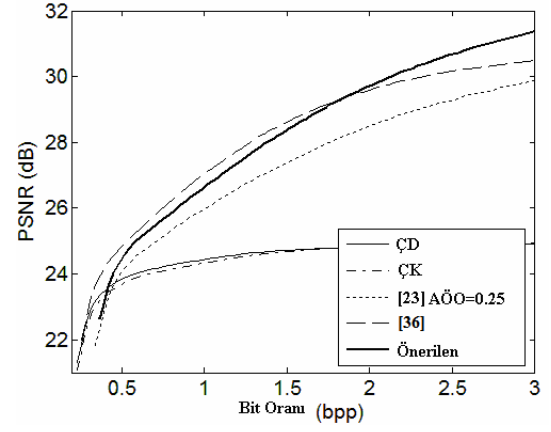


(f) 4 tanımlama alınması durumu

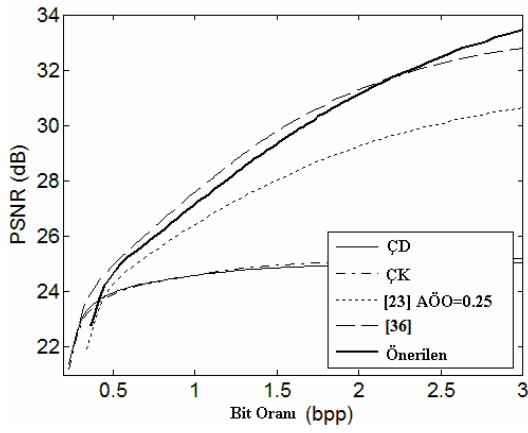
Şekil 4.4: Farklı AÖO değerleri için oran bozunum grafikleri (devam)



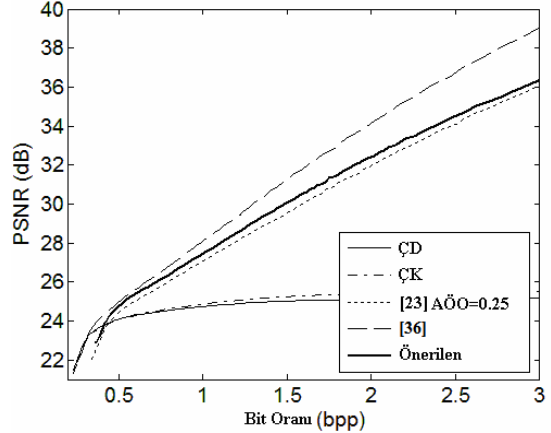
(a) 1 tanımlama alınması durumu



(b) 2 tanımlama alınması durumu

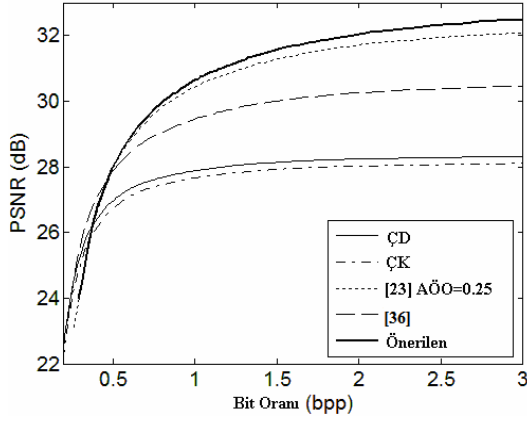


(c) 3 tanımlama alınması durumu

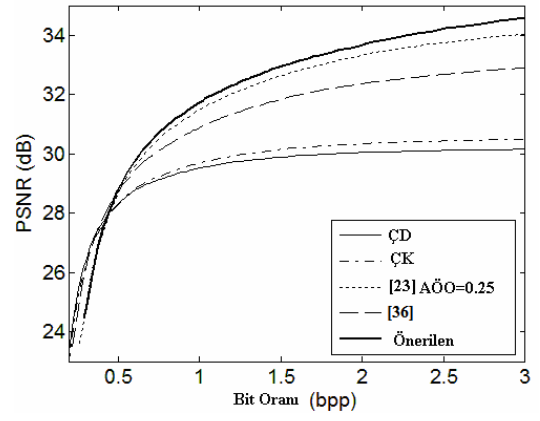


(d) 4 tanımlama alınması durumu

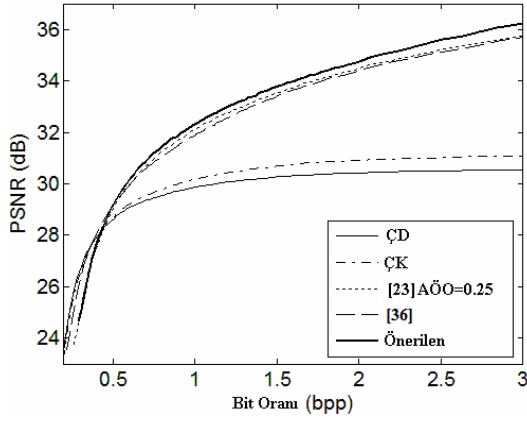
Şekil 4.5: Barbara imgesi için oran bozunum grafikleri



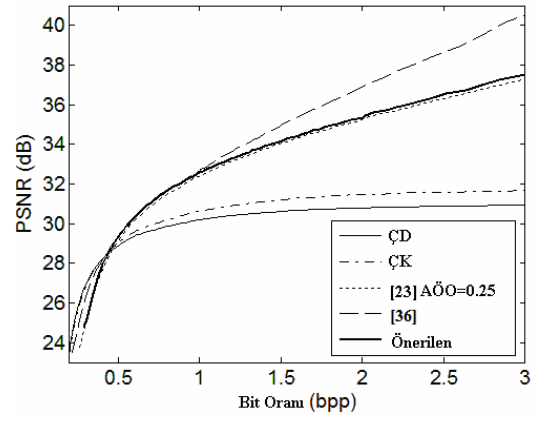
(a) 1 tanımlama alınması durumu



(b) 2 tanımlama alınması durumu



(c) 3 tanımlama alınması durumu



(d) 4 tanımlama alınması durumu

Şekil 4.6: Peppers imgesi için oran bozunum grafikleri

Önerilen yöntem ile karşılaştırma amacıyla kullanılan yöntemler çift doğrusal aradeğerleme (ÇD – bilinear interpolation) ve çift kübik aradeğerleme (ÇK – bicubic interpolation)'dir. [23]'de önerilen 2 boyutlu aşırı örnekleme (2B AÖ) temelli yaklaşım, [36]'de önerilen ise tek aşamalı en uygun süzgeçlemedir (EUS TA). Önerilen birleştirilmiş yaklaşımın birçok durumda en iyi sonucu verdiği görülmektedir. [36]'de sunulan yöntem, sadece alıcıya tüm tanımlamalar ulaştığında daha yüksek başarımlar göstermektedir. Bu durumun temel nedeni, önerilen yöntemdeki ön-işlem sırasında eklenen artıklık sonucu oluşan aşırı örneklemenin yüksek bit oranlarında etkili olmamasından kaynaklanmaktadır.

