

Video Sıkıştırma Çoklu Hareket Vektörlerinin Kullanılması

Çağrı Kılınç

DOKTORA TEZİ

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Ocak 2024

Use of Multiple Motion Vectors in Video Compression

Çağrı Kılınç

DOCTORAL DISSERTATION

Department of Electrical-Electronics Engineering

January 2024

Video Sıkıştırma Çoklu Hareket Vektörlerinin Kullanılması

Çağrı Kılınc

Eskişehir Osmangazi Üniversitesi

Fen Bilimleri Enstitüsü

Lisansüstü Eğitim ve Öğretim Yönetmeliği Uyarınca

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Telekomünikasyon-Sinyal İşleme Bilim Dalında

DOKTORA TEZİ

Olarak Hazırlanmıştır.

Danışman: Doç. Dr. Erol SEKE

Ocak 2024

ONAY

Elektrik-Elektronik Mühendisliđi Anabilim Dalı Doktora öđrencisi Çađrı Kılınç'ın DOKTORA tezi olarak hazırladıđı “Video Sıkıştırımda Çoklu Hareket Vektörlerinin Kullanılması” başlıklı bu çalışma, jürimizce lisansüstü yönetmeliđin ilgili maddeleri uyarınca deđerlendirilerek oyçokluđu ile kabul edilmiřtir.

Danışman : Doç. Dr. Erol SEKE

İkinci Danışman : -

Doktora Tez Savunma Jürisi:

Üye: Doç. Dr. Erol SEKE

Üye: Prof. Dr. M. Bilginer GÜLMEZOĐLU

Üye: Prof. Dr. Atakan DOĐAN

Üye: Dr. Öğr. Üyesi Faruk DİRİSAĐLIK

Üye: Dr. Öğr. Üyesi Can UYSAL

Fen Bilimleri Enstitüsü Yönetim Kurulu'nun tarih ve
..... sayılı kararıyla onaylanmıştır.

Prof. Dr. Fatma TÜMSEK

Enstitü Müdürü

ETİK BEYAN

Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü tez yazım kılavuzuna göre, Doç. Dr. Erol SEKE danışmanlığında hazırlamış olduğum “Video Sıkıştırma da Çoklu Hareket Vektörlerinin Kullanılması” başlıklı Doktora tezimin özgün bir çalışma olduğunu; tez çalışmamın tüm aşamalarında bilimsel etik ilke ve kurallarına uygun davrandığımı; tezimde verdiğim bilgileri, verileri akademik ve bilimsel ilke ve kurallara uygun olarak elde ettiğimi; tez çalışmamda yararlandığım eserlerin tümüne atıf yaptığımı ve kaynak gösterdiğimi ve bilgi, belge ve sonuçları bilimsel etik ilke ve kurallara göre sunduğumu beyan ederim. 02/01/2024

Çağrı Kılınç

İmza

ÖZET

Son zamanlarda teknolojiye meydana gelen deęişim ve gelişimlerle beraber video sıkıştırma işlemlerinde farklı standart ve algoritmaların kullanım ihtiyacı artmıştır. Bu nedenle video sıkıştırma işlemlerinde çoklu hareket vektörlerinin kullanılmasındaki temel amaç; çerçeveler arası yapılacak olan hareket kestirimi sürecinde orijinal bloęa en benzer olan bloęu oluşturmaktır. Kullanılan bu yöntem, kodlayıcı (encoder) tarafındaki birden fazla (çoklu) referans bloklarının ağırlıklarını hesaplamak amacıyla ek karmaşıklık (hesaplama) ve hareket vektörlerinin gönderilmesi için ekstra bit kullanılmasını gerektirse de özellikle karmaşık ve ayrıntılı sahnelerde birden fazla referans blok kullanılması daha doğru hareket tahminleri sağlayarak hareket bulanıklığını azaltabilir, kaliteli görüntüler elde edilmesine imkan sunabilir.

Doktora tez çalışmasında, en küçük kareler yöntemi (least squares) ile çoklu hareket vektörlerinin ağırlıklarının hesaplanmasında farklı testler uygulanmıştır. Bu testler üç ana noktada toplanmış olup; birinci nokta her bir ağırlığın en küçük kareler yöntemi ile hesaplanması, ikinci nokta ağırlıklardan birinin sabit tutulup kalan ağırlıkların hesaplanması, üçüncü nokta ise hesaplanmış olan ağırlıklar kümesinde K-means algoritması ile öbekleme (gruplama) yaparak ağırlık setlerinin bulunmasını içermektedir. Yapılan uygulamalar, video sıkıştırmada çoklu hareket vektörlerinin kullanımının video görüntü kalitesinde artış gösterdiğini ortaya koymuştur.

Anahtar Kelimeler: Video sıkıştırma, hareket vektörü, blok eşleştirme, çoklu referans.

SUMMARY

With the recent advancements and changes in technology, the demand for different standards and algorithms in video compression has increased. Therefore, the primary purpose of using multiple motion vectors in video compression is to generate the block that is most similar to the original block during the motion estimation process between frames. Although the method used requires additional complexity (computation) to calculate the weights of multiple reference blocks on the encoder side and extra bits to transmit the motion vectors, using multiple reference blocks, particularly in complex and detailed scenes, can provide more accurate motion predictions, reduce motion blur, and enable the acquisition of clear and high quality images.

In the doctoral thesis, various tests were conducted to calculate the weights of multiple motion vectors using the least squares method. These tests are summarized in three points; the first point involves calculating each weight using the least squares method, the second point involves keeping one weight constant and calculating the remaining weights, and the third point involves clustering the calculated weights using the K-means algorithm to find sets of weights. Through the applied tests and evaluations, it has been observed that the use of multiple motion vectors in video compression leads to an improvement in video image quality.

Keywords: Video compression, motion vectors, block matching, multiple references.

TEŐEKKÜR

Doktora eđitimim boyunca desteklerini esirgemeyen; alıŐma s¼recimin planlanmasında, iŐleyiŐinde ve deđerlendirilmesinde yardımcı olan danıŐmanım Do. Dr. Erol SEKE hocama, tez izleme komitesinde yer alan Prof. Dr. M. Bilginer G¼LMEZOĐLU ve Dr. Öğr. Üyesi Muammer AKAY hocalarıma teŐekkür ederim. Lisans¼st¼ eđitim s¼recimde manevi destekleri ile her zaman yanımda olan, teŐvik eden ve zorlandığım durumlarda fikir alışveriŐinde bulunduđum alıŐma arkadaşlarıma, eŐime ve aileme minnettarım.

İÇİNDEKİLER

	<u>Sayfa</u>
ÖZET	vi
SUMMARY	vii
TEŞEKKÜR	viii
İÇİNDEKİLER	ix
ŞEKİLLER DİZİNİ	xi
ÇİZELGELER DİZİNİ	xiii
SİMGELER VE KISALTMALAR DİZİNİ	xv
1. GİRİŞ VE AMAÇ	1
2. LİTERATÜR ARAŞTIRMASI	3
3. VIDEO SIKIŞTIRMA SİSTEMLERİ	11
3.1. Durgun Resim Kodlama	11
3.2. Video Sıkıştırma Aşamaları	16
3.2.1. Tahminleme	16
<u>3.2.1.1. Çerçeveler Arası Tahminleme</u>	<u>17</u>
<u>3.2.1.2. Blok Eşleştirme</u>	<u>17</u>
3.2.2. Dönüşüm Aşaması	18
3.2.3. Nicemleme	19
3.2.4. Entropi Kodlama	21
4. ÖNERİLEN YÖNTEM	22
4.1. En Küçük Kareler Yöntemi ile Dinamik Ağırlıklandırma	25
4.2. En Küçük Kareler Yöntemi ile Statik Ağırlıklandırma	27

4.3. En Küçük Kareler Yöntemi ile İndekslenirilmiş Ağırlıklandırma.....	27
5. BULGULAR VE TARTIŞMA	30
5.1. Veri Setleri.....	31
5.1.1. Matlab Ortamında Kullanılan Veri Setleri	31
5.1.2. JM 19.1 Ortamında Kullanılan Veri Setleri.....	33
5.2. Matlab Ortamında Yapılan Benzetimler.....	37
5.2.1. Tekli ve Çoklu Referans Kullanımı ile Benzetim Sonuçları	38
<u>5.2.1.1. Dinamik Ağırlıklandırma İle Benzetim Sonuçları</u>	<u>38</u>
<u>5.2.1.2. Statik Ağırlıklandırma İle Benzetim Sonuçları</u>	<u>41</u>
<u>5.2.1.3. İndekslenirilmiş Ağırlıklandırma İle Benzetim Sonuçları</u>	<u>45</u>
5.2.2. Karşılaştırma.....	49
5.3. JM 19.1 Ortamında Yapılan Benzetimler.....	52
5.3.1. 1080p Video Setleri ile Yapılan Benzetimler.....	52
5.3.2. 720p Video Setleri ile Yapılan Benzetimler.....	63
5.3.3. 360p Video Setleri ile Yapılan Benzetimler.....	67
6. SONUÇ VE ÖNERİLER.....	71
KAYNAKLAR DİZİNİ.....	73

ŞEKİLLER DİZİNİ

<u>Sekil</u>	<u>Sayfa</u>
3.1: JPEG standardının kodlayıcı ve çözücü kısımları ve aşamaları (Rojas vd.den, 2006)	12
3.2: Four People video setinden örnek bir çerçeve.....	13
3.3: Beyaz kare içerisine alınmış 8x8'lik blok örneği	14
3.4: JPEG aşaması, (a) örnek bloğun değerleri, (b) artık bloğun değerleri	14
3.5: AKD aşamasının sonucunda elde edilen katsayı değerleri	15
3.6: Nicemleme işlemi sonucunda elde edilen değerler	15
3.7: Zikzak sıralama örüntüsü örneği	16
3.8: 3 adım arama algoritması (Samet vd.den, 2015).....	18
3.9: Nicemleme tablosu	20
4.1: Tek referans bloğu ile fark bloğunun oluşturulması	22
4.2: Birden fazla referans bloğu kullanarak fark bloğunun oluşturulması	24
5.1: Video dizilerinden örnekler, (a) Four People, (b) Kristen & Sara, (c) Mob. Cal., (d) Parkrun, (e) Shields ve (f) Stockholm.....	32
5.2: (a) Aerial, (b) Boat, (c) Crowd Run ve (d) Dinner Scene Cropped videolarından kesitler.....	33
5.3: (a) Controlled Burn, (b) Driving POV, (c) Johnny, (d) Kristen & Sara, (e) Roller Coaster, (f) Vidyo3, (g) Vidyo4 ve (h) West Wind East videolarından kesitler.....	35
5.4: (a) Blue Sky, (b) Red Kayak, (c) Snow Mountain, (d) Speed Bag, (e) Stockholm ve (f) Touchdown Pass videolarından kesitler.....	37
5.5: Tüm setler için dinamik ağırlıklandırma ile elde edilen sonuçlar, (a) Four People, (b) Parkrun, (c) Kristen & Sara, (d) Shields, (e) Mob. Cal., (f) Stockholm	49
5.6: Statik ağırlıklandırma için elde edilen sonuçlar, (a) Four People, (b) Parkrun, (c) Kristen & Sara, (d) Shields, (e) Mob. Cal., (f) Stockholm	50
5.7: İndekslenirilmiş ağırlıklandırma için elde edilen sonuçlar, (a) Four People, (b) Parkrun, (c) Kristen & Sara, (d) Shields, (e) Mob. Cal., (f) Stockholm	51
5.8: Aerial, Boat, Crowd Run ve Dinner Scene video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları	53

ŞEKİLLER

<u>Sekil</u>	<u>Sayfa</u>
5.9: Food Market, Meridian Talk, Motorcycle ve Mountain Bike video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları. 55	
5.10: Old Town Cross, Pedestrian Area, Ritual Dance ve Riverbed video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları	57
5.11: Rush Field Cuts, Skater, Toddler Fountain ve Trees & Grass video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları	59
5.12: Tunnel Flag, Vertical Bees, Vertical Carnaby ve Walking In Street video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları	60
5.13: World Cup ve World Cup Far video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları.....	61
5.14: Controlled Burn, Driving POV, Johnny ve Kristen & Sara video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları. 64	
5.15: Roller Coaster, Vidyo3, Vidyo4 ve West Wind Easy video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları.....	65
5.16: Blue Sky, Red Kayak ve Snow Mountain video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları	68
5.17: Speed Bag, Stockholm ve Touchdown Pass video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları.....	69

ÇİZELGELER DİZİNİ

<u>Cizelge</u>	<u>Sayfa</u>
4.1: Kodlama sürecinin kısa bir özeti	23
4.2: 5 bitlik indeks numaraları ile ifade edilmiş örnek ağırlık setleri.....	28
4.3: 3 bitlik indeks numaraları ile ifade edilmiş örnek ağırlık setleri.....	29
5.1: MATLAB ortamında kullanılan test video dizileri ve açıklamaları.....	31
5.2: 1080p çözünürlük ve 130 kare sayısına sahip video setleri	34
5.3: 720p çözünürlük ve 130 kare sayısına sahip video setleri	36
5.4: 360p çözünürlük ve 130 kare sayısına sahip video setleri	36
5.5: 4x4 blok boyutu ile 1-3 arası referans ile elde edilen PSNR değerleri	39
5.6: 8x8 blok boyutu ile 1-3 arası referans ile elde edilen PSNR değerleri	40
5.7: 16x16 blok boyutu ile 1-3 arası referans ile elde edilen PSNR değerleri	41
5.8: 4x4 blok boyutunda, en benzer bloğun ağırlığının farklı değerlerde sabitlenmesi ile gelen PSNR değerleri.....	42
5.9: 8x8 blok boyutunda, en benzer bloğun ağırlığının farklı değerlerde sabitlenmesi ile gelen PSNR değerleri.....	43
5.10: 16x16 blok boyutunda, en benzer bloğun ağırlığının farklı değerlerde sabitlenmesi ile gelen PSNR değerleri.....	45
5.11: 4x4 blok boyutunda, kümeleme yöntemi ile referans blokların ağırlık bilgisinin 3 ve 5 bit ile temsil edildiği PSNR sonuçlarının, ağırlık bilgisinin ondalıklı olarak ifade edildiği PSNR sonuçlarıyla kıyaslaması	46
5.12: 8x8 blok boyutunda, kümeleme yöntemi ile referans blokların ağırlık bilgisinin 3 ve 5 bit ile temsil edildiği PSNR sonuçlarının, ağırlık bilgisinin ondalıklı olarak ifade edildiği PSNR sonuçlarıyla kıyaslaması	47
5.13: 16x16 blok boyutunda, kümeleme yöntemi ile referans blokların ağırlık bilgisinin 3 ve 5 bit ile temsil edildiği PSNR sonuçlarının, ağırlık bilgisinin ondalıklı olarak ifade edildiği PSNR sonuçlarıyla kıyaslaması	48
5.14: 1080p çözünürlükteki video setler için önerilen yöntem ve JM 19.1 arasındaki farklar	62

Cizelge**Sayfa**

5.15: 720p çözünürlükteki video setler için önerilen yöntem ve JM 19.1 arasındaki farklar	66
5.16: 360p çözünürlükteki video setler için önerilen yöntem ve JM 19.1 arasındaki farklar	67

SİMGELER VE KISALTMALAR DİZİNİ

Simgeler

\hat{B}_p	Tekrar Oluşturulmuş Bir Önceki Blok
B_c	Kodlanan Şu Anki Blok
B_d	Fark Bloğu
P_c	Sıkıştırılacak Şu Anki Çerçeve
3B	Üç Boyutlu
4K	3840x2160 Çözünürlüğü
B	Çift Yönlü Çerçeve
dB	Desibel
I	Çerçeve İçi
P	Tahmini Çerçeve
Q, \bar{Q}	Niceleme-Ters Niceleme
Tr, \bar{Tr}	Dönüşüm-Ters Dönüşüm
$V_{hareket}$	Hareket Vektörü

Kısaltmalar

AKD	Ayrık Kosinüs Dönüşümü
BD	Sinyal Ögelerinin Saniyedeki Atım Oranı
CCITT	Uluslararası Telgraf ve Telefon Danışma Komitesi
CNN	Konvolüsyonel Sinir Ağı
CU	Kodlama Ünitesi
FCN	Tamamen Konvolüsyonel Ağ
FPS	Saniye Başına Düşen Kare
H.264/AVC	Gelişmiş Video Kodlama
H.265/HEVC	Yüksek Verimlilikli Video Kodlama
IEC	Uluslararası Elektroteknik Komisyonu
ISO	Uluslararası Standartlar Teşkilatı
ITU	Uluslararası Telekomünikasyon Birliği
ITU-T	Uluslararası Telekomünikasyon Standartlaştırma Birimi

SİMGELER VE KISALTMALAR DİZİNİ**Kısaltmalar****Açıklama**

JPEG	Ortak Fotoğraf Uzmanları Grubu
JVT	Ortak Video Ekibi
LS	En Küçük Kareler Yöntemi
MB	Makro Blok
MPEG	Hareketli Görüntü Uzmanları Birliği
MSE	Ortalama Karesel Hata
MVP	Çoklu Görünüm Profili
PMD	Nesneler Arasındaki Farklılık/Benzerliği Ölçen Yakınlık Ölçüsü
PSNR	En Yüksek Sinyal Gürültü Oranı
PU	Kestirim Ünitesi
SAD	Mutlak Farkların Toplamı

1. GİRİŞ VE AMAÇ

Günümüz büyük veri çağında, internet üzerinden çeşitli bilgiler hem iletilmekte hem de depolanmaktadır. Bu aktarımlarda video ve görüntü gibi çoklu ortam içerikleri, mevcut web trafiğinin önemli bir bölümünü oluşturmaktadır. Büyük çaplı, yüksek çözünürlüklü veri iletiminden dolayı enerji tüketimi ve ağ bant genişliği kullanımı gibi zorluklarla karşı karşıya kalmaktadır. Çoklu ortam verilerinin etkili bir şekilde yönetilebilmesi, bu zorlukları aşmak için önem kazanmaktadır (Chai vd., 2021).