Kaybolan tanımlama sayısı arttıkça en uygun süzgeçlemenin avantajının arttığı görülmektedir. Kayıp tanımlamaların, alıcıya ulaşan tanımlamalar ile bu tanımlamaların içerdiği ilgili EUS katsayılarının süzgeçlenmesi sonucu yeniden oluşturulması ile elde edilen imge, ÇD ve ÇK aradeğerleme yaklaşımları ile elde edilen imgeden daha iyi sonuç vermektedir. Çünkü ÇD ve ÇK kullanımında aradeğerleme işlemi uygulanarak varolan imge değerlerinin belirli oranlarda ortalaması alınmaktadır. Oysa EUS yöntemi alınan tanımlamayı kullanarak kayıp tanımlamayı en iyi şekilde oluşturmaya çalışır.

AÖO değerinin başarımdaki etkisi incelendiğinde çok düşük bit oranlarında yüksek AÖO verimli değildir ve başarımın düşmesine neden olmaktadır. Bu yüzden yapılan denemeler sonucunda AÖO değeri 0.25’de sabit tutulmuştur. Böylece hem düşük bit oranlarında hem de yüksek bit oranlarında başarımın dengelenmesi sağlanmıştır. ÇD ve ÇK durumları basit aradeğerleme yöntemleri olduğundan bu yöntemler düşük başarım göstermişlerdir. Şekil 4.5’de ve Şekil 4.6’da sırasıyla Barbara ve Peppers imgeleri için oran bozunum sonuçları gösterilmektedir. Sonuçlardan görüldüğü üzere yüksek detay içeren Barbara imgesi [36]’deki yöntemde yüksek bit oranları için daha iyi sonuç vermektedir. Bunun anlamı yoğun detay içeren imgeler için yüksek bit oranlarında sıkıştırma yapıldığında aşırı örnekleme uygun değildir. Her ne kadar en uygun süzgeç katsayıları kullanılarak görüntü kalitesi bir miktar düzeltilmiş olsa da yüksek bit oranlarında kodlama ve fazla sayıda tanımlamanın alıcıya ulaşması durumunda standart en uygun süzgeçleme yaklaşımı daha yüksek başarım göstermektedir. Ayrıca yoğun detay içermeyen Peppers imgesinde aşırı örnekleme iyi performans sergilemekte ve önerdiğimiz yöntem en iyi sonucu vermektedir.

Hesapsal karmaşıklığa bakıldığında 512x512 piksellik bir imge çerçevesi için tüm kodlama süresi ortalama 29s, toplam 16 EUS katsayısı hesabının bir tanesi ortalama 560ms olmaktadır. Kodlama süresinin uzunluğu temel olarak Matlab’da tanımlamaları üretirken imgelerin dosyaya yazılması ve dosyadan tekrar okunmasından kaynaklanmaktadır. Bu süreler değerlendirilirken, JPEG kodlama ve EUS hesabı için kullanılan program kodunun herhangi bir optimizasyona tabii tutulmadığı göz önüne alınmalıdır.



Sonuç olarak, iki boyutlu aşırı örnekleme ile tek aşamalı en uygun süzgeçleme yaklaşımlarının birleştirilmesi yüksek detay içeren imgeler hariç düşük ve orta seviyede detay içeren imgeler için başarılı sonuçlar vermektedir. Yüksek detay içeren imgelerde aşırı örnekleme yeteri kadar etkili olmayıp standart en uygun süzgeçleme yaklaşımından biraz daha kötü sonuç vermektedir.

## **4.2 Dalgacık Temelli ÇTSN Yaklaşımları İçin En Uygun Süzgeçleme ile İmge Kodlama**

Bu tez çalışması kapsamında ayrıca [39]'deki çalışmada verilen ayrık dalgacık dönüşümü tabanlı ÇTSN yaklaşımı gerçekleştirilmiş ve bu yöntemle en uygun süzgeçleme yaklaşımı uygulanmıştır.

[39]'deki yöntemde temel olarak dalgacık dönüşümü ile elde edilen katsayıların ÇTSN ile birleştirilmesi ve bu yapılan çalışmanın başarımının JPEG temelli yaklaşımlara göre daha iyi olduğu gösterilmektedir. Bu yöntemde öncelikle giriş imgesi için dalgacık dönüşümü uygulanmakta, sonra her bir dalgacık katsayısı o katsayının bulunduğu alt-banda bağlı olacak şekilde ÇTSN kullanılarak indis atama işlemi gerçekleştirilmektedir. Elde edilen indisler aritmetik kodlama kullanılarak kanal ortamına iletilmektedir. Kod çözücünde alıcıya ulaşan tanımlamaların kayıpsız kod çözme işlemi sonrası elde edilen dalgacık katsayılarının kullanılması ile orijinal piksel değerleri geri çatılmaktadır.

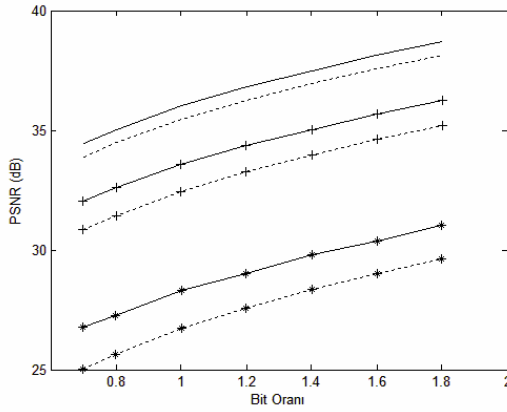
En uygun süzgeçlemeyi dalgacık katsayıları yerine imge düzleminde uygulamanın daha yüksek başarımlar verdiği yapılan denemelerde gösterilmiştir. Bu sebeple dalgacık tabanlı kodlama sonrası imge düzleminde Bölüm 4.1'de açıklanan şekilde en uygun süzgeçleme yapılmış, elde edilen sonuçlar Şekil 4.7 ve Şekil 4.8'de verilmiştir.

### **4.2.1 Deneysel Sonuçlar**

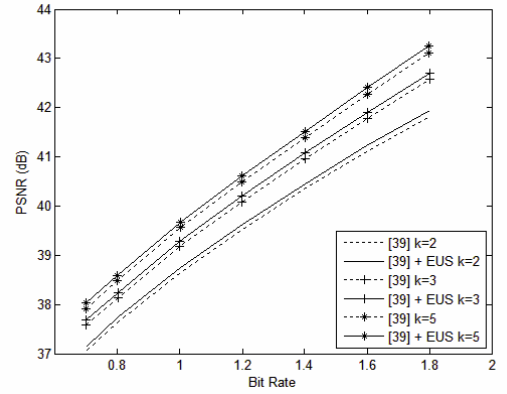
En uygun süzgeç boyutu diğer çalışmalarda olduğu gibi  $5 \times 5$  olarak belirlenmiştir. Tanımlama sayısı 2 olarak alınmıştır. Test amacıyla  $512 \times 512$  piksel boyutlarındaki Lena, Barbara, Peppers ve Goldhill imgeleri kullanılmıştır. Şekil 4.7'de görüldüğü

gibi indis atama matrisindeki köşegen sayısı ( $k$ ) arttıkça tek tanımlamanın alıcıya ulaşması durumunda kodlama başarımı daha düşük kalmakta, çift tanımlamanın ulaşması durumunda ise bu başarımlar daha yüksek olmaktadır. Bu, ÇTSN temelli ÇTK'nın temel ikilemidir. Kullanım amacına ve kullanılan kanalın özelliklerine göre bu parametrelerin ayarlanması ile istenen en uygun sonuçların alınması mümkündür.