Modern görüntüleme uygulamalarında sistem genel performansını belirleyen nokta, çoklu ortam veri alımının kalitesi ve sıkıştırma verimliliğidir (Chen vd., 2022). Güvenlik ve sıkıştırma yöntemlerindeki ilerlemeler daha iyi kalite, daha yüksek hız ve daha özel iletişim için giderek artan talebe yönelik uygun çözümler sunmaktadır (Tao vd., 2019). Dolayısıyla sürekli artan video çözünürlükleri ve gerçek zamanlı video iletim talep ve isteği araştırmacıları mevcut standartları iyileştirmeye, geliştirmeye ve yeni standartlar üretmeye zorlamaktadır. Bu çalışmaların bir sonucu olarak video sıkıştırma standartları, başta ITU-T ve ISO/IEC olmak üzere, bu iki kuruluş tarafından atılan adımlarla gelişmiştir. ITU-T H.261 (ITU-T, 1990) ve H.263'ü (ITU-T, 1995) üretirken, ISO/IEC ise MPEG-1 (ISO/IEC, 1993) ve MPEG-4 Visual'ı (ISO/IEC, 1999) üretmiştir. Ardından bu iki kuruluş, H.262/MPEG-2 Video ile H.264/MPEG-4 Advanced Video Coding (AVC) standartlarını oluşturmuştur. Bu iki standart birlikte güçlü bir etkiye sahip olmuş ve günlük hayatımızda giderek yaygınlaşan çeşitli ürünlere yerleştirilmiştir. En son bahsedilen iki kuruluş tarafından üretilen H.265 High Efficiency Video Coding (HEVC) ise en yeni video kodlama standartlarından biridir. HEVC önceki AVC standardından çoğu temel parçayı devralmasına rağmen, 4K veya daha yüksek çözünürlüklerle uğraşırken AVC'ye göre daha verimli video sıkıştırma yapmaktadır. HEVC, AVC ile neredeyse aynı kalitede videoları sıkıştırırken bit hızında %64'e kadar daha fazla verimlilik elde edebilmektedir (McCann vd., 2014). Örneğin; 64x64 makro blokların (MB) kullanılması, yüksek çözünürlüklü videolarda çok daha büyük bit hızı tasarrufları sağlamaktadır.

Tüm video sıkıştırma standartları ve çeşitlilikleri üç ölçülebilir niceliği dengelemek üzerine kuruludur; bit hızı, görüntü kalitesi ve hesaplama kaynakları. Genellikle daha fazla

hesaplama kaynağı kullanımı, daha yüksek görüntü kalitesi ve daha düşük bit hızı anlamına gelir. Yani, bit hızını aynı tutarken benzerlik aramasına daha fazla kaynak ayrıldığında, daha yüksek kaliteli çözümlenmiş görüntüler beklenir. Dolayısıyla bu gelişim sürecinde, her bir standardın beklenen dağıtım zamanında ürünlerde pratik kullanıma uygun olan hesaplama kaynaklarını dikkate alarak, sıkıştırma kapasitesini maksimize etmek ve veri kaybı dayanıklılığı gibi diğer özellikleri geliştirmek için sürekli çaba sarf edilmiştir (Sullivan vd., 2012).

Tez konusunun temel amacı, çoklu referans bloklarında lineer toplamının kullanılmasının etkilerini değerlendirmektir. Çoklu referans blokları kullanımın, PSNR değerlerini yükseltmesi yani görüntü kalitesini artırması beklenmektedir. İlerleyen bölümlerde öncelikle konu ile ilgili literatürdeki çalışmalar irdelenmiş, video sıkıştırma metodlarının hareket kestirimi kısmına ağırlık verilmiş ve bu çalışmalara kısaca değinilmiştir. Video sıkıştırma sistemleri başlığı altında konuya dair teorik bilgiler aktarılmıştır. Ardından bahsedilen problemin işleyişinde ve çözümünde önerilen yöntemler üzerinde durulmuş, en küçük kareler yöntemi ile dinamik, statik ve indekslendirilmiş ağırlıkların nasıl hesaplanması gerektiği anlatılmıştır. Tasarlanan yöntemler iki aşamada test edilmiştir. Birinci aşamada, standartların sunduğu referans yazılımlarının karmaşıklığından sıyrılmak için MATLAB ortamında deneyler yapılmıştır. İkinci aşamada ise H.264 standardının referans yazılımı üzerinde çift yönlü öngörü aşamasına müdahil olup en küçük kareler yöntemi ile referans bloklarının ağırlıklarının hesaplanmasının sonuçları test edilmiştir. Kullanılan veri setleri dahilinde planlanan deneysel çalışmaların çıktıları sunulmuş, bu çıktılar bulgular ve tartışma bölümünde tablolar halinde gösterilmiştir. Son olarak elde edilen sonuçlardan yola çıkılarak genel bir değerlendirme ile veriler yorumlanmıştır.

2. LİTERATÜR ARAŞTIRMASI

Hareket kestirimi, ardışık video kareleri arasındaki hareketli nesnelerin ve piksellerin tahmin edilmesidir. Doğal videolarda ardışık olan video çerçeveleri birbirine benzer piksel değerlerine sahiptir. Benzer çerçeveler birbirlerinden bağlantısız biçimde ya da bir çerçeve kendisinden önceki ve sonraki çerçevelerdeki farklılıkların belirlenmesi ve bu farklılıkların kodlanması ile sıkıştırılabilir. Hareket kestiriminin doğru yapılması, fark çerçevesinin daha az bitle kodlanmasını sağlar ve verimi artırır. (Nasır, 2007)

Bir video karesindeki hareketli nesnelere veya pikseller, bir sonraki karede belirli bir yönde hareket edecek gibi görünür. Hareket tahmini algoritmaları, bu hareketi tahmin etmek için yöntemler (optik akış, piksel özyineleme ve blok tabanlı eşleştirme) kullanır. Bunlardan blok tabanlı hareket tahmini algoritmaları, H.264/AVC ve M-PEG gibi en çok kullanılan, kodlayıcı-kod çözücülerin içerisinde en verimlisidir. (Çimtay, 2012). Blok tabanlı hareket kestirimi, bir kareyi bloklara böler ve her bloğun bir sonraki karedeki konumunu tahmin etmek için blok içindeki piksel değerlerini analiz eder. Hareketli nesnelerin konumlarındaki değişiklikleri tahmin edebilir ve sadece bu değişiklikleri kodlayarak sıkıştırılmış işlevsel bir video oluşturabilir.

Sonuç olarak, günümüz teknolojisinde meydana gelen gelişim ve değişimle beraber çağımızda akıllı telefon, bilgisayar, kamera, tablet, televizyon, ses sistemleri, taşınabilir veri aygıtları (USB bellek ve SD kartlar) vb. bilgi ve görüntü saklanabilen, iletilebilen araçların kullanımını artmaktadır. Bu nedenle video sıkıştırma yöntemlerinde yapılan ve yapılmakta olan iyileştirmelerde önem kazanmaktadır. İyileştirmelerde hedeflenen amaç; belli standart ve algoritmalar kullanarak video boyutlarının verimliliğinden ve işlevselliğinden mümkün olduğunca az ödün vererek daha iyi performans sergilemesini sağlamaktır. Tez kapsamında yapılan literatür araştırmasında video sıkıştırma yöntemlerinin hareket kestirimi kısmına ağırlık verilmiştir. İncelenen çalışmalar literatürde bulunan bazı yaklaşımlara göre aşağıda kısaca özetlenmiştir. Literatür incelemesi, bu tez kapsamında ele alınan konuya en uzak olan çalışmalardan başlayarak, tezde öne sürülen yöntemlere yakın olan çalışmalara doğru aktarılmıştır.

Video kodlama alanında, yapay sinir ağı yaklaşımlarında bulunan çalışmalara örnek olan araştırmalar şu şekildedir. Wang vd.nin (2018) araştırmalarında, birleşik tam bağlantılı ve evrimsel sinir ağı mimarisi tasarlanmıştır. Ele alınan temel fikir, geleneksel ara tahmin yöntemlerinin yerine sinir ağını kullanarak daha iyi tahmin performansı oluşturmaktır. Sinir ağı tabanlı mimari daha karmaşık örüntüleri modelleyebilir ve daha doğru hareket vektörü tahminleri yapabilmektedir. Araştırmada FCN'ye girdi olarak uzamsal ve zamansal komşu pikseller kullanılmıştır. FCN'nin çıktıları CNN'ye girdi olarak kaydedilmiştir. Sonuçlar % 8.6'ya kadar BD oranının azaltılabileceğini göstermiştir. Lee vd. (2020) yaptıkları çalışmada, CNN tabanlı video öngörü ağı tasarlanmıştır. Önceki kodlanmış çerçevelerden sanal çerçeve oluşturulmaktadır. Bu sanal çerçeveler, referans çerçeveleri yerine kullanılmaktadır. Oran-bozulma optimizasyonu sırasında daha iyi öngörülmuş resim üretmesi hedeflenmiştir. HEVC/h.264 standardının RA ayarlarında %2.9, LDB ayarlarında ise %5.7'lik kodlama iyileştirmesi sağlanmıştır. Van ve Nguyen'nin (2020) çalışmasında, kod çözücü tarafında, VVC standardı ile kodlanmış video setleri, çalışmada tasarlanan ağ yapısı ile kalite bakımından iyileştirilmiştir. Jia vd. (2022), FLAVR adlı interpolasyon ağını baz alarak nicemleme parametrelerini girdi olarak kabul eden bir ağ tasarımı oluşturmuştur. Tekrar oluşturulmuş resimler ile beslenen ağ, interpolasyona uğramış referans çerçeveleri oluşturmaktadır. Bu çerçeveler ek referanslar olarak ele alınıp hareket kestirimi ya da hareket telafisinde kullanılmaktadır. Yöntem bit akışında artış olmadan çalışılan video setlerinin Y,U,V bileşenlerinde %2.03, %6.96 ve %6.40'lık kodlama kazanımları sağlamıştır. Brand vd. (2022), yaptıkları çalışmada, kodlayıcının girişine ve kod çözücünün çıkışına eklenen alt-ağlar ile koşullu kodlamaya farklı bir tasarım getirilmiştir. Doğrusal artık ve genelleştirilmiş artık kodlama yaklaşımlarının ikisi de gerçekleştirilip daha iyi sonuç verenin seçildiği bir çözüm üretilmiştir. Artık blokların kodlama evresine odaklanıldığında diğer ağ yapılarından daha iyi sonuç verdiği görülmüştür. %27.8'lik bit hızı kazanımı sağlandığı belirtilmiştir. Bu iyileşmelerin çok az bir karmaşıklık artışı ile elde edildiği vurgulanmıştır.

Video kodlama alanında, hareket denkleme alanında bulunan çalışmalara örnek olan araştırmalar şu şekildedir. Eisert vd. (2000) tarafından yapılan araştırmada, üç boyutlu kafa modellemesi kullanılarak standart blok tabanlı kodlayıcı birleştirilmiştir. Çözücü tarafına öncelikli olarak başına üç boyutlu modellemesi gönderilmektedir. Video konferanslarda kullanılması hedeflendiği için bu tür özel bir durum için tasarlanan yöntem PSNR değerinde artış ve bit hızı kazanımı sağlamaktadır. Yaklaşımları h.263 ile

kıyaslandığında %35 oranında ortalama PSNR artışı sağlamıştır. Aynı PSNR değerlerinde sınıdıklarında öznel olarak kalitenin daha iyi olduğu belirtilmiştir. Kim vd. (2002), kodlama verimini arttırmak için, iyileştirme katmanında ağırlıklı toplam ile öngörü gerçekleştirmiştir. Çoklu referans çerçevesi kullanılmasına yönelik geliştirilen yöntem videoda oluşan kayma problemini de azaltmıştır. Yöntem görüntüdeki kayma problemini baskılamış ve daha H.263 standardından daha iyi kodlama performansı sergilemiştir. Gerçek zamanlı uygulamalar için hızlı küresel hareket kestirimi kullanılmıştır (Keller ve Averbuch, 2003). Gradyan yöntemlerinin uygulandığı bu çalışmada hesaplama karmaşıklığı azaltılmıştır. Murakami vd.nin (2010) yaptığı çalışmada, HEVC standardının atlama, birleşme ve çerçeveler arası modları için hareket vektörü öngörme sisteminde geliştirmeler yapılmıştır. Uzamsal ve zamansal hareket vektörü bilgisini elde etme yöntemleri sunulmuştur. HM-2.0 referans yazılımı ile kıyaslandığında sırayla RAHE, RALC, LDHE ve LDLC ayarlarında ortalama % 1.3, % 1.8, % 1.2 ve % 2.2 bit hızı azalması sağlanmıştır. Kodlama ve kod çözme zamanlarının çok artmadığı söylenmiştir. Zhang vd. (2016) tarafından, fark kalıntılarının dağılımı ile bazı hareket kestirimi işlemlerinin atlandığı yöntem tasarlanmıştır. Yöntem, bir video karesinin bloklarının farklı ara tahmin modları arasında hızlı bir şekilde karşılaştırılmasını içermektedir. Deneysel sonuçlar ile yöntemin yaklaşık % 4'lük bir bit oranı artışıyla birlikte % 77'ye varan oranlarda kodlama süresinden kazanım sağlanabileceğini ortaya koymuştur. Huang vd. (2017) tarafından, çerçeveler arası öngöründe mod karar verme sürecinin karmaşıklığı azaltılmıştır. Kenar tespiti yöntemleri ile basitleştirmişlerdir. Deneysel sonuçlar oran bozulma performansı yaklaşık kalırken kodlama süresinin azaltıldığını göstermektedir. Sonuçlar rasgele erişim ayarlarında %39,5'e varan kodlama süresi azalmasını ve buna karşılık olarak bit hızının % 1,97 oranında arttığını göstermektedir. Öngörü çerçevelerinde düşük gecikme ayarlarında ise %35,2 kodlama süresinde azalma ve %1.89 oranında bit hızında artış gözlemlenmiştir. Jiwani (2018), çerçeve içi tahmin modu için etkili bir algoritma tasarlamıştır. Komşu piksellerden, en küçük kareler yöntemi ile çerçeve içi tahmin modu için kestirim yapılmıştır. Sonuçlar, görsel kalite üzerinde minimal bir etkiyle hesaplama maliyetinde % 30'dan fazla azalma elde edildiğini göstermiştir. Ma vd. (2019) yaptıkları çalışmada, çerçeveler arası öngöründe çoklu referans çerçevesi kullanımı, hafızada fazla yer kaplamasına sebep olmaktadır. Bu durumu engellemek için tekrar oluşturulan çerçevelerden bazı alanlar kesilip arabellekte saklanmıştır. Bu sayede hafıza kullanımı azaltılmıştır. Burada birçok çerçevede tekrar eden benzer içeriklerden yararlanılmak istenmiştir. HEVC referans yazılımı ile yapılan

kıyaslamada, ortalama %5'lik Bjontegaard delta oranı azalması sağlanmıştır. Fan vd. (2019) çalışmasında, Gauss dağılımının genelleştirilmiş versiyonuna dayanarak tahmin sınırı filtrelemesi iyileştirilmiştir. Çalışma, video sıkıştırma sürecinin tahmin aşamasında uzamsal ve zamansal korelasyonu etkili kullanmayı önermektedir. uzamsal korelasyon; bir video karesi içerisinde yer alan pikseller arasındaki benzerlikleri, zamansal korelasyon ise; ardışık kareler arasındaki benzerlikleri ifade eder. Kullanılan yöntem, çerçeve içi ve çerçeveler arası kestirim modlarının birleştirilmesiyle oluşturulmuştur. Sadece çerçeve içi ayarları için % 1,3'lük bit hızı kazanımı; rastgele erişim ayarı için % 1.5, öngörü çerçevelerinde düşük gecikme ayarı için % 2.5 ve çift yönlü çerçevelerde düşük gecikme ayarı için % 1.2'lik bit hızı kazanımları elde edilmiştir. Kodlayıcı çalışma süresi %80'lere kadar artmıştır. Veena ve Rao (2021) çalışmasında ise, video kodlayıcılarında görsel algı ve sıkıştırma arasındaki dengeyi kurabilmek için optimize edilmiş çerçeve içi öngörü kullanmış, bunun için tasarlanan yaklaşımda PSNR, bit oranı ve kodlama süresi açısından geleneksel yaklaşıma göre daha iyi performans göstermiştir.

Küresel hareket denkleme alanında ise yapılan çalışmalara verilebilecek ilk örneklerden birisi, Steinbach vd. (1999) tarafından yapılan çalışmadır. Yaptıkları çalışmada, küresel hareket dengeleme örneklerinde genelde yapılan; tek bir hareket modeli çıkarılması yaklaşımına karşılık çoklu hareket modelleri kullanılması yoluna gidilmiştir. Afin hareket parametrelerini bulmak için en küçük kareler yöntemi kullanılmıştır. Yapılan testler sonucunda, bit oranının % 20 - % 35 arasında azaldığı kanısına varılmıştır. Hızlı çoklu referans çerçevesi hareket kestirimi yöntemi uygulanan Su ve Sun.'ın (2006) çalışmasında, PSNR performansının H.263 standardına göre daha iyi olduğu sergilenmiştir. Itani vd.nin (2010) yaptığı çalışmada, farklı blok boyutlarına uyum sağlayan, doğrudan hareket kestirimi gerçekleştirilmiştir. Kodlayıcı kısmında gerçekleştirilen hareket vektörü seçimi ile % 3.3'lük bit oranında gelişme sağlanmıştır. Suzuki ve Boon (2010) tarafından uygulanan ileriye yönelik çerçeveler arası kestirim yöntemi klasik hareket dengelemeli kestirime göre % 45 daha verimli bulunmuştur. Video sıkıştırmada ardışık video kareleri arasındaki farkları kullanarak düşük gecikmeli ara kare kodlamasının performansı geliştirilmesi hedeflenmiştir. Sampaio vd. (2012) tarafından yapılan çalışmada, HEVC kodlayıcılarının ara tahmin için düşük karmaşıklıkta karar ve hızlı bir hareket vektörü birleştirme algoritması sunmuştur. Ara tahmin aşamasında kullanılan hareket vektörleri düşük karmaşıklıkta birleştirilir ve aynı bölgenin farklı bloklarında bulunan hareket vektörlerini bir araya getirerek, karmaşık hareket

vektörü tahmin birimleri yerine basit karar birimi kullanmayı hedeflemiştir. PU ağaçlarının ara tahmini için sezgisel bir karar alma yöntemi geliştirilen çalışmada % 34'e kadar varan işlem maliyetlerinde azalma ve % 1.9'a varan bit oranında artma gerçekleştiğini göstermiştir. Glantz vd.nin (2010) yaptıkları çalışmada ise, küresel hareket denkleme tabanlı bir öngörü sistemi geliştirilmiştir. Önceden çözümlenmiş resimlerin zamansal filtrelemesi yaklaşımını da içeren çalışmada H.264 referans yazılımı üzerinde tasarlanmıştır. Ortalama % 4,6'lık bit hızı kazanımının sağlandığı çalışmada bazı video setlerinden % 14,87'yi bulan kazanımlarda elde edilmiştir.

Hareket vektörleri ile ilgili kısımlarda yapılan çalışmalar şu şekildedir. Zhang vd.nin (2008) yaptığı araştırmada, H.264 standardının JM8.6 sürümlü referans uygulaması ile yapılan kıyaslamada, daha az hesaplama süresiyle yakın PSNR değerleri elde edilmiştir. Çoklu hareket vektörlerinin ve bu hareket vektörlerinin korelasyonu ile gerçekleştirilen hareket kestirimi sonucunda kalite korunurken kodlama süresinin % 30'a varan değerlerde azaldığı sonucuna ulaşılmıştır. Lin vd. (2011) çalışmasında mevcut hareket bilgisini kullanmak için zamansal hareket kestiricileri oluşturulmuştur. Sonuçlar, kodlama süresinde meydana gelen küçük bir artışla % 2.2'lik bit oranı tasarrufu sağlanabileceğini göstermiştir. Xiong vd. (2014) çalışmalarında, hızlı bir piramit hareket ayrışması temelli CU (ara kod birimi) seçim algoritması sunmuş, HEVC'de CU seçimi sürecini hızlandırmayı hedef olarak belirlemişlerdir. Bu metotta, piramit hareket ayrışması kullanılarak blokların hareket vektörleri elde edilir ardından hareket vektörlerinin farklı ölçeklerdeki dağılımı çözümlenir. Yapılan çözümlenme, hareketli nesnelerin dağılımında bilgi verir. Çalışma sonucunda elde edilen verilerle minimal bir kalite kaybıyla ara-tahmin hızının artırıldığı iddia edilmiştir. Kudo vd. (2016) yapmış olduğu çalışmada, bir video karesinin hareket vektörlerinin, önceki karelerdeki hareket vektörlerinin devamı veya benzeri olarak tahmin edilmesini ifade etmişlerdir. Tahmin sürekliliği, video içeriğindeki hareketin düzgün bir şekilde devam ettiği hipotezine odaklanır ve hareketli nesnelerin sabit bir hız ve yönde hareket ettiği düşünülür. Bu hipotezde, önceki karelerdeki hareket vektörleri, mevcut karedeki hareket vektörlerinin tahmininde kullanılmıştır. Deneysel çalışmalarda H.265/HEVC standardının HM16.6 sürümlü referans yazılımına kıyasla % 1'lik bir bozulma iyileşmesi sağlanmıştır. Bir bloğa sabitlenen hareket vektörlerinin öngörü modunda kodlama verimliliğini arttırdığı gözlemlenmiştir. Blâser vd. (2019) tarafından yapılan çalışmada, çerçeveler arası geometrik öngörü yöntemi tasarlanmıştır. Üçgen öngörü modeli 14 ek geometrik modlarla arttırılarak

nesne sınırında daha verimli kodlama sonuçlarına ulaşılmıştır. Bjontegaard oranı değerlerinde ortalama %0.22'lik iyileşmeler gözlemlenmiştir. Kodlayıcı tarafında çalışma süresi %7 artarken, çözücü tarafında % 2 süre artışı da beraberinde gelmiştir. Asikuzzaman vd. (2020), çalışmalarında hiyerarşik GOP yapısı için kenar tabanlı hareket modellemesi önermişlerdir. Yöntemin devamında referans çerçevesi öngörülür, sonrasında bu referans çerçevesi, tek yönlü ve çift yönlü öngörü çerçevelerindeki referans çerçevesi listelerine eklenmektedir. Test sonuçlarında ortalama Bjontegaard delta oranı incelendiğinde, önerilen yöntemin HEVC yazılımına göre bit hızı kazanımı sağladığı görülmüştür. Brand vd. (2021) tarafından yapılan çalışmada, Farklı hareket vektörü oluşturma yaklaşımları için, çerçeveler arası öngörü işleminde, hareket modelleri arasında geçiş yapabilen, uygun hareket vektörü modelini seçen bir yöntem geliştirilmiştir. Sınıflama yaklaşımı ile her sınıf için uygun hareket modelini seçmektedir. Çalıştıkları tüm veri setlerinde 0.4 dB'lik ortalama bir artış elde edilmiştir. Karmaşık hareket içeren setlerde ise 0.88 dB'lik kazanım olmuştur. Deng vd.'nin (2022) çalışmalarında aday hareket vektörü listelerinden gelen hareket vektörü ve farkları kullanılarak geometrik hareket vektörü farklarının ayrıştırılması gerçekleştirilmiştir. ECM 1.0 rasgele erişim ayarlarında %27'lik BD oranı kazanımı, düşük gecikmeli çift yönlü öngörü ayarında %0.42'lik BD-oranı kazanımı sağlanmıştır. Kodlayıcı tarafında karmaşıklık ve verim arasında makul bir takas ile bu sonuçların elde edildiğini belirtmişlerdir.

Ağırlıklı öngörü alanında yapılan çalışmalar şunlardır. Shen vd. (2004), histogram farkını, parlaklık değişiminin belirlenmesinde ölçüt olarak kullanıp uyum sağlayabilen ağırlıklı öngörü yöntemi tasarlamıştır. Parlaklık değişimi olan video setleri incelendiğinde, kodlama veriminin arttığı gözlemlenmiştir. Aoki ve Miyapnoto (2008) tarafından yapılan çalışmada, Parlaklık azalma ve artma efekti olan videolarda ağırlıklı öngörü için tasarladıkları formül ile parametre kestirimi gerçekleştirilmiştir. Var olan yöntemlerle karşılaştırıldığında bit hızı düşürümü daha az olsa da bu yöntemlerden daha az işlem yükü yarattığından, çalışma süresini kısaltmaktadır. Gerçek zamanlı kodlayıcılarda da kullanılabileceği belirtilmiştir. Erabadda vd.'nin (2020) çalışmasında, uzun vadeli referans çerçevesi, çerçeveler arası öngörü yönteminde kullanılmak üzere sanal referans çerçevesi ürettikleri bir yaklaşım geliştirilmiştir. Deneysel sonuçlar gözlemlendiğinde %2.3'lük ortalama iyileşme sağlandığı BDBR değerlerinde görülmüştür. Sun vd. (2021) çalışmalarında, çerçeve içi öngörüdeki açısal öngörüye benzer bir yaklaşımda bulunmuşlardır. Çerçeveler arası öngörüye yöntemlerini dahil etmişlerdir. Buna ek olarak

hareket vektörü rafine etme yöntemi de uygulanmıştır. Rasgele erişim ayarı için %0.58, düşük gecikmeli çift yönlü öngörü ayarı için %1.48 bit hızında azalma sağlanmıştır. Ek olarak hareket vektörü rafinesi yapıldığında değerler %0.9 ve %2'ye çıkmıştır.

Tez kapsamında çalışılacak konuya yakın çalışmalar, geçmişten günümüze şu şekildedir. Flierl vd. (1998) çalışmada, uygun bir referans bloğu seçme algoritması tasarlanmıştır. Bir kestirim problemi olarak ele alıp o şekilde en uygun referans bloğu seçimi gerçekleştirilmiştir. Referans bloğu sayısı 1'den 2'ye çıktığında 1.5 dB'lik, 1'den 4'e çıktığında ise 2.2 dB'lik kazanım elde edilmiştir. Flierl ve Girod (2003) yaptıkları çalışmada, klasik çift yönlü öngörü ile çoklu referans yaklaşımı kıyaslanmıştır. Çoklu referans yaklaşımları aslında çift yönlü öngörü yöntemini kapsamakta olup yöntem üzerine artı referans çerçeveleri eklenmiştir. Küçük nicemleme parametreleri için öngörü hatasının azaldığı görülmüştür. Yüksek nicemleme parametrelerinde ise çoklu referans tercihleri azalmış ve dolayısıyla etkisi de azalmıştır. Öngörü hatası azaldığında kodlanacak artık bloklarda bit gereksinimini de azalttığından fazladan ek bilgi gönderme kabiliyetini de arttırmaktadır. Yaklaşım hesaplama maliyetini arttırmıştır ve bu yöntemin zayıf noktalarından biri olarak gözükmektedir. Chen vd. (2016) yaptıkları çalışmada, Genelleştirilmiş çift yönlü öngörü geliştirmesi yapılmıştır. Hareket vektörlerinin aranması ve aralık değerlerinin gönderiminde kodlanması ile ilgili çalışmalar gerçekleştirilmiştir. Hareket kestiriminde, toplamı 1 olacak ağırlık çiftleri sabit olarak belirlenmiş ve dahası ağırlıklardan bir tanesi yeterli olmaktadır. İki ağırlığın toplamı 1 olarak kabul edilmiştir, dolayısıyla birinci ağırlık seçildiğinde ikinci ağırlık bu eşitlikten elde edilmektedir. Ağırlıklar $-1/4 \dots 5/4$ aralığında yedi ağırlıktan oluşmaktadır. Ortalama Y bileşeninde Bjontegaard oranı azalması %1 olarak maksimum azalma ise Roller Coaster setinde %3.4 olarak kaydedilmektedir. Daha iyi kestirim sonuçlarının daha az hataya yol açacağı ve bunun da artık kodlamasına yansıtacağı belirtilmektedir. %1.1'lik ortalama kazanım, beraberinde kodlama sürecini iki kat uzatmıştır. Chen vd. (2018) yaptıkları çalışmada, kodlama blokları seviyesinde, önerilen ağırlıklı öngörü ve ortalama alan mod arasında seçim yapabilen bir yöntem oluşturulmuştur. Referansların ağırlıklandırılması, referans blokların buldukları çerçevelerin, şu anki çerçeve uzaklığına göre oluşturulmuştur. Bölme işleminin ağırlıkları toplamı 16 olacak şekilde çerçeve uzaklıklarının oranlarına göre ağırlık tablosu hesaplanmıştır. Deneysel sonuçlara göre önerilen yöntem kullanılan CIF/240p video setleri ele alındığında ortalama %1.59 PSNR ve %1.22 SSIM kazanımı sağlanmıştır. 4 CIF/480p

video setlerinde ise, önerilen yöntemde ortalama %1.48 ve %1.12 PSNR ve SSIM kazanımı sağlanmıştır. Winken vd.'nin (2019) çalışmasında uygulanan çoklu referans yaklaşımında, standartta bulunan çift yönlü öngörü yöntemi temel alınmıştır. Çift yönlü öngörü hesaplaması yapıldıktan sonra ek referans için, ek hareket kestirimi yapılmaktadır. Ek referansın ağırlık parametrelerinin hesabında oran-bozulma maliyeti SAD ve yaklaşık bit hızından yararlanılmıştır. Oran-bozulma maliyeti en iyi bulunandan düşük olana ya da ek referans sayısı limitine kadar hareket kestirimi tekrar edilmektedir. İki ek referanstan fazla eklendiğinde kodlama kazancının azaldığı belirtilmiştir. Ek ağırlıkların uyumlu olarak belirlenmesi ile elde edilen kodlama performansının sabit ağırlıklar kullanılarak elde edilen performanstan daha iyi olduğu gözlemlenmiştir. Yüksek bit hızında daha iyi çalıştığı görülmüştür. Li vd. (2022) Çoklu referans bloklarını kullanmaya dayanan bu yöntemde, eklenen referans bloklarının hareket vektörleri, çözücü tarafında hâlihazırda bulunan vektörlerden elde edilmektedir. Referansların ağırlıklandırılmasında ise üç farklı yaklaşımda bulunulmuştur. Aritmetik ortalama ile ağırlıklandırma, referans 1 çerçevelerin kodlanacak bloğun bulunduğu çerçeveye olan uzaklığına bağlı ağırlıklandırma ve referans blokların kodlanacak blok ile aralarındaki piksel değerlerinin farklarının toplamına bağlı bir yöntem ile ağırlıklandırmadır. AV1 referans yazılımı ile karşılaştırıldıklarında ortalama % 0.8 Bjontegaard delta değerlerinde azalma elde edilmiştir. Bazı veri setlerinde bu değer %3'lere de ulaşmıştır. Bu sonuçlar kodlayıcı kısmında %20 çalışma süresi artışı ile elde edilmiştir.

3. VİDEO SIKIŞTIRMA SİSTEMLERİ

Video sıkıştırma sistemleri, bir video dosyasının ham (orijinal) dosyadan daha az alan kaplayacak ve transfer edilebilecek ağlar üzerinden olabildiğince yüksek kalitede, hızlı ve işlevsel gönderilmesini sağlayan işlemler bütünüdür. Hareketli görüntülerin yanı sıra ses ve diğer veri bileşenlerini içeren karmaşık veri akışlarını da ele alan video sıkıştırma algoritmaları, görüntü ve ses kodlamasını birleştirerek, video verilerinin sıkıştırılarak daha küçük boyutta depolanmasını ve aktarılmasını sağlamaktadır.