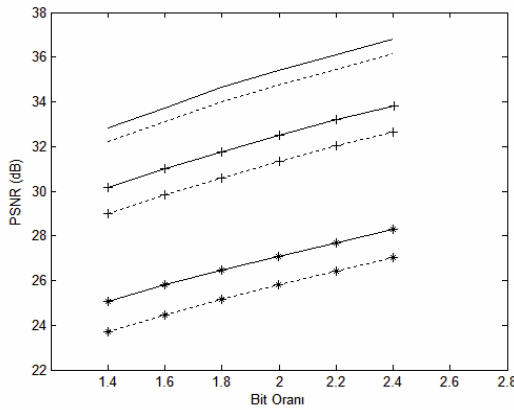
Şekil 4.7 ve Şekil 4.8'de verilen deneysel sonuçlar imge karakteristiklerine göre değerlendirildiğinde alıcıya tek tanımlama ulaşması durumunda önerilen en uygun süzgeçleme yaklaşımının farklı karakteristiklerdeki imgeler için PSNR değerinde 2dB'ye yakın önemli iyileştirmeler yaptığı görülmüştür. Ancak her iki tanımlamanın da alıcıya ulaşması durumunda en uygun süzgeçlemenin yaptığı iyileştirme sınırlı olmaktadır.



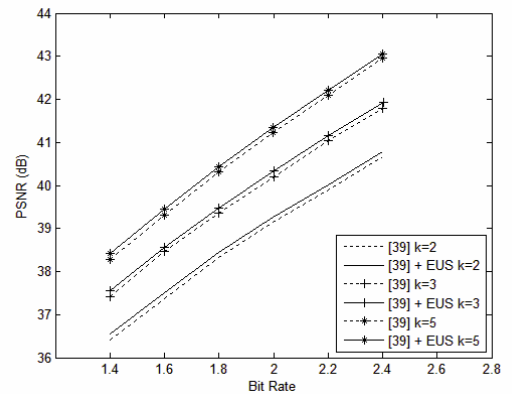
(a) Lena, Tek Tanımlama



(b) Lena, İki Tanımlama

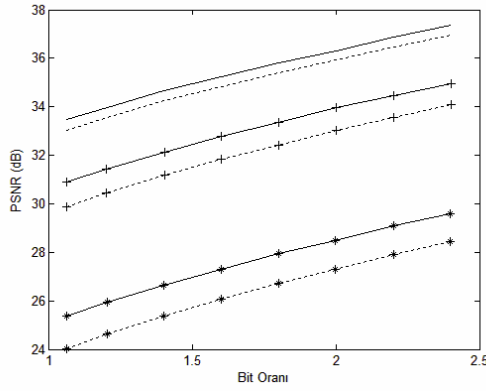


(c) Barbara, Tek Tanımlama

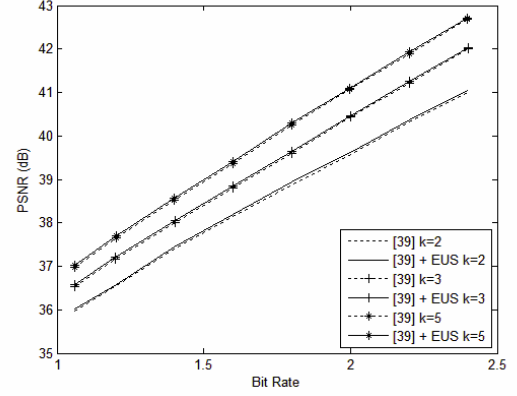


(d) Barbara, İki Tanımlama

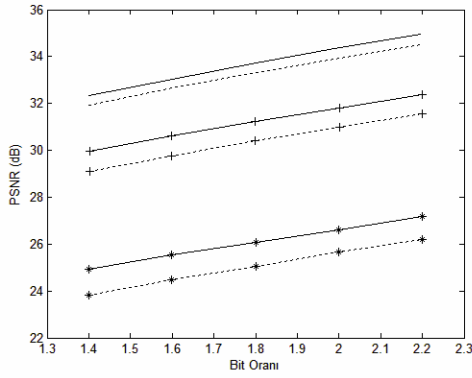
Şekil 4.7: Lena ve Barbara imgeleri için dalgacık tabanlı kodlama sonuçları



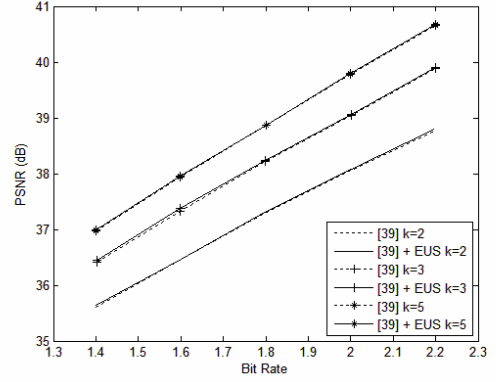
(a) Peppers, Tek Tanımlama



(b) Peppers, İki Tanımlama



(c) Goldhill, Tek Tanımlama



(d) Goldhill, İki Tanımlama

Şekil 4.8: Peppers ve Goldhill imgeleri için dalgacık tabanlı kodlama sonuçları

Bu durum özellikle yoğun düz tonlu bölgeler içeren Peppers ve Goldhill imgeleri için verilen sonuçlarda açıkça görülmektedir.

İşlem sürelerine bakıldığında kodlama işlemi ortalama 2s, EUS katsayı değerlerinin bulunması işlemi 2.8s sürmektedir. Hesaplanan bu süreler için kodlama işleminde herhangi bir optimizasyon yapılmamıştır.

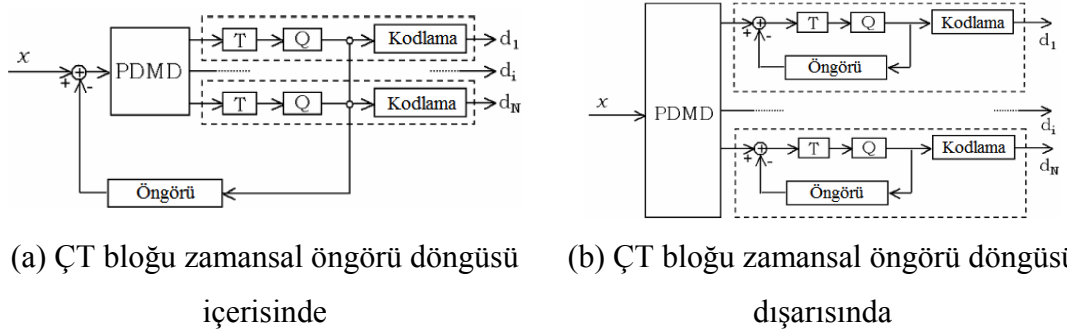
Tez kapsamında kullanılan en uygun süzgeçleme yaklaşımının sadece AKD tabanlı kodlama kullanan ÇTK yaklaşımlarında değil dalgacık tabanlı kodlama kullanan ÇTK yaklaşımları için de başarımda artış sağladığı bu çalışma ile gösterilmiştir.

## 5. EN UYGUN SÜZGEÇLEME KULLANARAK VIDEO İÇİN ÇOKLU TANIMLAMALI KODLAMA

### 5.1 EUS Yönteminin Çok Aşamalı Aşağı Örnekleme Temelli ÇT Video Kodlamaya Uygulanması

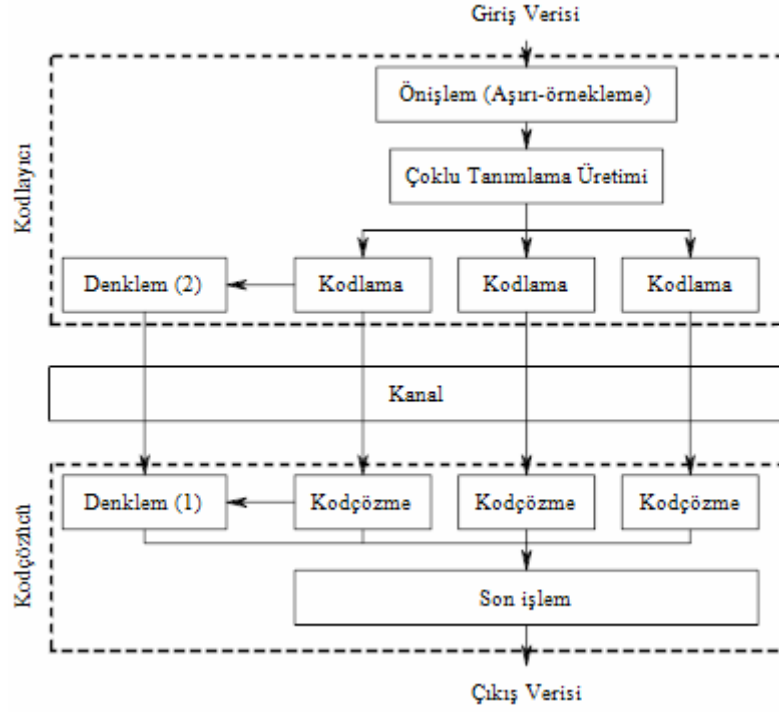
Tez çalışmasının bu kısmında EUS yöntemini kullanan ÇTK tabanlı bir video sıkıştırma yaklaşımı sunulmuştur. Önerilen yöntem, kodlama işleminden önce video çerçevelerinin çok-aşamalı aşağı örnekleme temeline çalışır.

ÇTK'nın ses ve görüntü kodlamadaki kullanımı göreceli olarak kolay olmasına rağmen, video söz konusu olduğunda kodçözücüde uyumsuzluk (mismatch) veya kayma (drift) olarak isimlendirilen kestirim hataları oluşabilir. Bu sebeple ÇTK'nın video kodlamaya uygulanması çok daha karmaşık olmaktadır. Tipik bir video kodlayıcısında, hareket kestirim tabanlı zamansal öngörü gerçekleştirilerek videoyu verimli bir şekilde sıkıştırmak için ardışık video çerçevelerinde varolan zamansal ilinti kullanılır. Fakat iki tanımlama ile ÇTK yaklaşımı kullanılmışsa üç farklı öngörü hatası olabilir. Herhangi bir zamanda kodlayıcı ve kodçözücüdeki öngörü hatası farklı tanımlamaların kullanımından dolayı aynı değilse, çerçeve-içi (intra) kodlanmış çerçeve alınana kadar öngörünün uyumsuzluğundan dolayı videonun düzgün bir şekilde geri oluşturulması mümkün değildir.



Şekil 5.1: ÇT video kodlama döngüsünde uyumsuzluğu giderme[27]

Uyumsuzluğu gidermenin en kolay yolu Şekil 5.1'deki gibi ÇT yöntemini ayrıık öngörü döngüsü dışında kullanmaktır [27]. Fakat bu durumda kodlama başarımı düşebilir ve iki tanımlama alınması durumunda iyileştirilmiş kalitedeki videoyu elde etmek için yan tanımlamaları birleştirme gibi ek işlemler gerekebilir. [40]'de olası bir uyumsuzluğun önüne geçmek için birbirine benzer iki yan kodlayıcının öngörülerini, öngörü döngüsü içerisinde kaba bir nicemleyici ile nicemlenmiştir. [40]'deki başarıım kötüleşmesini hafifleten bir yaklaşım [41]'de önerilmiştir. [40] ve [41]'nin öngörü döngüsünde bulunan nicemleyici [42]'deki çalışmada kaldırılmıştır ve uyumsuzluğu önleyen merkezi kodçözücü geliştirilmiştir. Kayma etkisinin üstesinden gelmenin diđer bir yolu bir miktar kaymaya izin verip, üç farklı yöntemin önerildiđi [43]'de gösterildiđi gibi kayma kontrol altında tutmaktır. [44]'de video çerçeveleri artık (redundant) ve iyileştirme (refinement) parçalarına bölünmüştür. Bu durumda artık parça tüm tanımlamalarda kullanılırken iyileştirme parçaları sadece bazı tanımlamalar içerisinde kodlanır. Böylelikle, artık kısım videolarda uyumsuzluğu önleyici bir yapı olan hareket dengeleme işleminde kullanılan referans çerçevelerini oluşturan tüm tanımlamalarda kullanılabilir. [45]'de AKD yerine eşleşen takip (matching pursuits) ve tüm tanımlamalar alındığında ÇTK'nın başarımını değerlendirmek için en büyük olabilirlik kestirimi [43]'deki üç öngörü döngüsü dikkate alınarak gerçekleştirilmiştir. [46]'de sunulan yöntemde, [43]'deki üç öngörü döngüsünün kullanılmış fakat uyumsuzluktan sakınmak için yan bilginin iletimine gereksinim ortadan kaldırılmıştır. [27]'de video iletimi için iki boyutlu aşırı örnekleme (AÖ) kullanan ÇTK tabanlı çok-aşamalı aşağı örnekleme (polyphase downsampling-PD) yaklaşımı sunulmuştur. Ayrıca uyumsuzluğu önlemek için iki farklı yaklaşım önerilmiştir. Bunlardan birincisi, [43]'de sunulan üç öngörücü döngü yaklaşımının kullanımını temel alırken, ikincisi, tanımlamaların uyumsuzluđunu önlemek için kodlama döngüsünden önce üretilmesini önermektedir. [26]'da iki boyutlu aşırı örnekleme kullanılarak kodlama verimliliđinin artırılabilceđi gösterilmiştir.



Şekil 5.2: ÇTK temelli çok aşamalı video kodlama

Bu çalışmada videonun ÇTK ile kodlanması için önerilen yöntemin öbek gösterimi Şekil 5.2’de verilmektedir. Şekil 5.2’den görüldüğü gibi, Bölüm 3.2.1’de imgeye uygulanan, tanımlamalara kontrollü artıklık ekleme işlemi, aynı şekilde burada da gerçekleştirilmiştir. [27]’de AKD’si alınan imgeye her iki yönde sıfır dolgulama önerilmişti. [26]’da önerilen yöntemde sıfır dolgulama işlemi, sıfır yayma işleminin hangi yönde en verimli olacağına göre seçilerek sadece o yöne uygulanmıştır. Sıfır dolgulama kullanılarak elde edilen aşırı örneklenmiş veri daha sonra [47]’de gösterildiği gibi aşağı örneklenerek çoklu tanımlamalara ayrıştırılır. Son olarak tüm tanımlamalar birbirlerinden ayrı olarak kodlanır, sıkıştırılır ve alıcıya iletilir. Alıcıda, uygun şekilde alınan tanımlamalar kodçözücüye taşınır. Alıcı çıkış verisini yaratmak için gelen tanımlamaları birleştirir.