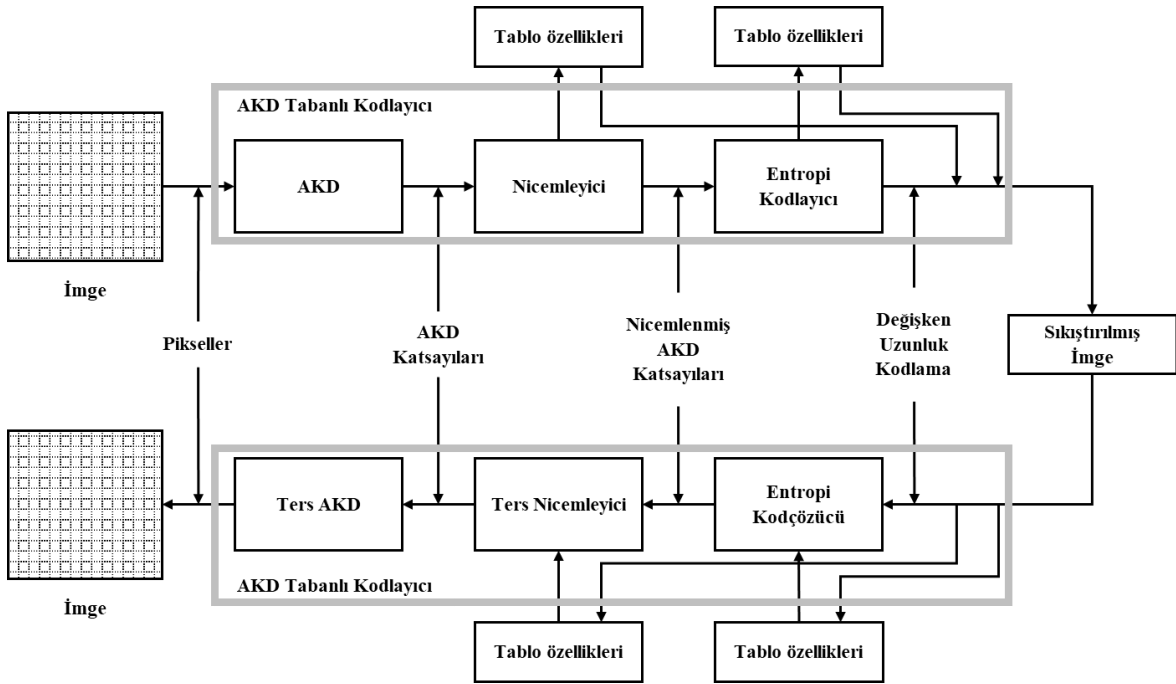
Video sıkıştırma sistemleri, iki yöntem üzerinden uygulanmaktadır. İlk yöntem olan kayıpsız sıkıştırma, sıkıştırılan verinin orijinal haline tam olarak dönebilmesini sağlar. Yani bir dosyanın kalite kaybı olmadan sıkıştırılması ve ayıklandığında tekrardan çalışabilir hale getirilebilmesi işlemidir. Bu yöntemde, bir bitlik değişimlerin bile veri içeriğinin bozulmasına neden olabilecek uygulamalarda ihtiyaç duyulmaktadır (Artuğer ve Özkaynak, 2018). En yaygın kullanılan kayıpsız veri sıkıştırma yöntemi Huffman kodlamasıdır. Diğer bir yöntem olan kayıplı sıkıştırma ise, video dosya boyutunu azaltırken sürekli tekrar eden ses, görüntü verilerini geri dönülemeyecek oranda tamamen ortadan kaldırabilen işlemidir. (Mesut, 2006). Burada temel nokta, veri içeriğinin deforme olmasına neden olmayacak düzeyde silinmelere izin vererek üst seviyede sıkıştırma hedefine ulaşmaktır. Dolayısıyla çoğu uygulamalarda daha kaliteli sıkıştırma oranlarına ihtiyaç duyulduğu için kayıplı sıkıştırma yöntemleri tercih edilmektedir. Yüksek oranda sıkıştırma alanı elde etmek için bu algoritmalar, insan gözünün algısallığını kullanarak veri kaybını minimize etmektedir. Bu nedenle resim görüntü verileri kayıplı sıkıştırma algoritmaları için uygundur.

3.1. Durgun Resim Kodlama

Durgun resim kodlama, bir görüntünün hareketsiz, statik (durağan) bir şekilde temsil edilmesi ve görüntünün daha küçük bir dosya boyutuna sığdırılmasını sağlayan işlemlerdir. Bu algoritmalar, görüntüdeki görsel ve yapısal özellikleri kullanarak; piksellerin değerlerini, tekrarlayan desenleri, renk örüntü ve benzerliklerini gruplandırabilmektedir.

Durgun resim kodlama yöntemlerinden olan en iyi örnek, JPEG (Joint Photographic Experts Group) standardıdır. JPEG, CCITT ve ISO tarafından birlikte oluşturulan, renkli ve gri tonlamalı görüntüleri destekleyen, Şekil 3.1’de görüldüğü gibi farklı aşamalardan meydana gelen bir video kodlama yöntemlerinin bütününden oluşmaktadır.

JPEG kodlama süreci, resmin 8x8’lik bloklara bölünmesi ile başlamaktadır (Artuğer ve Özkaynak, 2018). Bu bloklara ayrıık kosinüs dönüşümü uygulanmaktadır. Bu dönüşümün sonucu olarak 64 adet frekans katsayısı oluşmaktadır. Dönüşümde az da olsa veri kayıpları olabilmektedir. Sonrasında ise asıl veri kayıplarının gerçekleşeceği nicemleme aşaması uygulanmaktadır. Bu aşamada yüksek frekanslı kısımların elenmesi amaçlanmaktadır. Aşamanın sonunda gereksiz kabul edilen çoğu değerin katsayısı 0 olmuştur. Çoğu değerin sıfır olması sağlandığından RLE (Run Length Encoding) uygulanması ile ilk sıkıştırma gerçekleştirilmektedir. Son olarak Huffman ve benzeri entropi kodlaması kullanılarak tekrar bir sıkıştırma yapılmaktadır. RLE ve Huffman kodlamaları kayıpsız kodlamalar olduğu için burada herhangi bir veri kaybı olmamaktadır



Şekil 3.1: JPEG standardının kodlayıcı ve çözücü kısımları ve aşamaları (Rojas vd.den, 2006)

Renkli bir resim örneği üzerinden yapılacak olsaydı, öncelikle klasik RGB(Kırmızı, Yeşil, Mavi) renk uzayından, YUV renk uzayına geçilmektedir. YUV renk uzayında, aydınlık, mavi ve kırmızı seviyelerinin bilgileri bulunmaktadır. İnsan gözü, aydınlık seviyesine, renk seviyelerinden daha duyarlı olduğu için Cb ve Cr (mavi ve kırmızı) renk seviyeleri alt örnekleme tabii tutulmaktadır. Aydınlık seviyesi için çözünürlük 1280x720 iken, örneğin Cb ve Cr için 640x360 çözünürlüğü kullanılabilir. JPEG standardının temel işleyişi, kolaylık olması açısından, gri seviye örnek bir resim üzerinden metnin devamındaki gibi anlatılmaktadır.

Four People video setindeki çerçevelerden birinin dörtte birlik sol üst kısmı Şekil 3.2'deki gibi kullanılmaktadır. Resim öncelikli olarak 8x8'lik bloklara bölünmektedir. Arka plandaki "Natural" kelimesinin ilk harfine denk gelen, beyaz kare içerisinde gösterilen 8x8'lik blok Şekil 3.3'te verilmektedir.



Şekil 3.2: Four People video setinden örnek bir çerçeve



Şekil 3.3: Beyaz kare içerisine alınmış 8x8'lik blok örneği

Örnek blok, 8 bitlik (0-255 arası) değerlerden oluşmaktadır. Blokları oluşturan piksellerin daha küçük değerlerden oluşmasını sağlamak için, tüm piksellerden 128 değeri çıkarılmaktadır. Şekil 3.4'te örnek blok ve çıkartma işleminin sonucunda oluşan artık blokların içerdiği piksel değerleri verilmektedir.

152	138	152	170	178	184	175	174	24	10	24	42	50	56	47	46
114	79	102	140	168	185	174	174	-14	-49	-26	12	40	57	46	46
87	39	64	110	152	184	180	177	-41	-89	-64	-18	24	56	52	49
86	41	70	85	133	181	182	182	-42	-87	-58	-43	5	53	54	54
92	58	90	90	101	160	185	180	-36	-70	-38	-38	-27	32	57	52
98	67	110	110	79	130	182	188	-30	-61	-18	-18	-49	2	54	60
101	64	123	141	84	98	166	193	-27	-64	-5	13	-44	-30	38	65
105	63	132	170	105	71	134	186	-23	-65	4	42	-23	-57	6	58

(a)

(b)

Şekil 3.4: JPEG aşaması, (a) örnek bloğun değerleri, (b) artık bloğun değerleri

Bu aşamadan sonra, artık blok üzerine, AKD uygulanmaktadır. AKD sonrasında, frekansı düşük katsayıların değerleri yüksek çıkarken, frekansı yüksek olan katsayı değerleri düşük çıkmaktadır. Şekil 3.5'te görüldüğü üzere, sol üst köşede yüksek değerler toplanırken sağ alt köşede küçük değerler toplanmaktadır. AKD uygulanırken, sayıların veri tiplerine göre ufak da olsa kayıplar oluşmaktadır. Fakat, asıl veri kayıpları bu aşamada uygulanacak olan nicemleme işleminden sonra gerçekleşmektedir.

9	-263	37	-3	43	73	36	6
89	-18	-47	103	-18	-45	2	-3
71	94	-33	-63	39	7	-15	-3
41	35	26	-16	-37	5	-1	-4
18	23	-5	-2	0	-10	2	4
8	-2	6	-1	-3	0	0	1
1	4	-1	-1	1	1	-2	-2
0	0	-1	0	0	3	-3	-2

Şekil 3.5: AKD aşamasının sonucunda elde edilen katsayı değerleri

Nicemleme işleminde, AKD işleminin çıktısı, yaygın bir şekilde kullanılan nicemleme tablosu ile işleme alınmaktadır. Bu işlem sonucunda elde edilen değerler de aşağı yuvarlanmaktadır. Bu yuvarlama işlemi neticesi ile de geri döndürülemeyen veri kayıpları oluşmaktadır. Nicemleme işleminin sonucu Şekil 3.6'da verilmektedir.

1	-24	4	0	2	2	1	0
7	-2	-3	5	-1	-1	0	0
5	7	-2	-3	1	0	0	0
3	2	1	-1	-1	0	0	0
1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Şekil 3.6: Nicemleme işlemi sonucunda elde edilen değerler

Nicemleme sonucunda, bloğun çoğunluğu 0 değerinden oluşmaktadır. Dolayısıyla, 0 değerlerinin çokluğundan yararlanarak sıkıştırma oranı artırılabilir. Blok içindeki değerlerin, vektör haline dönüştürülüp bit akışı olarak gönderilmesi gerekmektedir. Bu durumda gönderilecek değerler, iyi bir şekilde sıralandığında, değişken uzunluklu kodlama yöntemlerinden alınacak verim artabilmektedir. Değerlerin sıralanması için yaygın olarak Şekil 3.7'deki zikzak örüntüsü kullanılmaktadır.

Coding (HEVC) gibi video sıkıştırma standartlarında video tahmini yöntemleri kullanılmaktadır. Bu standartlarda çerçeve içi ve çerçeveler arası tahminler kullanılmaktadır. Çerçeve içi tahminleme, komşu blokların mekânsal bilgilerini keşfetmektedir (Lainema vd.,2012). Öte yandan, çerçeveler arası tahminleme, önceki kodlanmış çerçevelerdeki gereksiz bilgilere dayanmaktadır (Rhee vd., 2012).

3.2.1.1.Çerçeveler Arası Tahminleme

Tahmin oluşturulduktan sonra, kodlayıcılar bu kodlama sürecinin tahmin hatasını (tahmin artıkları olarak adlandırılmaktadır) dönüştürür ve nicemlendirir. Son olarak, nicemlendirilmiş artıklar ve ek bilgiler entropi kodlanmakta ve bit akışına paketlenmekte olup, böylece kodlama süreci tamamlanmaktadır. Bu yöntemler, videonun içindeki hareketleri ve desenleri analiz ederek gelecekteki kareleri tahmin etmekte ve sıkıştırma algoritması tarafından kullanılarak daha düşük veri oranları elde edilmektedir.

3.2.1.2.Blok Eşleştirme

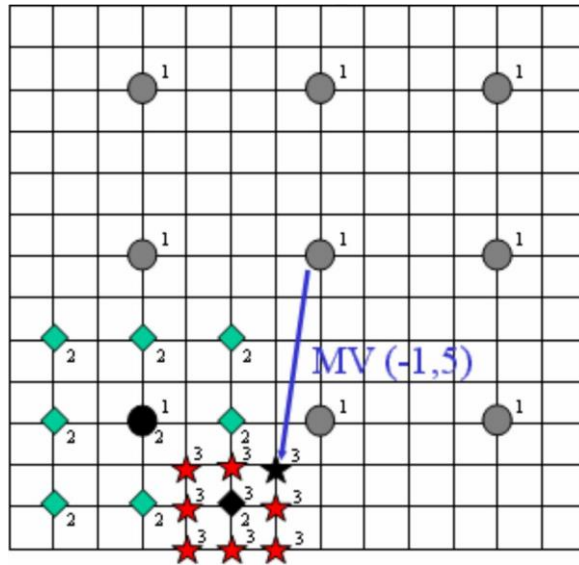
Çerçeveler arası tahminlemede, blok eşleştirme algoritmaları önem arz etmektedir. Blok eşleştirmede, önceden kodlanmış çerçeveler içerisinden kodlanmak istenen bloğa benzer bir blok bulunmak istenmektedir. Bu blok ile kodlanacak blok arasındaki fark az olacağından kodlanacak artık blok daha az boyutta olacaktır. Blok eşleştirme algoritmalarından yaygın olarak kullanılan algoritmalar şunlardır:

- Tam Arama
- 3 Adım Arama

Tam arama, blok eşleştirme algoritmalarından en temel olanıdır. Tarama yapılacak yer, önceden kodlanmış bir çerçeve ya da çerçevenin kodlanmış olan bölgesinden oluşmaktadır. Kodlanacak olan bloğun konumu, arama yapılacak önceden kodlanmış çerçevede, arama için merkez kabul edilmektedir. Her yöne belirli bir piksel uzunluğunda bir arama alanı belirlenmektedir. Genellikle kare ya da dikdörtgen şeklinde bir arama alanı oluşmaktadır. Bu alan içerisinde bir köşeden başlayarak kodlanacak blok iki yöne piksel piksel kaydırılarak benzerlik açısından tüm olasılıklar kontrol edilmektedir. Arama alanında

kodlanacak bloğa en benzer piksel değerlerine sahip konum merkezli blok referans olarak seçilmektedir. İşlem yükü olarak maliyetli olsa da tam arama algoritması arama alanındaki en uygun referans bloğunu bulmaktır.

3 adım aramada da arama alanı, tam arama algoritmasıyla aynı şekilde oluşturulduktan sonra, kodlanacak blok Şekil 3.8'deki gibi arama başlamaktadır. 1 numaralı adımda, ilk arama konumları belirlenmektedir. Bu konumlardan, benzerliğin en çok olduğu konum merkez seçilerek daha dar bir alanda 2 ile numaralandırılmış yeni arama noktaları oluşturulmaktadır. Bu adımda da en benzer konum bulunup 3. adıma geçilmektedir. 3. adımda arama alanı tekrar daralmaktadır. Bu son adımda bulunan en benzer blok referans olarak seçilmekte ve konumu hareket vektörü oluşturmak için kullanılmaktadır. Tüm arama alanında, bulunabilecek en uygun bloğu bulamama ihtimali olsa da işlem yükünün az olması bu algoritma için avantaj sağlamaktadır.



Şekil 3.8: 3 adım arama algoritması (Samet vd.den, 2015)

3.2.2. Dönüşüm Aşaması

Video sinyalinin farklı temsil biçimlerine dönüştürülmesidir. Dönüşüm aşaması, görüntüdeki benzer özellikleri tanımlamak ve bu özelliklerin yüksek frekans bileşenlerini azaltmak için kullanılmaktadır. Dönüşüm aşaması, veriyi farklı bir temsil biçimine dönüştürerek daha az miktarda bilgiyle ifade edilmesinin önünü açmaktadır. En yaygın

kullanılan dönüşüm yöntemlerinden biri, JPEG ve JPEG2000 gibi algoritmalarda kullanılan AKD (Ayrık Kosinüs Dönüşümü) adı verilen bir dönüşüm yöntemidir. 8x8 blok boyutu için yöntem, Denklem 3.1'deki şekilde ifade edilmektedir. C_u ve C_v değerlerinin koşulları ise Denklem 3.2'de gösterilmektedir. Bloklarda sol üst köşe (0,0) olarak kabul edilerek AKD yöntemi uygulanmaktadır.

$$F(u, v) = \frac{1}{4} C_u C_v \sum_{x=0}^7 \sum_{y=0}^7 f(x, y) \cos\left(\frac{(2x+1)u\pi}{16}\right) \cos\left(\frac{(2y+1)v\pi}{16}\right) \quad (3.1)$$

$$\begin{aligned} u = 0 \text{ için,} & \quad C_u = \frac{1}{\sqrt{2}} \\ u \neq 0 \text{ için,} & \quad C_u = 1 \\ v = 0 \text{ için,} & \quad C_v = \frac{1}{\sqrt{2}} \\ v \neq 0 \text{ için,} & \quad C_v = 1 \end{aligned} \quad (3.2)$$

AKD, bir görüntü veya video sinyalinin frekans bileşenlerini temsil etmek için kullanılmaktadır. Bu dönüşüm, görüntünün farklı frekanslardaki bileşenlerini ayırarak, enerji yoğunluğunun düşük frekans bileşenlerinde yoğunlaştığı bilgisinden faydalanmaktadır. AKD, bloklara bölünmüş bir resmin, her bir bloğu için uygulanmaktadır. JPEG için blok boyutu 8x8 piksel boyutlarında seçilmektedir. AKD, her bloğun frekans bileşenlerini hesaplamaktadır. AKD, bir bloğun piksel değerlerini kosinüs fonksiyonlarına dönüştürmekte ve bu fonksiyonları kullanılarak frekans bileşenlerini elde etmektedir. Düşük frekans bileşenleri, bloktaki genel düzeni ve büyük yapısal bileşenleri temsil ederken, yüksek frekans bileşenleri, ayrıntıları ve ince detayları temsil etmektedir.

3.2.3. Nicemleme

Dönüştürülen video sinyalinin hassasiyetini azaltarak veri miktarını azaltma işlemidir. Nicemleme, analog değerleri belli bir aralık içindeki örnek değerlere yaklaştırarak daha az bit kullanımını sağlar. Bu adımda, daha az önemli görüntü bilgileri atılır veya daha az hassas bir temsil kullanılır. Video sıkıştırma nicemleme, bir görüntü veya video sinyalinin hassaslığını azaltarak veriyi sıkıştırma sürecidir. Bu süreçte, yüksek çözünürlüklü

(yüksek hassasiyetli) veri daha düşük çözünürlüklü (daha düşük hassasiyetli) bir forma dönüştürülmektedir. Nicemleme, verinin daha az miktarda bit ile temsil edilmesini sağlayarak veri sıkıştırmasında önemli bir rol oynamaktadır. Nicemleme işleminde, Denklem 3.3'ten yararlanılmaktadır. $C_{u,v}$, dönüşüm aşaması sonucunda elde edilen 8x8'lik bloğun değerlerinden oluşmaktadır. $Q_{u,v}$ değerleri, Şekil 3.9'da verilen nicemleme tablosundan çekilmektedir. Farklı tablolar da kullanılabileceği gibi, bu tablo sıklıkla kullanılanlar arasında yer almaktadır.

$$C_{u,v} = \text{AşağıYuvarla} \left(\frac{C_{u,v} + \frac{Q_{u,v}}{2}}{Q_{u,v}} \right) \quad (3.3)$$

AKD aşamasından sonra, frekans bileşenleri nicemlenmektedir. Bu adımda, frekans bileşenlerinin hassasiyeti azaltılmakta ve daha az miktarda bit kullanılarak temsil edilmektedir. Nicemleme seviyesi, dönüştürülen verinin sıkıştırma oranını ve kalitesini etkilemektedir. Daha yüksek nicemleme seviyesi, daha fazla veri kaybına neden olmakla beraber daha yüksek sıkıştırma oranı da sağlamaktadır. Dönüştürülmüş ve nicemlenmiş frekans bileşenleri daha sonra çeşitli sıkıştırma teknikleriyle kodlanmakta ve saklanmaktadır. Bu kodlama işlemi, dönüştürülmüş verinin sıkıştırma oranını daha da artırmaktadır.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

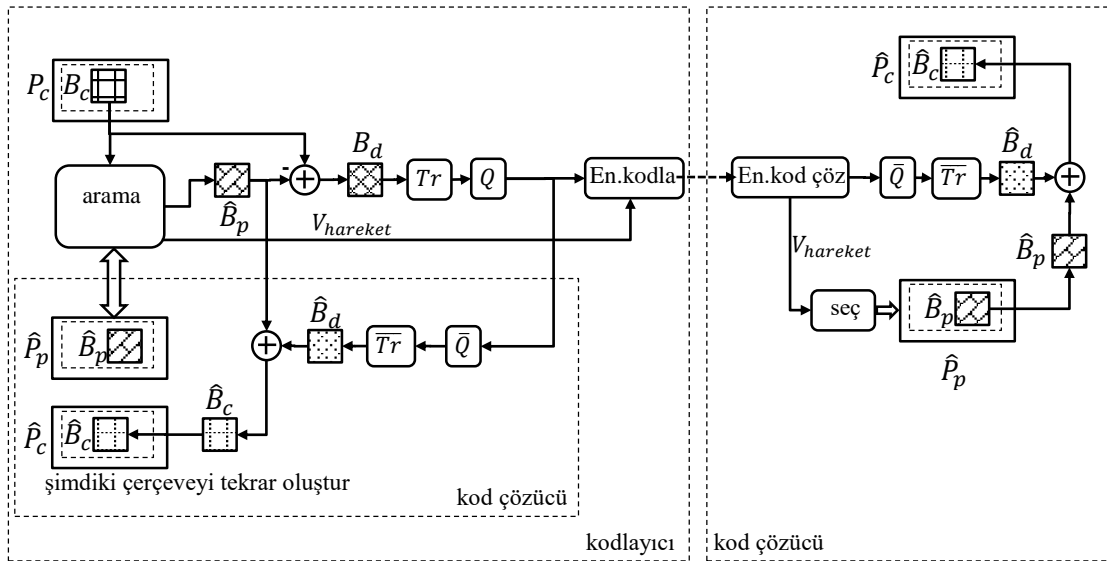
Şekil 3.9: Nicemleme tablosu

3.2.4. Entropi Kodlama

Nicemlenmiş verileri daha verimli bir şekilde temsil etmek için kullanılan yöntemdir. Entropi kodlama, sık sık tekrar eden sembollerin daha kısa bit dizileriyle temsil edildiği bir tekniktir. Amacı, veri dizilerinin istatistiksel yapılarını kullanarak daha az bit kullanarak veriyi temsil etmektir. Entropi kodlaması, sık kullanılan sembolleri daha kısa bit dizileriyle temsil ederken, nadir kullanılan sembolleri daha uzun bit dizileriyle temsil etmektedir. Bu sayede, verinin sıkıştırılmış hali daha küçük bir boyuta sahip olmaktadır. Entropi, bir veri dizisindeki belirsizlik seviyesini temsil etmektedir. Düşük entropi, bir veri dizisinde düşük belirsizlik ve yüksek düzenlilik anlamına gelirken, yüksek entropi daha yüksek belirsizlik ve düzensizlik anlamına gelmektedir. Entropi kodlaması, veri dizisindeki sembollerin olasılıklarını kullanarak daha verimli bir kodlama sağlamaktadır. Öncelikle, sembollerin olasılıkları hesaplanmakta ve sembollerin en yaygın olanları daha kısa kodlara, nadir olanları ise daha uzun kodlara atanmaktadır. Bu kodlar, bir kod tablosunda saklanmakta ve kodlanacak veri dizisi bu kod tablosu kullanılarak sıkıştırılmaktadır. Örneğin, Huffman kodlama ve Aritmetik kodlama gibi yöntemler, tekrarlanan sembolleri daha kısa kodlarla temsil etmek için kullanılmaktadır. Bu sayede veri boyutu daha da azaltılabilmektedir. Huffman kodlaması, veri dizisinin başında kod tablosunu içeren bir başlık eklemektedir. Bu başlık, veriyi doğru şekilde çözebilmek için gerekli olan kodlama bilgilerini içermektedir. Kodlanan veri dizisi daha sonra sıkıştırılmış olarak saklanmakta veya iletilmektedir. Çözme işlemi, kodlanmış veriyi kullanarak kod tablosunu kullanarak orijinal veri dizisini elde etmeyi sağlamaktadır.

4. ÖNERİLEN YÖNTEM

Hareket kestirimi, bit hızı tasarrufunu arttırmaya yardımcı olan adımlardan birisidir. Hareket kestiriminde, mevcut çerçevedeki bloklar, önceden kodlanmış ve çözümlenmiş bloklar içinde araştırılmaktadır. Mevcut çerçevedeki kodlanacak bloğa en benzer olan blok referans olarak alınmaktadır. Kodlanacak blok ve referans blok arasındaki fark dönüşüm aşamasından sonra kodlanmaktadır. Bu bloklar birbirlerine benzer oldukları için aralarındaki fark da küçük çıkmakta ve dolayısıyla dönüşüm ve nicemleme aşamalarından geçmiş artık blok daha az sayıda bit ile temsil edilebilmektedir. Nicemleme aşamasından geçmiş bloklar, göreceli konum (hareket) vektörü ve birkaç ek parametre bilgisi ile birlikte entropi kodlamasına tabii tutulmaktadır. Bu işlemler tüm çerçeveye uygulanmakta ve bit hızında azalmayı sağlamaktadır. Kodlanacak olan bloklar için tek referans bloğu kullanılan video kodlama akışı Şekil 4.1’de verilmektedir.



Şekil 4.1: Tek referans bloğu ile fark bloğunun oluşturulması

Tek referans bloğu kullanıldığında, arama alanında bulunan en benzer blok referans olarak seçilmektedir. En benzer olan blokların bulunmasında Denklem 4.1 ve 4.2’den yararlanılabilmektedir. Bu denklemlerdeki, b ve r sırası ile kodlanacak blokları ve referans

bloklarını ifade ederken, n ve m yatay ve dikey düzlemdeki blok boyutlarını temsil etmektedir.

$$MSE = \frac{1}{nm} \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} |b(i,j) - r(i,j)|^2 \quad (4.1)$$

$$SAD = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} |b(i,j) - r(i,j)| \quad (4.2)$$

Bu çalışmada uygulanan çoklu referans yaklaşımında, bir ya da iki ek benzer olan blok aday olarak alınmaktadır. Çizelge 4.1’de PSNR hesabına kadar olan süreç adımlar halinde aşağıda verilmektedir.

Çizelge 4.1: Kodlama sürecinin kısa bir özeti

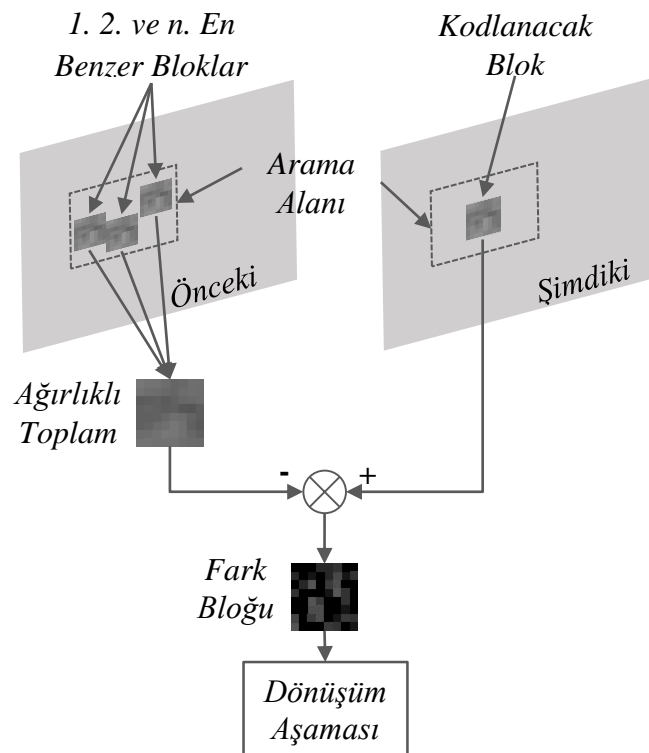
Algoritma	
Adım 1:	<i>Çerçeve 8x8 makro bloklara bölünür.</i>
Adım 2:	<i>Önceki çerçevede, şimdiki çerçevenin sol üst makro bloğuyla başlar.</i>
Adım 3:	<i>Şimdiki çerçeve üzerinde aynı konuma sahip makro bloğu çevresinde bir önceki karede hedef bloğa her yönde 1 blok uzaklıkta olan bir arama alanı oluşturulur.</i>
Adım 4:	<i>Alanda tam bir arama yapılır ve en benzer n tane blok bulunur.</i>
Adım 5:	<i>Bu n tane blok referans blokları olarak kabul edilir.</i>
Adım 6:	<i>Tahmin edilen bloğun bunların doğrusal toplamı olması için en küçük kareler yöntemiyle referans bloklarının ağırlıkları hesaplanır.</i>
Adım 7:	<i>Bu blokların ağırlıklı toplamı ile tahmin edilen bir blok oluşturulur.</i>
Adım 8:	<i>Kalan makro blokları için 2. adımdan 8. adıma kadar olan işlemi tekrarlanır.</i>
Adım 9:	<i>Tahmin edilen çerçevenin tamamı oluşturulur.</i>
Adım 10:	<i>Tahmin edilen çerçeve ile geçerli çerçeve arasındaki PSNR değeri ölçülür.</i>

Kodlanacak blok başına birden fazla referans bloğu kullanıldığında kodlanacak blok ve oluşturulmuş blok arasındaki fark olacağından dönüşüm aşamasını uygulamadan da tekli ve çoklu referans blok yöntemlerinin arasındaki fark karşılaştırılabilmektedir.

Şekil 4.1'deki akıştan farklı olarak Çizelge 4.1'de dönüşüm ve entropi kodlaması aşamaları göz ardı edilmiştir. Buradaki amaç, bit kazanımlarından ziyade resim kalitesini ölçmemizi sağlayan Denklem 4.3'te verilmiş PSNR değerindeki kazançlara odaklanılmaktadır. Denklemdeki \max_p sembolü çerçeve ve bloklardaki maksimum piksel değerini temsil etmektedir.

$$PSNR = 20 \log \left(\frac{\max_p}{\sqrt{MSE}} \right) \quad (4.3)$$

Deneysel çalışmalarda, önceki çerçeveye başvuran video çerçevesi belirlenen blok boyutlarına bölünmektedir. Her blok, önceki çerçevedeki bloğun konumuna merkezlenmiş her yöne bir blok komşuluğundaki bir alan içerisinde SAD metriği kullanılarak araştırılmaktadır. Basitliği nedeniyle tercih edilen SAD metriğinin değeri ne kadar küçükse, bloklar birbirine o kadar benzer çıkmakta ve minimum değeri (tam eşleşme) sıfır olmaktadır. En benzer bloklar, SAD metriği tarafından aday olarak seçilmekte ve Şekil 4.2'de gösterilen çoklu referans blokları yaklaşımı için kullanılmaktadırlar.



Şekil 4.2: Birden fazla referans bloğu kullanarak fark bloğunun oluşturulması

4.1. En Küçük Kareler Yöntemi ile Dinamik Ağırlıklandırma

Şu anki çerçevede kodlanacak bloğun önceki çerçeveden seçilen blokların ağırlıklı toplamı ile kestirimi gerçekleştirilmektedir. Kodlanacak olan blok ve ağırlıklı toplam ile elde edilmiş blok arasındaki fark ile artık blok oluşturulmaktadır. Bu artık blok dönüşüm aşamasından sonra entropi kodlaması uygulanacak olan bloktur. Sıkıştırılmış veri, dönüştürülmüş ve nicemleme uygulanmış fark bloğunu, çoklu hareket vektörlerini ve referans blokların ağırlıklarını içermektedir. Kodlanacak bloğa en yakın olan lineer toplamı oluşturmak için referans bloklarının ağırlıkları en küçük kareler yöntemi ile hesaplanmaktadır.