İşlemlerin çoğu kodlayıcı kısmında yapıldığı için kodçözücü tarafında işlem yükü azaltılmıştır.

EUS süzgeç katsayılarını elde etme, imge kısmında hesaplandığı gibidir. Bu sebeple burada sadece ifadeler ve neyi temsil ettiği gösterilmiştir.

Video işlemlerinde EUS için işlemler çerçeve çerçeve yapılmaktadır. Bu sebeple çerçevenin en uygun süzgeçlenmiş hali (5.1)'deki gibi ifade edilebilir. Burada  $\tilde{I}$  kodlanıp geri oluşturulmuş video çerçevesini  $\mathbf{G}$ , boyutları  $l \times l$  olan iki boyutlu çekirdeği,  $\hat{I}$  süzgeçlenmiş video çerçevesini ve  $*$  evrişim işlemini belirtmektedir.

$$\hat{I} = \tilde{I} * \mathbf{G} \quad (5.1)$$

En uygun süzgecin ( $\mathbf{G}$ ) katsayıları ise (5.2)'deki gibi hesaplanabilir. Bu eşitlikte  $I$  asıl video çerçevesini göstermektedir.

$$\min_{\mathbf{G}} \|I - \hat{I}\|_2^2 = \min_{\mathbf{G}} \|I - \tilde{I} * \mathbf{G}\|_2^2 \quad (5.2)$$

Bu ifadenin çözümü için [37]'de ayrıntıları verilen özyineli önkoşullanmış eşlenik gradyenler yöntemi [49]'de açıklanan şekilde uygulanmıştır. Çekirdek boyutu [51]'deki deneyimler sonucu  $5 \times 5$  boyutunda seçilmiştir. Asıl video çerçevesi  $I$  kullanılarak aşırı örnekleme oranı  $\nu$  ile aşırı örneklenmiş  $I_{os}$  çerçevesi oluşturulur. Bu aşırı örneklenmiş video çerçevesi istenilen aşağı örnekleme örüntüleri kullanılarak  $I_{os}^1$  ve  $I_{os}^2$  olan iki tanımlama şeklinde ikiye ayrılır. Bu tanımlamalar video kodlayıcıya gönderilir ve kodlanan tanımlamalar  $\tilde{I}_{os}^1$  ve  $\tilde{I}_{os}^2$  şeklinde elde edilir.



$$\min_{\mathbf{G}_{1,1}} \|I_{os}^1 - \tilde{I}_{os}^1 * \mathbf{G}_{1,1}\|_2^2, \quad \hat{I}_{os}^1 = \tilde{I}_{os}^1 * \mathbf{G}_{1,1} \quad (5.3)$$

$$\min_{\mathbf{G}_{1,2}} \|I_{os}^2 - \tilde{I}_{os}^1 * \mathbf{G}_{1,2}\|_2^2, \quad \hat{I}_{os}^2 = \tilde{I}_{os}^1 * \mathbf{G}_{1,2}$$

$$\min_{\mathbf{G}_{2,1}} \|I_{os}^1 - \tilde{I}_{os}^2 * \mathbf{G}_{2,1}\|_2^2, \quad \hat{I}_{os}^1 = \tilde{I}_{os}^2 * \mathbf{G}_{2,1} \quad (5.4)$$

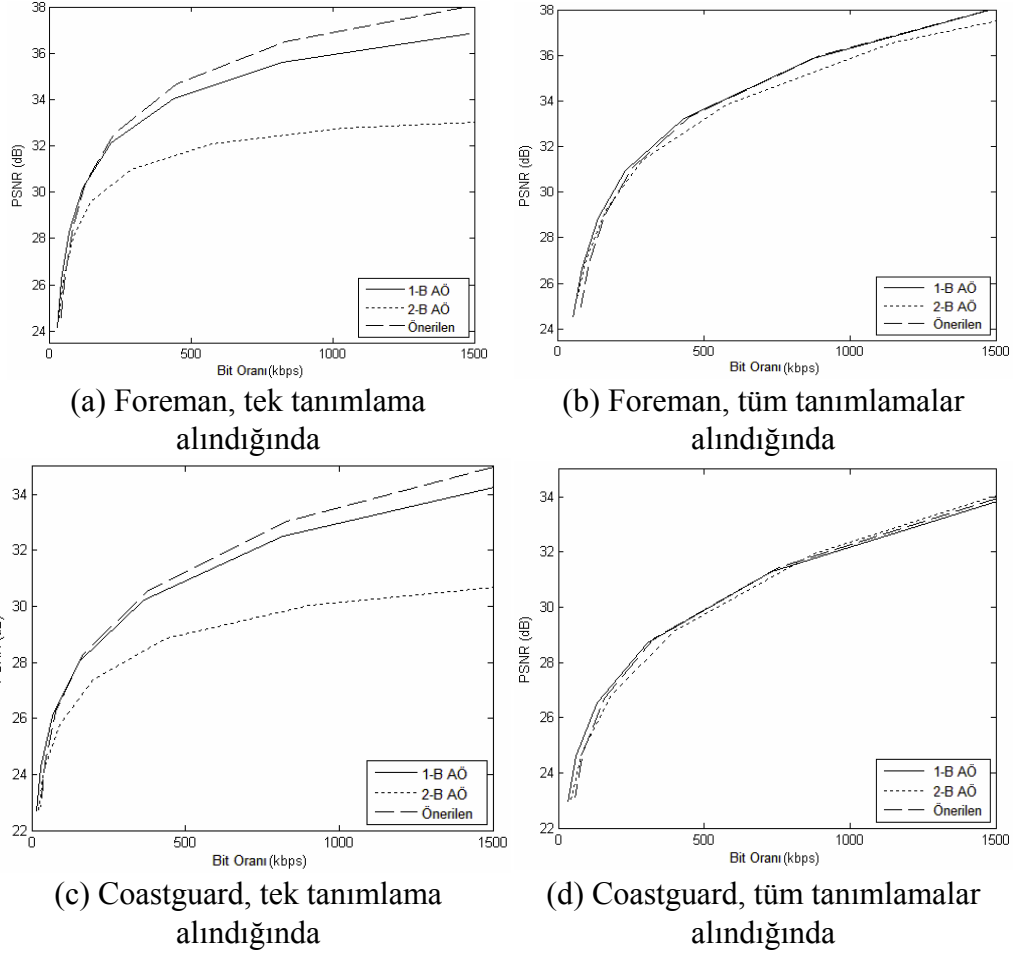
$$\min_{\mathbf{G}_{2,2}} \|I_{os}^2 - \tilde{I}_{os}^2 * \mathbf{G}_{2,2}\|_2^2, \quad \hat{I}_{os}^2 = \tilde{I}_{os}^2 * \mathbf{G}_{2,2}$$

Her bir tanımlama için iki adet en uygun süzgeç hesaplanır. Birinci ve ikinci tanımlamalar için gerekli hesaplamalar (5.3) ve (5.4)'de verilmiştir. Buradan elde edilen iki adet en uygun süzgece ait katsayılar, her bir tanımlama için bit akışına çoklanır. Kodçözücüde, her bir tanımlama ve o tanımlamaya ait olan en uygun süzgeç katsayıları hem diğer tanımlamaların kestirimi, hem de tanımlamanın kendisinin iyileştirilmiş halini elde etmek için kullanılır. Bu yüzden eğer bir tanımlama bozulursa, kalan tanımlamalar ve ilgili en uygun süzgeç katsayıları kayıp tanımlamayı kestirmek için kullanılır. EUS katsayıları ve alınan tanımlamalar kullanılarak kestirilen kayıp tanımlama(lar), alınan tanımlamalar ile birleştirilir. Birleştirilen tanımlamalar sonucu geri elde edilen çerçeve içerisinde kodlayıcıda yapılan aşırı örnekleme etkisinin giderilmesi için frekans uzayına geçilerek eklenen aşırı örnekleme verisi çıkarılır. Tekrardan uzamsal boyuta geçilerek video çerçevesi geri elde edilmiş olur. Böylece son-işlem kısmı gerçekleşir ve işlem tamamlanır.