Çerçevelerin 8x8'lik bloklara bölüldüğü bir örnek üzerinden gidecek olursak seçilmiş olan bloklar (b_1, b_2, \dots, b_n) ve kodlanacak blok (b) Denklem 4.4'teki matris formlarından

$$\begin{bmatrix} b_{1_{11}} & \dots & b_{1_{18}} \\ \vdots & \ddots & \vdots \\ b_{1_{81}} & \dots & b_{1_{88}} \end{bmatrix}, \dots \begin{bmatrix} b_{n_{11}} & \dots & b_{n_{18}} \\ \vdots & \ddots & \vdots \\ b_{n_{81}} & \dots & b_{n_{88}} \end{bmatrix} \quad (4.4)$$

64x1'lik Denklem 4.5'te gösterilen vektörlere dönüştürülmektedir.

$$\begin{bmatrix} b_{1_{11}} \\ \vdots \\ b_{1_{18}} \\ b_{1_{21}} \\ \vdots \\ b_{1_{28}} \\ \vdots \\ \vdots \\ b_{1_{88}} \end{bmatrix}, \begin{bmatrix} b_{2_{11}} \\ \vdots \\ b_{2_{18}} \\ b_{2_{21}} \\ \vdots \\ b_{2_{28}} \\ \vdots \\ \vdots \\ b_{2_{88}} \end{bmatrix}, \dots \begin{bmatrix} b_{n_{11}} \\ \vdots \\ b_{n_{18}} \\ b_{n_{21}} \\ \vdots \\ b_{n_{28}} \\ \vdots \\ \vdots \\ b_{n_{88}} \end{bmatrix}. \quad (4.5)$$

Daha sonra en küçük kareler ile Denklem 4.6'daki doğrusal denklem oluşturulmaktadır.

$$A x = b \quad (4.6)$$

Denklemden 4.7’de görüldüğü üzere A matrisi vektör haline dönüştürülmüş seçilmiş bloklardan \hat{x} vektörü lineer toplamda kullanılacak blokların ağırlıklarından, b vektörü ise kodlanmak istenen (yani şimdiki çerçevedeki asıl blok) bloktan oluşmaktadır.

$$A = \begin{bmatrix} b_{111} & b_{211} & & b_{n11} \\ \vdots & \vdots & & \vdots \\ b_{118} & b_{218} & & b_{n18} \\ b_{121} & b_{221} & \dots & b_{n21} \\ \vdots & \vdots & & \vdots \\ b_{128} & b_{228} & & b_{n28} \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ b_{188} & b_{288} & & b_{n88} \end{bmatrix}, \hat{x} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix}, b = \begin{bmatrix} b_{11} \\ \vdots \\ b_{18} \\ b_{21} \\ \vdots \\ b_{28} \\ \vdots \\ b_{88} \end{bmatrix} \quad (4.7)$$

Hesaplanacak ağırlıklardan oluşan \hat{x} vektörü, yalnız bırakılıp Denklem 4.8’deki denklem elde edilmektedir.

$$\hat{x} = (A^T A)^{-1} A^T b. \quad (4.8)$$

Bu işlemlerden sonra tahmin edilen blok Denklem 4.9’da gösterildiği gibi referans bloklarının ağırlıklı toplamıyla oluşturulmaktadır.

$$\hat{b} = A\hat{x}. \quad (4.9)$$

Kodlanacak blok ve tahmin edilen blok arasındaki fark, Denklem 4.10’daki gibi hesaplanarak artık blok elde edilmektedir.

$$res = b - \hat{b}. \quad (4.10)$$

Bu aşamadan sonra sırasıyla dönüşüm, nicemleme ve entropi kodlama aşamaları gerçekleştirilmektedir.

çerçevedeki her bir blok için en küçük kareler yöntemi ile hesaplanmış ağırlık üçlüleri, öbekleme yaklaşımı ile istenilen sayıda gruplara bölünmektedir. Bu kümeler, N toplam küme sayısı olmak üzere, 1'den N'ye kadar indekslenmektedir. Küme sayısının 32 olarak seçildiği durumda K-means algoritması ile bulunan olası ağırlık setleri Çizelge 4.2'deki şekilde oluşturulmaktadır.

Çizelge 4.2: 5 bitlik indeks numaraları ile ifade edilmiş örnek ağırlık setleri

İndeks	w1	w2	w3
0	0,14	0,43	0,43
1	1,00	0,00	0,00
2	1,03	-0,80	0,79
3	-0,66	0,87	0,82
4	0,82	0,21	-0,03
5	0,40	0,35	0,25
6	0,64	0,35	0,01
7	-0,10	0,62	0,48
8	0,26	0,49	0,25
9	0,80	0,05	0,15
10	4,53	-2,16	-1,23
11	1,00	0,96	-0,96
12	0,25	0,72	0,04
13	1,62	-0,02	-0,59
14	0,51	0,20	0,30
15	0,99	0,47	-0,45
16	0,53	0,31	0,16
17	1,31	-0,37	0,05
18	1,08	0,13	-0,21
19	0,61	0,75	-0,35
20	-0,30	1,78	-0,45
21	0,65	0,03	0,32
22	0,45	0,45	0,09
23	0,46	0,06	0,48
24	0,77	0,40	-0,17
25	0,67	0,20	0,14
26	0,32	0,28	0,40
27	0,13	0,10	0,78
28	0,67	-0,23	0,56
29	0,56	0,53	-0,09
30	0,98	3,09	-3,08
31	0,89	-0,17	0,28

Küme sayısının 8 olarak seçildiği durumda K-means algoritmasının öbeklediği ağırlık kümeleri ise Çizelge 4.3'te verilmiştir. Böylece kodlanacak blok için gerekli ağırlık bilgisi, o blok için 3 bit ile sınırlandırılmış olmaktadır.

Çizelge 4.3: 3 bitlik indeks numaraları ile ifade edilmiş örnek ağırlık setleri

İndeks	w1	w2	w3
0	1,00	0,00	0,00
1	0,74	0,63	-0,36
2	0,46	0,44	0,10
3	0,71	-0,17	0,46
4	0,05	0,53	0,42
5	0,37	0,28	0,35
6	0,63	0,16	0,21
7	0,73	0,30	-0,03

Sonrasında ağırlık setlerinin olduğu çizelgenin öncelikle kodlanıp çözücü tarafına gönderilmesi gerekmektedir. Bu çizelge bir kere gönderildikten sonra, referans bloklarının ağırlıkları çizelgedeki indeks numaraları aracılığıyla gönderilmektedir. Böylece, ondalıklı sayılar formunda olan ağırlıkları ifade etmek için gerekecek bit sayısı istenilen sayıda sınırlandırılabilir. Tez kapsamında 8 ve 32 üçlü ağırlık setleri oluşturulmuş ve indeks numaraları 3 ve 5 bitlik sayılarla ifade edilebilecek şekilde sınırlandırılmıştır. Deneysel çalışmada ağırlıkların kümelendirilerek gönderilmesi ile dinamik ağırlıklandırma yaklaşımı da karşılaştırılmıştır.

5. BULGULAR VE TARTIŞMA

Deneysel çalışmalar iki aşamadan oluşmaktadır. Birinci aşamada standartların sunduğu referans yazılımlarının karmaşıklığından sıyrılmak için MATLAB ortamında deneyler yapılmıştır. İkinci aşamada ise H.264 standardının JM 19.1 referans yazılımı üzerinde çift yönlü öngörü aşamasına müdahil olunup en küçük kareler yöntemi ile referans bloklarının ağırlıklarının hesaplanmasının sonuçları test edilmiştir. MATLAB ortamında yapılan çalışma, var olan standartların karmaşık yapısından sıyrılmak ve önerilen yöntemlerin hızlı bir şekilde geliştirilip sınanması için yapılmıştır.

Video kodlama, birkaç aşamadan oluşan bir süreçtir. Son aşama, entropi kodlaması (Bkz. Şekil 4.1), açık döngü bir süreç olduğundan, karşılaştırma amaçları için bu aşamayı dahil etmeye gerek yoktur. Gerçek kayıp nicemleme aşamasında oluşmaktadır. Ayrıca $Tr-Q$ çiftinden daha az sayıda nicemleme çıktısı (yani daha az bit) için Tr aşamasına girdiğinin daha küçük sayılardan oluşması beklenmektedir. Yani Tr aşamasına giren artık bloğun değerleri daha küçükse, sıkıştırma oranının daha yüksek olması beklenmektedir. Bu nedenle, görüntü kalitesi deneylerinde, tek ve çoklu referans vektör durumlarının fark değerleri karşılaştırılmıştır. Açıktır ki, $Tr-Q$ aşamalarının uygulandığı ve uygulanmadığı durumlardaki PSNR değerleri aynı olmayacaktır. Ancak, kalite artışını gözlemlemek açısından, bu şekilde bir değerlendirme yeterli olacaktır. Bu yaklaşım, test ve karşılaştırmaları basitleştirmekte ve amaçlanan ana geliştirmenin sonuçlarının incelenmesini kolaylaştırmaktadır. MATLAB ortamında yapılan çalışmalar bu yaklaşım göz önünde bulunarak yapılmıştır.

JM 19.1 referans yazılımında yapılan çalışmalarda Boyce vd. (2018) tarafından yazılan JVET ortak test koşulları takip edilmiştir. Koşullarda bulunan düşük gecikmeli çift yönlü öngörü ayarları kullanılmıştır. Bu sayede test koşulları literatürde de kullanılan belli bir temel üzerine oturtulmuştur. Tez kapsamında önerilen yöntemlerden, en küçük kareler ile çoklu referansların ağırlıklarının dinamik olarak bulunduğu yöntem JM 19.1 referans yazılımı ile karşılaştırılmıştır.

5.1. Veri Setleri

Veri setleri, deneysel çalışmalar iki aşamada gerçekleştirildiği için, MATLAB ortamında kullanılanlar ve JM 19.1 ortamında kullanılanlar olarak iki ayrı alt başlıkta ele alınmaktadır. MATLAB ortamında, geliştirme ve test süreçleri kısa tutulmak istenildiğinden erişimi açık olan video setlerinden tek bir çözünürlükteki az sayıdaki setler tercih edilmiştir. JM 19.1 referans yazılımında ise, literatürde de kullanılan farklı çözünürlükteki video seti gruplarından yararlanılmıştır.

5.1.1. Matlab Ortamında Kullanılan Veri Setleri

MATLAB ortamında kullanılan video setleri, genel erişime açık olan “Xiph.org Video Test Media” koleksiyonundan seçilmiştir. Çözünürlük olarak 1280x720 olan video setleri kullanılmıştır. Günümüzde orta seviyede kabul edilebilecek çözünürlükler tercih edilmesinin, önerilen yöntemin test edilmesinde yeterli olacağı düşünülmüştür. Önerilen yöntemlerin test edildiği video setleri Çizelge 5.1’de gösterilmiştir. Videoların içerikleriyle ilgili kısa açıklamalar da çizelgede verilmiştir.

Çizelge 5.1: MATLAB ortamında kullanılan test video dizileri ve açıklamaları

Video Setleri	Çöz.	Kare Sayısı	FPS	Açıklamalar
<i>Four People</i>	1280 x 720	600	60	<i>Dört kişinin, masa başında konuşmasını içeren sahneler bulundurmaktadır. Arka plan posterlerin bulunduğu değişmeyen bir duvar görüntüsüne sahiptir.</i>
<i>Kristen & Sara</i>	1280 x 720	600	60	<i>İki kişinin konuşmasını içeren video setinin arka planı, değişim barındırmayan bir duvar ve asılı bir posterden oluşmaktadır.</i>
<i>Mob. Cal.</i>	1280 x 720	504	50	<i>Video seti, detaylı görüntülere ve değişen sahnelere sahiptir. Eski ahşap bir gemi, takvim ve oyuncak trenden oluşan sahneler içermektedir.</i>
<i>Parkrun</i>	1280 x 720	504	50	<i>Parkta, şemsiye ile koşan bir adamın görüntüsü yer almaktadır. Arka planda karlı ve yağmurlu bir ağaçlık bulunmakta olup ayrıntılı bir yapıya sahiptir.</i>
<i>Shields</i>	1280 x 720	504	50	<i>Birbirinden farklı ve ayrıntılı motiflere sahip şövalye kalkanlarının önünde yürüyen bir adamın görüntüsünü içermektedir.</i>
<i>Stockholm</i>	1280 x 720	604	50	<i>Kamera kaydırma tekniği ile Stockholm kasabasının görüntüsünü içermektedir. Evleri, kasabadan geçen nehri ve araçları barındırmaktadır.</i>

Deneyler için, video dizisinin ilk 100 karesi gri seviyeli resimlere dönüştürülmektedir. Algoritma, 99 ardışık kare çiftine uygulanmaktadır. Yani, ilk ve ikinci çerçeveler, önceki çerçeve ve geçerli çerçeve olarak ayarlanmaktadır. Benzer şekilde, 2. ve 3. çerçeveler çift olarak alınmakta ve bu şekilde devam etmektedir. Bu şekilde gerçekleştirilen deneysel çalışma, algoritmanın performansını değerlendirmek için bir temel sağlamaktadır. Şekil 5.1’de de görüldüğü üzere, video dizileri, farklı özelliklere sahip gerçek hayat senaryolarını temsil etmektedir.



Şekil 5.1: Video dizilerinden örnekler, (a) Four People, (b) Kristen & Sara, (c) Mob. Cal., (d) Parkrun, (e) Shields ve (f) Stockholm

“Four People” seti, dört kişinin masa başında konuşmasını ve hareketsiz bir arka planı içermektedir. “Kristen & Sara” dizisi de benzer bir şekilde, hareketsiz bir arka plan

önünde konuşan iki kişinin görüntülerini içerisinde barındırmaktadır. "Mob. Cal." dizisi, ahşap bir gemi, takvim ve oyuncak tren gibi değişken sahnelerden oluşmaktadır. "Parkrun" setinde, bir adamın şemsiyesiyle parkta koşması ve arka planda karlı ve yağmurlu bir ormanın detaylı yapısı bulunması gibi detaylı ve hareketli bir sahne yer almaktadır. "Shields" dizisinde, şövalye kalkanları önünde yürüyen bir adamın görüntüsü ve kalkanların detaylı motifleri dikkat çekmektedir. "Stockholm" dizisi ise Stockholm şehrinin farklı karelerini içermekte olup, evler, nehir ve hareket eden araçlar gibi unsurlarıyla ayrıntılı görüntülerden oluşmaktadır.

5.1.2. JM 19.1 Ortamında Kullanılan Veri Setleri

JM 19.1 ortamında yapılan deneyler için, Zhao vd. (2021) tarafından yazılan test koşullarında belirtilen, erişime açık olan AOM (Alliance for Open Media) grubunun video setlerinden 1080p, 720p ve 360p çözünürlükte olanlar kullanılmıştır. 1080p çözünürlükte toplamda 22 adet video seti bulunmaktadır. İlk 4 video seti Şekil 5.2'de örnek olması açısından verilmiştir.



(a)



(b)



(c)



(d)

Şekil 5.2: (a) Aerial, (b) Boat, (c) Crowd Run ve (d) Dinner Scene Cropped videolarından kesitler.

1080p çözünürlükteki videolar incelendiğinde, çoğunluğun fazlaca hareket barındırdığı gözlemlenmiştir. Hareketler, sabit bir arka plan önünde değişim gösteren nesnelere oluştuğu gibi, kameranın kendisinin de konum değiştirmesinden kaynaklı oluştuğu görülmüştür. Çizelge 5.2’de video setleri ve kısa açıklamaları paylaşılmaktadır. *Aerial, Boat, Dinner Scene Cropped, Meridian Talk, Old Town Cross* ve *Vertical Bee* setleri değişimin diğerlerine göre daha az olduğu video setleridir.

Çizelge 5.2: 1080p çözünürlük ve 130 kare sayısına sahip video setleri

Video Setleri	FPS	Açıklamalar
<i>Aerial</i>	60	<i>Kırsal bir arazinin gökyüzünden çekilmiş görüntülerini içermektedir.</i>
<i>Boat</i>	60	<i>Bir tekneden çekilmiş gökyüzünün ve ağaçların görüntüsü bulunmaktadır.</i>
<i>Crowd Run</i>	50	<i>Çayırdaki koşu yapan kalabalık yarışmacı gruplarının görüntüsünü içermektedir.</i>
<i>Dinner Scene Cropped</i>	30	<i>Yemek yiyen iki adamın sahnesini bulundurmaktadır.</i>
<i>Food Market</i>	60	<i>Bir yiyecek pazarının görüntülerinden oluşmaktadır.</i>
<i>Meridian Talk</i>	60	<i>İki adamın karşılıklı konuşmasını içeren sahneler bulunmaktadır.</i>
<i>Motorcycle</i>	30	<i>Ağaçlı yolda giden bir motosikletlinin bakış açısından sahneler içermektedir.</i>
<i>Mountain Bike</i>	30	<i>Toprak bir patikada, bisikletçinin bakış açısından görüntüler içermektedir.</i>
<i>Old Town Cross</i>	50	<i>Gökyüzünden çekilmiş eski bir şehrin görüntüsü bulunmaktadır.</i>
<i>Pedestrian Area</i>	25	<i>Şehirde yürüyen yayaların sahnelerini içermektedir.</i>
<i>Ritual Dance</i>	60	<i>Davul eşliğinde dans eden bir yerlinin görüntülerinden oluşmaktadır.</i>
<i>Riverbed</i>	25	<i>Nehir kıyısında akan su görüntülerini içermektedir.</i>
<i>Rush Field Cuts</i>	30	<i>Futbol sahasına inen taraftarların görüntüleri bulunmaktadır.</i>
<i>Skater227</i>	30	<i>Patenci bir kızın görüntülerini içermektedir.</i>
<i>Toddler Fountain</i>	30	<i>Fıskiyeler arasında koşuturan bir bebeğin sahnelerinden oluşmaktadır.</i>
<i>Trees & Grass</i>	30	<i>Ağaçların ve çimenlerin bulunduğu görüntülerden oluşmaktadır.</i>
<i>Tunnel Flag</i>	60	<i>Bir karayolu tünelindeki görüntülerden oluşmaktadır.</i>
<i>Vertical Bees</i>	30	<i>Çiçeğe konmuş arıların bulunduğu görüntüler içermektedir.</i>
<i>Vertical Carnaby</i>	60	<i>Öz çekim yaparken yürüyen bir adamın görüntülerini içermektedir.</i>
<i>Walking In Street</i>	30	<i>Sokakta yürüyen insanların görüntüleri bulunmaktadır.</i>
<i>World Cup</i>	30	<i>Dünya kupası seyreden insanların görüntülerini içermektedir.</i>
<i>World Cup Far</i>	30	<i>Dünya kupası seyreden insanların uzak görüntülerini içermektedir.</i>

720p çözünürlüklü video setleri ise Şekil 5.3'te paylaşılmıştır. Toplamda 8 adet 720p çözünürlüklü video bulunmaktadır. Bu videoların yarısı, durağan bir arka plan üzerinde konuşmakta olan insanlardan oluşmaktadır.