### 5.1.1 Deneysel Sonuçlar

Deneysel sonuçlar,  $(352 \times 288)$  standart boyutlardaki CIF video formatındaki 300 çerçevelik Foreman ve Coastguard dizileri kullanılarak elde edilmiştir. Tüm bit oranları için başarımlar ve işlem karmaşıklığını dengelediğinden aşırı örnekleme değeri  $\nu = 0.25$  alınmıştır. Literatürde genellikle iki tanımlama kullanıldığından deney sonuçlarında sadece bu durum dikkate alınmıştır. Önerilen en uygun süzgeçleme yaklaşımının başarımlarını değerlendirmek için standart video kodlayıcı H.264/AVC [51] kullanılmıştır. Testler referans yazılımın 10.1 sürümü [52] kullanılarak yapılmıştır.

Deneysel sonuçlar, önerilen en uygun süzgeçleme yaklaşımının, literatürde verilen geleneksel ÇTK yöntemlerine (tanımlamaların bazılarının muhtemel kayba uğradığı) göre başarımı önemli derecede artırdığını göstermektedir.



Şekil 5.4: Farklı test dizileri ve yöntemleri için ÇTK sonuçları

Şekil 5.4 Foreman ve Coastguard dizileri için oran-bozunum eğrisini göstermektedir. İletim hatası durumunda kanaldaki bir tanımlamaya ait tüm verinin kaybedildiği varsayılmıştır. Bir boyutlu (1-B) AÖ ve iki boyutlu (2-B) AÖ, sırasıyla [26] ve [27]'de önerilen aşırı örnekleme yaklaşımlarını göstermektedir. Bu yaklaşımlar karşılaştırma amacıyla gerekli değişiklikler yapılarak video kodlamaya uygulanmıştır. İki boyutlu aşırı örneklemeden sonra yatay ve dikey olmak üzere iki olası aşağı örnekleme yönü vardır [27]. Daha iyi sonuç verdiği için burada yatayda aşağı örnekleme sonuçları verilmiştir. Aşırı örnekleme ve aşağı örneklemenin her iki yöne uygulanması durumunda dört olası senaryo elde edilir [26]. Yapılan denemelerde dikey yönde sıfır dolgulama ve yatayda aşağı örnekleme

yapılması tek boyutlu aşırı örnekleme yaklaşımında daha iyi sonuçlar vermektedir. Bu yüzden 1-B AÖ için sadece bu durumdaki sonuçlar verilmiştir. 1-B AÖ ve 2-B AÖ için kayıp tanımlamaların geri elde edilmesinde doğrusal uzamsal kestirim kullanılabilir.

Süzgeç katsayılarının getirdiği ek yük göreceli olarak küçüktür ve her bir süzgeç katsayısı sıkıştırma olmaksızın 16 bit ile temsil edilip bit akışına çoklanır. Sonuçta  $5 \times 5 \times 16 \text{bit} \times 30 \text{fps} = 12 \text{kbps}$  ek yük oluşmaktadır. Bu yük Şekil 5.3'de verilen sonuçlara dahil edilmiştir.

Diğer taraftan tez kapsamında geliştirilen yöntem, paket kaybı durumunda sadece bir tanımlama alındığında 1-B AÖ yaklaşımını önemli derecede iyileştirmektedir. Alıcıya tüm tanımlamalar ulaştığında önerdiğimiz yöntemin başarımlı artışı eklediğimiz artırıktan dolayı daha az olmaktadır.

## SONUÇLAR VE ÖNERİLER

Bu çalışmada çoklu tanımlamalı kodlanmış imge ve videoların en uygun süzgeçlenmesi için çeşitli yöntemler geliştirilmiştir. Çoklu tanımlamalı kodlama yapısı temel olarak verileri birbiri ile ilintili tanımlamalara dönüştürerek kanaldaki herhangi bir bozulma durumuna karşı veriyi alıcıda kabul edilebilir kalitede geri elde etmekte kullanılan bir yöntemdir. Bu yapıya uyarlanan en uygun süzgeçleme yönteminin başarımı artırdığı gösterilmiştir.

Aşağı örnekleme temelli çoklu tanımlamalı kodlama yöntemine uygulanan en uygun süzgeçleme işleminde yüksek miktarda detay içeren imgeler için yüksek bit oranlarında sıkıştırma yapıldığında aşırı örnekleme uygun değildir. Her ne kadar en uygun süzgeç katsayıları kullanılarak kalite bir miktar düzeltilmiş olsa da yüksek bit oranlarında kodlama ve fazla sayıda tanımlamanın alıcıya ulaşması durumunda standart en uygun süzgeçleme yaklaşımı daha yüksek başarımlar göstermektedir. Sonuç olarak, iki boyutlu aşırı örnekleme ile tek aşamalı en uygun süzgeçleme yaklaşımlarının birleştirilmesi yüksek detay içeren imgeler hariç düşük ve orta seviyede detay içeren imgeler için başarılı sonuçlar vermektedir.

Dalgacık temelli çoklu tanımlamalı skaler nicemleyici için en uygun süzgeçleme işlemi ile sadece AKD tabanlı kodlama kullanan çoklu tanımlamalı kodlama yaklaşımlarında değil dalgacık tabanlı kodlama kullanan çoklu tanımlamalı kodlama yaklaşımları için de başarımda artış sağladığı gösterilmiştir.

Önerilen video kodlama yönteminde alıcıya sadece tek tanımlamanın ulaşması durumunda en uygun süzgeçleme yaklaşımının sıkıştırma performansını artırdığı gösterilmiştir. Her iki tanımlamanın da alıcıya ulaşması durumunda ise performansta hafif bir artış olmakta fakat bu durum hız bozunum eğrilerine yansımamaktadır. Bunun sebebi her iki tanımlamanın da alıcıya ulaşması durumunda süzgeç

katsayılarının getirdiği ek bit miktarı yükünü karşılayacak kadar bir iyileşmenin sağlanamamasıdır.

En uygun süzgeç katsayılarını herhangi bir sıkıştırma yapmadan göndermek yerine entropi kodlama uygulayarak göndermek sıkıştırma verimliliğini artıracaktır. Önerilen yöntemlerde EUS katsayılarının sıkıştırma işlemi yapılmadan gönderildiği varsayılmıştır. Kayıpsız kodlama tekniklerinden biri olan entropi kodlama tekniği EUS katsayılarının sıkıştırılarak kanal üzerinden daha verimli bir şekilde iletilmesini sağlar. Bununla ilgili [53]'deki çalışmada EUS katsayılarının öncelikle FDKM ardından Golomb-Rice (GR) yöntemi kullanılarak %50 oranına kadar sıkıştırılabileceği gösterilmiştir.

Bölüm 4.2'de önerilen çalışmada EUS katsayılarının hesaplanması işlemi 2.8s sürmektedir. Kodlarda herhangi bir optimizasyon yapılmamıştır. Kodlama optimizasyonu ile en uygun süzgeç katsayılarının hesaplanması işleminin gerçek zamanlı gerçekleştirilebilmesi mümkündür. [49]'deki H.263+ ve H.264/AVC kodlayıcıları ile yapılan çalışmada en uygun süzgeçleme katsayılarının P4 1.7 GHz Centrino mobile işlemci ve 1GB RAM'e sahip bilgisayar ile ortalama olarak 50ms altında hesaplandığı gösterilmiştir. [50]'de iki boyutlu en uygun süzgeçleme yerine tek boyutlu EUS yaklaşımı denenmiş ve başarımı çok az düşürmesine karşılık işlem süresini 12 ms civarına düşürmüştür. Özetle en uygun süzgeçleme işleminin video verisi için bile gerçek zamanlı olarak yapılması mümkündür.

İleriki çalışmalarda EUS yöntemi, çiftler ile ilişkilendirme dönüşümü (Pairwise correlation transform-PCT) temelli ÇTK yöntemlerine uygulanabilir. EUS yönteminin ÇTSN yöntemi ile birleştirilerek H.264/AVC video kodlayıcıya uygulanması üzerine çalışılması da mümkündür.

## KAYNAKLAR

- [1] Goyal, V.K., "Multiple Description Coding: Compression Meets the Network", *IEEE Signal Processing Magazine*, 18(5), 74-93, (2001).
- [2] Goyal, V.K. "Single and Multiple Description Transform Coding with Bases and Frames", *In Proc. of SIAM*, 2001.
- [3] Goyal, V.K., "Theoretical Foundations of Transform Coding," *IEEE Signal Processing Mag.*, 18, 9-21, (2001).
- [4] Jayant, N.S., "Subsampling of a DPCM Speech Channel to Provide Two 'Self-Contained' Half-rate Channels," *Bell Syst. Tech. J.*, 60(4), 501-509, (1981).
- [5] Jayant, N.S. and Christensen, S.W. "Effects of Packet Losses in Waveform Coded Speech and Improvements due to an Odd-even Sample-interpolation Procedure," *IEEE Trans. Commun.*, 29, 101-109, (1981).
- [6] Papoulis, A., "Probability, Random Variables, and Stochastic Processes", 3rd ed., *McGraw-Hill*, (1991).
- [7] Berger, T., "Rate Distortion Theory", *Prentice-Hall*, (1971).
- [8] Campana, O, "Codifica a Descrizioni Multiple applicata ad H.264/AVC", Doktora Tezi, *Universitata di Padova*, Italy, 39, 2004.
- [9] Cover, T.M. and Thomas, J.A., "Elements of Information Theory", *Wiley*, (1991).
- [10] Gray, R.M., "Source Coding Theory", *Kluwer*, (1990).
- [11] Effros, M., "Optimal Modeling for Complex System Design," *IEEE Signal Processing Mag.*, 15, 51-73, (1998).
- [12] Gersho, A. and Gray, R.M., "Vector Quantization and Signal Compression", Boston, MA: *Kluwer*, (1992).
- [13] Gray, R.M. and Neuhoff, D.L., "Quantization," *IEEE Trans. Inform. Theory*, 44, 2325-2383, (1998).
- [14] Shannon, C.E., "Coding Theorems for a Discrete Source with a Fidelity Criterion," *IRE Int. Conv. Rec.*, 7(4), 142-163, (1959). Reprinted with changes in *Information and Decision Processes*, Machol, R.E., Ed., New York: *McGraw-Hill*, 93-126, (1960).

- [15] Gamal A.A. and Cover, T.M., "Achievable Rates for Multiple Descriptions," *IEEE Trans. Inform. Theory*, 28, 851-857, (1982).
- [16] Zhang, Z. and Berger, T., "New Results in Binary Multiple Descriptions," *IEEE Trans. Inform. Theory*, 33(4), 502-521, (1987).
- [17] Venkataramani, R., Kramer, G. and Goyal, V.K., "Multiple Description Coding with Many Channels," *IEEE Trans. Inform. Theory*, 49(9), 2106-2114, (2003).
- [18] Ozarow, L., "On a Source-coding Problem with Two Channels and Three Receivers," *Bell Syst. Tech. J.*, 59(10), 1909-1921, (1980).
- [19] Zamir, R., "Gaussian Codes and Shannon Bounds for Multiple Descriptions," *IEEE Trans. Inform. Theory*, 45, 2629-2635, (1999).
- [20] Liu, Y., Oraintara, S., "Feature-Oriented Multiple Description Wavelet-Based Image Coding", *IEEE Trans. on Image Processing*, 16(1), 121-131, (2007).
- [21] Sun, G., Samarawickrama, U., Liang, J., Tian, C., Tu, C., Tran, T.D., "Multiple Description Coding with Prediction Compensation", *IEEE Trans. on Image Processing*, 18(5), 1037-1047, (2009).
- [22] Reibman, A.R., Jafarkhani, H., Wang, Y., Orchard, M.T., Puri, R., "Multiple Description Video Coding Using Motion-Compensated Temporal Prediction", *IEEE Trans. on Circuits and Systems for Video Technology*, 12(3), 193-204, (2002).
- [23] Franchi, N., Fumagalli, M., Lancini, R., "Flexible Redundancy Insertion in a Polyphase Down Sampling Multiple Description Image Coding", *IEEE Int. Conf. Multimedia and Expo*, 605-608, (2002).
- [24] Jiang, W., Ortega, A., "Multiple Description Coding via Polyphase Transform and Selective Quantization", *Proc. SPIE*, 3653, 998-1008, (1981).
- [25] Shirani, S., Gallant, M., Kossentini, F., "Multiple Description Image Coding Using Pre- and Post-processing", *IEEE Int. Conf. Information Technology: Coding and Computing*, 35-39, (2001).
- [26] Tillo, T., Olmo, G., "Data-Dependent Pre- and Postprocessing Multiple Description Coding of Images", *IEEE Trans. on Image Processing*, 16(5), 1269-1280, (2007).
- [27] Franchi, N., Fumagalli, M., Lancini, R. and Tubaro, S., "Multiple description video coding for scalable and robust transmission over IP", *IEEE Trans. Circuits Syst. Video Technol.*, 15(3), 321-334, (2005).
- [28] Vaishampayan, V.A., "Design of Multiple Description Scalar Quantizers", *IEEE Trans. Inform. Theory*, 39, 821-834, (1993).