(a)



(b)



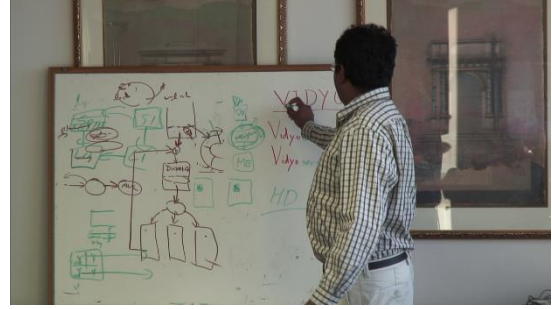
(c)



(d)



(e)



(f)



(g)



(h)

Şekil 5.3: (a) Controlled Burn, (b) Driving POV, (c) Johnny, (d) Kristen & Sara, (e) Roller Coaster, (f) Vidyo3, (g) Vidyo4 ve (h) West Wind East videolarından kesitler.

Aynı çözünürlükteki diğer videolar ile kıyaslandığında *Johnny, Kristen & Sara, Vidyo3* ve *Vidyo4* değişimin az olduğu durağan videolar olarak gözükmetedirler. Çizelge 5.3'te 720p çözünürlükteki video setleri ve açıklamaları paylaşılmaktadır.

Çizelge 5.3: 720p çözünürlük ve 130 kare sayısına sahip video setleri

Video Setleri	FPS	Açıklamalar
<i>Controlled Burn</i>	30	<i>Yanmakta olan tek katlı bir evin görüntüleri yer almaktadır.</i>
<i>Driving POV</i>	60	<i>Çift şeritli bir yolda hareket eden aracın sürücü bakış açısından görüntüler içermektedir.</i>
<i>Johnny</i>	60	<i>Sabit bir arka plan önünde konuşan bir adamın görüntülerinden oluşmaktadır.</i>
<i>Kristen & Sara</i>	60	<i>Sabit bir arka plan önünde konuşan iki kadının bulunduğu görüntüler bulunmaktadır.</i>
<i>Roller Coaster</i>	60	<i>Eğlence parklarında bulunan hız treninden görüntüler içermektedir.</i>
<i>Vidyo3</i>	60	<i>Beyaz tahtada konu anlatan bir adamla ilgili sahneler içermektedir.</i>
<i>Vidyo4</i>	60	<i>Bir toplantı odasında konuşma yapan bir adamın görüntüleri bulunmaktadır.</i>
<i>West Wind Easy</i>	30	<i>Rüzgârda dalgalanan otlar ile ekranın solunda akan yazıların bulunduğu görüntülerden oluşmaktadır.</i>

360p çözünürlükteki videolar ise toplamda 6 adettir ve Çizelge 5.4'te video setleri ve açıklamaları bulunmaktadır. *Snow Mountain* aralarındaki en durağan video seti olarak gözükmetedir. Diğer video setleri, kamera hareketlerine, durağan bir arka plan önünde yer alan nesne hareketlerine sahip gözükmetedir.

Çizelge 5.4: 360p çözünürlük ve 130 kare sayısına sahip video setleri

Video Setleri	FPS	Açıklamalar
<i>Blue Sky</i>	25	<i>Ağaçlar arasından gözükmekte olan mavi bir gökyüzünün görüntülerinden oluşmaktadır.</i>
<i>Red Kayak</i>	30	<i>Hızla akan bir nehirdeki kırmızı kano görüntüleri yer almaktadır.</i>
<i>Snow Mountain</i>	30	<i>Uzakta bulunan karlı bir dağ görüntüsü yer almakta ve çok durağan bir yapıya sahip gözükmetedir.</i>
<i>Speed Bag</i>	30	<i>Kum torbası ve arkasından geçen bir sporcunun görüntüleri bulunmaktadır.</i>
<i>Stockholm</i>	60	<i>Stockholm şehrinin uzaktan görünüşü bulunmakta ve kamera soldan sağa doğru yavaşça hareket etmektedir.</i>
<i>Touchdown Pass</i>	30	<i>Bir Amerikan futbol maçındaki görüntülerden oluşmaktadır.</i>

Bu çözünürlükteki video setlerinden elde edilmiş kesitler Şekil 5.4'te verilmektedir. *Red Kayak* video seti için alınan kesitte kırmızı kano gözükmese de ilerleyen sahnelerde ortaya çıkmaktadır.



(a)



(b)



(c)



(d)



(e)



(f)

Şekil 5.4: (a) Blue Sky, (b) Red Kayak, (c) Snow Mountain, (d) Speed Bag, (e) Stockholm ve (f) Touchdown Pass videolarından kesitler.

5.2. Matlab Ortamında Yapılan Benzetimler

Önerilen yöntem kısmında bulunan, tasarlanan üç yöntem için de MATLAB ortamında testler gerçekleştirilmiştir. Bu testlerde, tasarlanan yaklaşımların etkileri gözlemlendiği gibi blokların boyutları da değiştirilerek, blok boyutunun da etkileri

irdelenmiştir. MATLAB ortamında yapılan deneysel çalışmalar, son olarak karşılaştırma alt başlığı altında değerlendirilmeye alınmıştır.

5.2.1. Tekli ve Çoklu Referans Kullanımı ile Benzetim Sonuçları

100 çerçeve içerisinde alınan sonuçlarda, çerçeve çiftlerinden elde edilmiş PSNR değerlerinin her video seti ve referans sayısı için ortalaması, maksimumu ve minimumu alınmıştır. Blok boyutlarındaki değişimin etkilerinin gözlemlenebilmesi için deneyler 4x4'lük, 8x8'lik ve 16x16'lık boyutlarda olmak üzere her bir yöntem için uygulanmıştır.

5.2.1.1. Dinamik Ağırlıklandırma İle Benzetim Sonuçları

Dinamik ağırlıklandırma ile bulunan sonuçlar kullanılan blokların boyutlarına göre üç alt başlıkta incelenmektedir. Bu alt başlıklar 4x4, 8x8 ve 16x16 boyutlu bloklarla ilgili benzetim sonuçlarını içermektedir.

4x4 Boyutlu Bloklar İle Benzetim Sonuçları

Çizelge 5.5'te, referans bloğu sayısının artması ile yeniden oluşturulan çerçevelerin kalitesinde iyi yönde gelişme olduğu görülmektedir. 4x4 blok boyutunda, referans bloğu sayısının artışı ile genellikle PSNR değerlerinde de artış gözlemlenmektedir. Bu, daha fazla referans bloğu kullanmanın hareket kestirimi ve dolayısıyla video kalitesi üzerinde olumlu bir etki olduğunu göstermektedir.

Parkrun ve *Shields* video setleri için maksimum PSNR değerleri, 3 referans bloğu kullanıldığında en yüksek değere ulaşırken, *Four People* ve *Kristen & Sara* gibi diğer videolarda PSNR değeri aynı boyutta artmamaktadır. Küçük blok boyutlarının, değişimin yüksek olduğu setlerde daha avantajlı olması beklenmekte olup sonuçlar da bunu destekler nitelikte gözükmektedir. Ortalama PSNR değerleri de referans sayısı arttıkça artış göstermektedir, bu da genel olarak daha fazla referans bloğu kullanılmasının kodlama kalitesini artırdığını göstermektedir.

Çizelge 5.5: 4x4 blok boyutu ile 1-3 arası referans ile elde edilen PSNR değerleri

Video Setleri	PSNR	1 Ref.	2 Ref.	3 Ref.
<i>Four People</i>	<i>Maksimum</i>	41,42	42,11	42,40
	<i>Minimum</i>	35,14	35,91	35,96
	<i>Ortalama</i>	40,18	40,86	41,05
<i>Kristen & Sara</i>	<i>Maksimum</i>	42,34	43,09	43,48
	<i>Minimum</i>	38,78	39,76	39,96
	<i>Ortalama</i>	41,80	42,60	42,93
<i>Mob. Cal.</i>	<i>Maksimum</i>	34,73	35,44	35,79
	<i>Minimum</i>	26,67	28,29	29,17
	<i>Ortalama</i>	30,27	31,72	32,43
<i>Parkrun</i>	<i>Maksimum</i>	27,78	29,18	29,74
	<i>Minimum</i>	23,16	24,80	25,52
	<i>Ortalama</i>	25,20	26,81	27,56
<i>Shields</i>	<i>Maksimum</i>	34,39	35,20	35,59
	<i>Minimum</i>	30,85	32,29	33,01
	<i>Ortalama</i>	32,41	33,75	34,33
<i>Stockholm</i>	<i>Maksimum</i>	33,53	34,69	35,21
	<i>Minimum</i>	31,89	33,28	33,81
	<i>Ortalama</i>	32,50	33,85	34,41

8x8 Boyutlu Bloklar İle Benzetim Sonuçları

Çizelge 5.6'da, blok boyutlarının 8x8 olması ile yeniden oluşturulan çerçevelerin kalitesinde 4x4'e göre PSNR değerlerinin video setleri genelinde arttığı görülmektedir. 8x8 blok boyutunda, PSNR değerlerinin genelde 4x4 blok boyutuna göre daha düşük olduğu görülmektedir. Video setlerin genelinde 4x4 boyutlu blok kullanımına kıyasla 1 dB'lik bir PSNR düşüşü gözlemlenmektedir. Bu durum, daha küçük blok boyutlarının hareket tahmini için daha detaylı bilgi sağlayabileceğini ve bu sayede daha yüksek PSNR değerleri elde edilebileceğini düşündürmektedir.

8x8 boyutlu blokların sonuçları incelendiğinde, referans bloğu sayısının artışıyla PSNR'de görülen artışın 8x8 blok boyutunda da devam ettiği görülmektedir.

Çizelge 5.6: 8x8 blok boyutu ile 1-3 arası referans ile elde edilen PSNR değerleri

Video Setleri	PSNR	1 Ref.	2 Ref.	3 Ref.
	<i>Maksimum</i>	40,38	41,04	41,45
<i>Four People</i>	<i>Minimum</i>	36,57	36,99	37,18
	<i>Ortalama</i>	39,51	40,24	40,66
	<i>Maksimum</i>	41,20	41,96	42,45
<i>Kristen & Sara</i>	<i>Minimum</i>	38,93	39,67	40,02
	<i>Ortalama</i>	40,59	41,45	41,95
	<i>Maksimum</i>	34,18	34,67	34,91
<i>Mob. Cal.</i>	<i>Minimum</i>	25,79	27,83	28,91
	<i>Ortalama</i>	29,59	31,09	31,78
	<i>Maksimum</i>	26,95	28,21	28,65
<i>Parkrun</i>	<i>Minimum</i>	22,04	24,05	24,85
	<i>Ortalama</i>	24,31	25,87	26,55
	<i>Maksimum</i>	33,45	34,29	34,72
<i>Shields</i>	<i>Minimum</i>	29,81	31,46	32,46
	<i>Ortalama</i>	31,60	33,15	33,75
	<i>Maksimum</i>	32,60	33,62	34,10
<i>Stockholm</i>	<i>Minimum</i>	30,87	32,23	32,84
	<i>Ortalama</i>	31,48	32,89	33,40

16x16 Boyutlu Bloklar İle Benzetim Sonuçları

Çizelge 5.7’de, blok boyutlarının 16x16 olması ile yeniden oluşturulan çerçevelerin kalitesinde 8x8’e göre PSNR değerlerinin video setleri genelinde düştüğü görülmektedir. Bu, daha büyük blok boyutlarının detay kaybına yol açabileceğini ve bu nedenle hareket tahmininde daha az etkili olabileceğini göstermektedir.

Mob. Cal. ve *Parkrun* gibi bazı videolarda, referans bloğu sayısının artması ile PSNR değerlerinde çok fazla bir artış gözlemlenmemektedir. Bu da büyük blok boyutlarının, bu özel içerikler için hareket tahmini açısından yeterince detay sağlamadığını düşündürmektedir.

Çizelge 5.7: 16x16 blok boyutu ile 1-3 arası referans ile elde edilen PSNR değerleri

Video Setleri	PSNR	1 Ref.	2 Ref.	3 Ref.
<i>Four People</i>	<i>Maksimum</i>	39,37	40,06	40,48
	<i>Minimum</i>	35,76	36,23	36,49
	<i>Ortalama</i>	38,59	39,25	39,65
<i>Kristen & Sara</i>	<i>Maksimum</i>	40,59	41,32	41,78
	<i>Minimum</i>	38,20	38,77	39,14
	<i>Ortalama</i>	39,80	40,64	41,16
<i>Mob. Cal.</i>	<i>Maksimum</i>	33,62	33,94	34,13
	<i>Minimum</i>	25,36	27,47	28,12
	<i>Ortalama</i>	29,11	30,57	31,19
<i>Parkrun</i>	<i>Maksimum</i>	26,49	27,58	27,91
	<i>Minimum</i>	21,51	23,52	24,37
	<i>Ortalama</i>	23,88	25,37	26,00
<i>Shields</i>	<i>Maksimum</i>	32,62	33,33	33,77
	<i>Minimum</i>	29,05	30,67	31,78
	<i>Ortalama</i>	30,96	32,46	32,99
<i>Stockholm</i>	<i>Maksimum</i>	32,00	32,88	33,26
	<i>Minimum</i>	30,23	31,64	32,13
	<i>Ortalama</i>	30,79	32,16	32,59

5.2.1.2. Statik Ağırlıklandırma İle Benzetim Sonuçları

Birinci ağırlık (en benzer bloğun ağırlığı), video boyunca tek bir sabit değere ayarlanmaktadır. Birinci ağırlık w_1 , 0,4'ten 1,0'a 0,1 artışlarla yedi değere sabitlenmektedir. İkinci ve üçüncü en benzer blokların ağırlıkları olan w_2 ve w_3 ise w_1 değerleri sabit tutulurken en küçük kareler yöntemi kullanılarak hesaplanmaktadır.

4x4 Boyutlu Bloklar İle Benzetim Sonuçları

Çizelge 5.8'de, bu kısımda elde edilen PSNR değerleri gösterilmektedir. *Four People* video seti için w_1 0,8 değerinde iken 0,65dB ile diğer sabit ağırlıklara göre en iyi PSNR değerinin alındığı görülmektedir. Bu değer, 4x4 blok boyutlu iki referans kullanılan dinamik ağırlıklandırma yöntemine yakın bir kalite artışı vermektedir. En kötü sonuçlardan birisi olan -21,83dB değerini ise 0,4 ağırlığı seçildiğinde vermiştir. *Kristen & Sara* video seti için de

en iyi artış olan 0,69dB'lik değer w_1 0,8 değerine sahipken elde edilmiştir. En kötü PSNR çıktısı olan -27,62dB'lik değeri de birinci ağırlık 0,4'e sabitlenmişken gelmiştir. *Mob. Cal.* video seti için w_1 0,7 değerinde iken 1,58dB ile diğer sabit ağırlıklara göre en iyi PSNR değerinin alındığı görülmektedir. Bu değer, 4x4 blok boyutlu iki referans kullanılan dinamik ağırlıklandırma yönteminden daha yüksek bir kalite artışı vermektedir. *Parkrun* video seti için de en iyi artış olan 2,01dB'lik değer w_1 0,6 değerine sahipken elde edilmiştir. Bu değer, 4x4 blok boyutlu iki referans kullanılan dinamik ağırlıklandırma yönteminden çok daha yüksek bir PSNR artışı vermektedir. *Shields* seti için w_1 0,7 değerinde iken 1,49dB ile diğer sabit ağırlıklara göre en iyi PSNR değerinin alındığı gözlemlenmektedir.