- [29] Vaishampayan, V.A., Domaszewicz, J., "Design of Entropy-constrained Multiple Description Scalar Quantizers", *IEEE Trans. Inform. Theory*, 40, 245–250, (1994).
- [30] Servetto, S.D, Vaishampayan, V.A., Sloane, N.,J.,A, "Multiple Description Lattice Vector Quantization", *IEEE Data Compression Conf.*, 13-22, (1999).
- [31] Conway, J.H., Sloane, N.J.A., "Fast Quantizing and Decoding Algorithms for Lattice Quantizers and Codes", *IEEE Trans. Inform. Theory*, 28(2), 227–232, (1982).
- [32] Tian, C., Hemami, S.S., "A New Class of Multiple Description Scalar Quantizer and Its Application to Image Coding", *IEEE Signal Proc. Letters*, 12(4), 329–332, (2005).
- [33] Johnsen, S.T., Standeren, M., "Evaluate Multiple Description Coding as an Image Processing Method for Transferring Information in Error-Prone Networks with Low Transmission Rate, Related to Quality, Bitrate and Filesize", Yüksek Lisans Tezi, Agder University College , Grimstad, (2005).
- [34] Tsaig, Y., Elad, M., Milanfar, P., Golub, G.H., "Variable Projection for Near-Optimal Filtering in Low Bit-Rate Block Coders", *IEEE Transactions on Circuits and Systems for Video Technology*, 15(1), 154 -160, (2005).
- [35] Urhan, O. and Ertürk, S., "Parameter Embedding Mode and Optimal Post Process Filtering for Improved WDCT Image Compression", *IEEE Trans. Circuit Syst. Video Technol.*, 18(4), 528-532, (2008).
- [36] Yapıcı, Y., Demir, B., Ertürk, S., Urhan, O., "Downsampling Based Multiple Description Coding with Optimal Reconstruction Filters", *IEEE Signal Processing and Communications Applications Conference*, (2007).
- [37] Hestenes M.R. and Stiefel, E., "Methods of Conjugate Gradients for Solving Linear Systems", *Journal of Research of the National Bureau of Standards*, 49(6), 409-436, (1952).
- [38] Romero, S., Romero, L.F., "An Optimized Preconditioned Conjugate Gradient Algorithm", *Technical Report No: UMA-DAC-02/11*, (2002).  
[www.ac.uma.es/inv-des/inves/reports/2002/UMA-DAC-02-11.ps](http://www.ac.uma.es/inv-des/inves/reports/2002/UMA-DAC-02-11.ps) (Ziyaret tarihi: 10 Ekim 2007).
- [39] Servetto, S.D., Ramchandran, K., Vaishampayan, V.A., Nahrstedt, K., "Multiple Description Wavelet Based Image Coding," *IEEE Trans. on Image Processing*, 9(5), 813-826, (2000).
- [40] Vaishampayan, V. A. and John, S. "Balanced Interframe Multiple Description Video Compression", *Proc. of IEEE Int. Conf. on Image Processing (ICIP)*, 3, 812-816, (1999).



- [41] Regunathan, S. L. and Rose, K. "Efficient Prediction in Multiple Description Video Coding", *Proc. of IEEE Int. Conf. on Image Processing (ICIP)*, 1, 1020-1023, (2000).
- [42] Nathan, R. and Zamir, R., "Multiple Description Video Coding with Unquantized Prediction Loop", *Proc. of IEEE Int. Conf. on Image Processing (ICIP)*, 1, 982-985, (2001).
- [43] Reibman, A.R., Jafarkhani, H., Wang, Y., Orchard, M.T. and Puri, R., "Multiple Description Coding of Video Using Motion Compensated Prediction", *Proc. of IEEE Int. Conf. on Image Processing (ICIP)*, 3, 837-841, (1999).
- [44] Boulgouris, N.V., Zachariadis, K.E., Leontaris, A.N. and Strintzis, M.G., "Drift-free Multiple Description Coding of Video", *Proc. of IEEE 4th Workshop on Multimedia Signal Processing*, 105-110, (2001).
- [45] Tang, X. and Zakhor, A., "Matching Pursuits Multiple Description Coding for Wireless Video", *IEEE Trans. Circuits Syst. Video Technol.*, 12(6), 566-575, (2002).
- [46] Lee, Y.C., Altunbaşak, Y., and Mersereau, R.M., "A Drift-free Motion-Compensated Predictive Encoding Technique for Multiple Description Coding", *Proc. of Int. Conf. on Multimedia and Expo (ICME)*, 3, 581-584, (2003).
- [47] Ateş, Ç., Ürgün, Y., Demir, B., Urhan, O. and Ertürk, S., "Polyphase Downsampling Based Multiple Description Image Coding Using Optimal Filtering with Flexible Redundancy Insertion", *Proc. of IEEE Int. Conf. on Signals and Electronic Systems (ICSES)*, 193-196, (2008).
- [48] Yapıcı, Y., Demir, B., Ertürk, S. and Urhan, O. "Down-Sampling Based Multiple Description Image Coding Using Optimal Filtering", *Journal of Electronic Imaging*, 17(3), (2008).
- [49] Kim, D.H., Oh, Y., Urhan, O., Ertürk, S. and Chang, T.G., "Optimal Post Process/in-loop Filtering for Improved Video Compression Performance", *IEEE Trans. Consumer Electron.*, 53(4), 1687-1693, (2007).
- [50] Akbulut, O., Urhan, O., Ertürk, S. and Chang, T.G., "One-Dimensional Processing for Efficient Optimal Post-Process/in-loop Filtering in Video Coding", *IEEE Trans. Consumer Electron.*, 54(3), 1346-1354, (2008).
- [51] Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, "Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264/ISO/IEC 14496-10 AVC)", *JVT-G050*, (2003).
- [52] <http://iphome.hhi.de/suehring/tml/download/> (*Ziyaret Tarihi: 10 Eylül 2007*).
- [53] Akbulut, O., Ertürk, S., "Optimal Son-İşlem veya Döngü-İçi Video Süzgeçlemesi İçin Tek-Boyutlu Süzgeçleme ve Süzgeç Katsayılarının Sıkıştırılması", *IEEE Sinyal İşleme ve İletişim Uygulamaları Kurultayı*, (2008)

## **KİŞİSEL YAYINLAR ve ESERLER**

1. ATEŞ, Ç., ÜRGÜN, Y., DEMİR, B., URHAN, Y., ERTÜRK, S., "Polyphase downsampling based multiple description image coding using optimal filtering with flexible redundancy insertion", *IEEE Int. Conf. On Signal and Electronic Systems*, 193-196, September 2008, Poland.
2. ATEŞ, Ç., URHAN, O., ERTÜRK, S., "Çok Aşamalı Aşağı Örnekleme Temelli Çoklu Tanımlamalı Video Kodlama için En Uygun Süzgeçleme", *XVII IEEE Sinyal İşleme ve İletişim Uygulamaları Kurultayı*, 676-679, Nisan 2009, Antalya.

## **ÖZGEÇMİŞ**

1983 yılında İzmir’de doğdu. İlk, orta ve lise öğrenimini İzmir’de tamamladı. 2002 yılında girdiği Kocaeli Üniversitesi Mühendislik Fakültesi Elektronik ve Haberleşme Mühendisliği Bölümü’nden 2007 yılında Elektronik ve Haberleşme Mühendisi olarak mezun oldu. 2007 yılında Kocaeli Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Haberleşme Ana Bilim Dalı’nda Yüksek Lisans öğrenimine başladı. 2009 yılında mezun olma durumundadır.

2009 Şubat başından itibaren Kocaeli Üniversitesi Mühendislik Fakültesi Elektronik ve Haberleşme Bölümü Elektronik Ana Bilim Dalı’nda Araştırma Görevlisi olarak görev yapmaktadır.