Çizelge 5.8: 4x4 blok boyutunda, en benzer bloğun ağırlığının farklı değerlerde sabitlenmesi ile gelen PSNR değerleri

Video Setleri	PSNR	Birinci Referansın Sabit Tutulduğu Ağırlıklar						
		0.4	0.5	0.6	0.7	0.8	0.9	1.0
<i>Four People</i>	<i>Maksimum</i>	18,58	41,14	41,59	41,90	42,02	41,92	41,65
	<i>Minimum</i>	18,14	35,81	35,95	36,00	36,03	36,01	35,93
	<i>Ortalama</i>	18,35	40,19	40,54	40,76	40,82	40,72	40,46
<i>Kristen & Sara</i>	<i>Maksimum</i>	14,28	41,93	42,44	42,80	42,96	42,91	42,65
	<i>Minimum</i>	14,11	39,02	39,27	39,44	39,50	39,46	39,32
	<i>Ortalama</i>	14,18	41,56	42,03	42,35	42,49	42,42	42,15
<i>Mob. Cal.</i>	<i>Maksimum</i>	27,90	32,39	33,41	34,34	35,01	35,25	34,97
	<i>Minimum</i>	25,88	28,95	28,88	28,66	28,30	27,84	27,30
	<i>Ortalama</i>	27,14	31,38	31,72	31,85	31,74	31,38	30,82
<i>Parkrun</i>	<i>Maksimum</i>	25,67	29,03	29,28	29,29	29,11	28,70	28,15
	<i>Minimum</i>	23,32	25,32	25,28	25,08	24,73	24,27	23,71
	<i>Ortalama</i>	24,63	27,11	27,21	27,09	26,77	26,28	25,68
<i>Shields</i>	<i>Maksimum</i>	27,95	34,41	34,85	35,15	35,18	34,99	34,67
	<i>Minimum</i>	27,10	32,50	32,76	32,68	32,35	31,88	31,30
	<i>Ortalama</i>	27,54	33,58	33,84	33,90	33,74	33,38	32,87
<i>Stockholm</i>	<i>Maksimum</i>	33,08	34,76	34,87	34,83	34,64	34,30	33,85
	<i>Minimum</i>	31,95	33,58	33,60	33,50	33,27	32,93	32,48
	<i>Ortalama</i>	32,47	34,13	34,18	34,06	33,80	33,42	32,94

İki referans kullanılan dinamik ağırlıklandırma yöntemine yakın bir kalite artışı gözlemlenmiştir. *Stockholm* video seti için en iyi artış olan 1,68dB'lik değer w_1 0,6 değerine sahipken elde edilmiştir. İki referans kullanılan dinamik ağırlıklandırma yönteminden çok daha yüksek bir PSNR artışı vermektedir. Sonuçlardan görüldüğü üzere, genel olarak, iki

referans bloğu kullanıldığı benzetimlere yakın sonuçlar vermesine rağmen, üç referans blok kullanılması durumunda elde edilen değerlerden düşük PSNR verileri elde edilmiştir.

8x8 Boyutlu Bloklar İle Benzetim Sonuçları

Çizelge 5.9’da, bu kısımda elde edilen PSNR değerleri gösterilmektedir. *Four People* video seti için w_1 0,8 değerinde iken 0,69dB ile diğer sabit ağırlıklara göre en iyi PSNR değerinin alındığı görülmektedir. Bu değer, 8x8 blok boyutlu iki referans kullanılan dinamik ağırlıklandırma yöntemine yakın bir kalite artışı vermektedir. *Kristen & Sara* video seti için de en iyi artış olan 0,77dB’lik değer w_1 0,8 değerine sahipken elde edilmiştir.

Çizelge 5.9: 8x8 blok boyutunda, en benzer bloğun ağırlığının farklı değerlerde sabitlenmesi ile gelen PSNR değerleri

Video Setleri	PSNR	Birinci Referansın Sabit Tutulduğu Ağırlıklar						
		0.4	0.5	0.6	0.7	0.8	0.9	1.0
<i>Four People</i>	<i>Maksimum</i>	33,61	40,17	40,58	40,86	40,96	40,89	40,63
	<i>Minimum</i>	31,56	36,72	36,89	36,97	36,98	36,90	36,75
	<i>Ortalama</i>	32,54	39,62	39,96	40,16	40,20	40,07	39,80
<i>Kristen & Sara</i>	<i>Maksimum</i>	17,42	40,95	41,41	41,71	41,85	41,77	41,50
	<i>Minimum</i>	17,00	39,09	39,38	39,56	39,61	39,52	39,32
	<i>Ortalama</i>	17,23	40,61	41,02	41,28	41,36	41,25	40,95
<i>Mob. Cal.</i>	<i>Maksimum</i>	31,29	31,76	32,61	33,55	34,24	34,53	34,32
	<i>Minimum</i>	28,66	28,74	28,58	28,24	27,75	27,18	26,54
	<i>Ortalama</i>	30,23	30,77	31,11	31,22	31,08	30,69	30,11
<i>Parkrun</i>	<i>Maksimum</i>	26,00	28,15	28,33	28,32	28,13	27,78	27,30
	<i>Minimum</i>	23,37	24,71	24,60	24,33	23,88	23,27	22,60
	<i>Ortalama</i>	24,59	26,22	26,28	26,13	25,81	25,34	24,77
<i>Shields</i>	<i>Maksimum</i>	33,47	33,88	34,13	34,29	34,23	34,02	33,66
	<i>Minimum</i>	31,34	31,82	32,11	31,85	31,43	30,87	30,23
	<i>Ortalama</i>	32,61	33,22	33,38	33,32	33,05	32,61	32,04
<i>Stockholm</i>	<i>Maksimum</i>	33,52	33,73	33,81	33,75	33,55	33,24	32,83
	<i>Minimum</i>	32,53	32,65	32,62	32,48	32,22	31,88	31,42
	<i>Ortalama</i>	33,02	33,18	33,18	33,04	32,77	32,39	31,93

Kötü sonuçlardan birisi olan -23,36dB’lik değeri de birinci ağırlık 0,4’e sabitlenmişken gelmiştir. *Mob. Cal.* video seti için w_1 0,7 değerinde iken 1,63 dB ile diğer sabit ağırlıklara göre en iyi PSNR değerinin alındığı görülmektedir. Bu değer, 4x4 blok

boyutlu iki referans kullanılan dinamik ağırlıklandırma yöntemine yakın bir kalite artışı vermektedir. *Parkrun* video seti için de en iyi artış olan 1,97 dB'lik değer w_1 0,6 değerine sahipken elde edilmiştir. Bu değer, 8x8 blok boyutlu iki referans kullanılan dinamik ağırlıklandırma yönteminden çok daha yüksek bir PSNR artışı vermektedir. *Shields* seti için w_1 0,7 değerinde iken 1,78dB ile diğer sabit ağırlıklara göre en iyi PSNR değerinin alındığı gözlemlenmektedir. İki referans kullanılan dinamik ağırlıklandırma yöntemine yakın bir kalite artışı gözlemlenmiştir. *Stockholm* video seti için en iyi artış olan 1,7 dB'lik değer w_1 0,5 ve 0,6 değerlerine sahipken elde edilmiştir. İki referans kullanılan dinamik ağırlıklandırma yönteminden daha yüksek bir PSNR artışı vermektedir. Sonuçlardan görüldüğü üzere, genel olarak, iki referans bloğunun kullanıldığı benzetimlere yakın sonuçlar vermesine rağmen, üç referans blok kullanılması durumunda elde edilen değerlerden düşük PSNR verileri elde edilmiştir.

16x16 Boyutlu Bloklar İle Benzetim Sonuçları

Çizelge 5.10'da, bu kısımda elde edilen PSNR değerleri gösterilmektedir. *Four People* video seti için w_1 0,8 değerinde iken 0,63dB ile diğer sabit ağırlıklara göre en iyi PSNR değerinin alındığı görülmektedir. Bu değer, 16x16 blok boyutlu iki referans kullanılan dinamik ağırlıklandırma yöntemine yakın bir kalite artışı vermektedir. *Kristen & Sara* video seti için de en iyi artış olan 0,75dB'lik değer w_1 0,8 değerine sahipken elde edilmiştir. Kötü sonuçlardan birisi olan -16,08 dB'lik değeri de birinci ağırlık 0,4'e sabitlenmişken gelmiştir. *Mob. Cal.* video seti için w_1 0,7 değerinde iken 1,57dB ile diğer sabit ağırlıklara göre en iyi PSNR değerinin alındığı görülmektedir. Bu değer, 16x16 blok boyutlu iki referans kullanılan dinamik ağırlıklandırma yöntemine yakın bir kalite artışı vermektedir. *Parkrun* video seti için de en iyi artış olan 1,89dB'lik değer w_1 0,6 değerine sahipken elde edilmiştir. Bu değer, 16x16 blok boyutlu iki referans kullanılan dinamik ağırlıklandırma yönteminden daha yüksek bir PSNR artışı vermektedir. *Shields* seti için w_1 0,7 değerinde iken 1,71dB ile diğer sabit ağırlıklara göre en iyi PSNR değerinin alındığı gözlemlenmektedir. İki referans kullanılan dinamik ağırlıklandırma yöntemine yakın bir kalite artışı gözlemlenmiştir. *Stockholm* video seti için en iyi artış olan 1,61dB'lik değer w_1 0,5 değerine sahipken elde edilmiştir. İki referans kullanılan dinamik ağırlıklandırma yönteminden daha yüksek bir PSNR artışı vermektedir.

Çizelge 5.10: 16x16 blok boyutunda, en benzer bloğun ağırlığının farklı değerlerde sabitlenmesi ile gelen PSNR değerleri

Video Setleri	PSNR	Birinci Referansın Sabit Tutulduğu Ağırlıklar						
		0.4	0.5	0.6	0.7	0.8	0.9	1.0
<i>Four People</i>	<i>Maksimum</i>	37,97	39,39	39,76	39,98	40,04	39,91	39,63
	<i>Minimum</i>	35,31	36,03	36,18	36,25	36,26	36,18	36,03
	<i>Ortalama</i>	37,40	38,68	38,99	39,18	39,22	39,12	38,88
<i>Kristen & Sara</i>	<i>Maksimum</i>	25,19	40,19	40,68	41,03	41,19	41,15	40,91
	<i>Minimum</i>	22,86	38,24	38,51	38,68	38,73	38,66	38,46
	<i>Ortalama</i>	23,72	39,78	40,19	40,46	40,55	40,44	40,16
<i>Mob. Cal.</i>	<i>Maksimum</i>	31,08	31,43	32,03	32,92	33,58	33,87	33,72
	<i>Minimum</i>	27,85	27,96	27,90	27,68	27,30	26,74	26,07
	<i>Ortalama</i>	29,76	30,26	30,58	30,68	30,53	30,15	29,58
<i>Parkrun</i>	<i>Maksimum</i>	26,58	27,51	27,67	27,70	27,57	27,27	26,85
	<i>Minimum</i>	23,58	24,26	24,12	23,82	23,32	22,71	22,05
	<i>Ortalama</i>	24,84	25,74	25,77	25,62	25,31	24,86	24,33
<i>Shields</i>	<i>Maksimum</i>	32,89	33,22	33,38	33,44	33,32	33,11	32,80
	<i>Minimum</i>	30,65	31,10	31,41	31,11	30,65	30,07	29,42
	<i>Ortalama</i>	32,20	32,54	32,67	32,61	32,34	31,92	31,38
<i>Stockholm</i>	<i>Maksimum</i>	32,87	32,98	33,02	32,97	32,81	32,54	32,18
	<i>Minimum</i>	31,80	31,94	31,96	31,82	31,56	31,20	30,77
	<i>Ortalama</i>	32,27	32,40	32,39	32,26	32,01	31,67	31,25

Sonuçlardan görüldüğü üzere, genel olarak, üç referans blok kullanılması durumunda elde edilen değerlerden düşük PSNR verileri elde edilmesine rağmen, iki referans bloğunun kullanıldığı benzetimlere yakın ve bazı setlerde iyi sonuçlar vermiştir.

5.2.1.3. İndekslenmiş Ağırlıklandırma İle Benzetim Sonuçları

Test çerçeveleri boyunca elde edilen ağırlık üçlüleri, K-means algoritması kullanılarak öbekleme yapılmıştır. Tüm ağırlık üçlülerinin 8 veya 32 farklı üçlülerden biri olması sağlanmaktadır. Böylece kod çözücü kısmına gönderilecek hareket vektörü bilgisi 3 veya 5 bit ile ifade edilebilecektir. PSNR değerlerinin, dinamik ağırlıklandırma yöntemine göre azalacağı aşikâr olmasına rağmen kodlanacak hareket vektörü bilgisinin düşecek olması sıkıştırma oranını arttıracaktır.

4x4 Boyutlu Bloklar İle Benzetim Sonuçları

Çizelge 5.11’de, bu kısımda elde edilen PSNR değerleri gösterilmektedir. Ondalıklı gösterim başlığı dinamik ağırlıklandırma yönteminin sonuçlarından oluşmaktadır. 4x4 blok boyutunda, 3 bit ve 5 bit ile öbeleme kullanıldığında PSNR değerleri, ondalıklı gösterimle elde edilen PSNR değerlerine oldukça yakın gözükmektedir. Bu durum, ağırlıkların öbeleme yönteminin, orijinal dinamik ağırlıklandırma yöntemine kıyasla nispeten düşük bir kalite kaybıyla veri iletimi verimliliğini artırabileceğini göstermektedir.

Four People ve *Kristen & Sara* setlerinde, 5 bit ağırlıklar, 3 bit ağırlıklarına göre daha yüksek PSNR değerleri sağlamakta ve bu da daha fazla bit kullanımının doğal olarak kalite kaybının daha az olacağını göstermektedir.

Çizelge 5.11: 4x4 blok boyutunda, kümeleme yöntemi ile referans blokların ağırlık bilgisinin 3 ve 5 bit ile temsil edildiği PSNR sonuçlarının, ağırlık bilgisinin ondalıklı olarak ifade edildiği PSNR sonuçlarıyla kıyaslaması

Video Setleri	PSNR	3 Bit			5 Bit		Ondalıklı	
		1 Ref.	2 Ref.	3 Ref.	2 Ref.	3 Ref.	2 Ref.	3 Ref.
<i>Four People</i>	<i>Maksimum</i>	41,42	42,10	42,13	42,11	42,27	42,11	42,40
	<i>Minimum</i>	35,14	35,22	35,19	35,79	35,12	35,91	35,96
	<i>Ortalama</i>	40,18	40,81	40,80	40,83	40,89	40,86	41,05
<i>Kristen & Sara</i>	<i>Maksimum</i>	42,34	43,07	43,24	43,09	43,35	43,09	43,48
	<i>Minimum</i>	38,78	39,19	39,47	39,41	39,50	39,76	39,96
	<i>Ortalama</i>	41,80	42,56	42,66	42,59	42,77	42,60	42,93
<i>Mob. Cal.</i>	<i>Maksimum</i>	34,73	35,42	35,54	35,45	35,69	35,44	35,79
	<i>Minimum</i>	26,67	28,26	28,85	28,29	29,08	28,29	29,17
	<i>Ortalama</i>	30,27	31,68	32,10	31,71	32,30	31,72	32,43
<i>Parkrun</i>	<i>Maksimum</i>	27,78	29,16	29,54	29,16	29,64	29,18	29,74
	<i>Minimum</i>	23,16	24,79	25,37	24,81	25,47	24,80	25,52
	<i>Ortalama</i>	25,20	26,80	27,37	26,81	27,49	26,81	27,56
<i>Shields</i>	<i>Maksimum</i>	34,39	35,18	35,39	35,19	35,51	35,20	35,59
	<i>Minimum</i>	30,85	32,28	32,85	32,30	32,94	32,29	33,01
	<i>Ortalama</i>	32,41	33,74	34,12	33,75	34,26	33,75	34,33
<i>Stockholm</i>	<i>Maksimum</i>	33,53	34,65	35,02	34,67	35,12	34,69	35,21
	<i>Minimum</i>	31,89	33,28	33,68	33,28	33,76	33,28	33,81
	<i>Ortalama</i>	32,50	33,81	34,24	33,82	34,32	33,85	34,41

8x8 Boyutlu Bloklar İle Benzetim Sonuçları

Çizelge 5.12’de, bu kısımda elde edilen PSNR değerleri gösterilmektedir. 8x8 blok boyutunda da 4x4 boyutuna benzer bir eğilim gözlemlenmektedir. Ancak, 8x8 boyutunda PSNR değerleri genel olarak 4x4 boyutuna göre daha düşük, bu da daha büyük blok boyutlarının bazı detayları kaçırabileceğine işaret etmektedir.

5 bit ile ağırlıklandırma kullanımı, 3 bit kullanımına göre genellikle daha iyi sonuçlar vermekte ve bu da yine daha fazla bit kullanımının daha iyi kalite korumasına katkı sağladığını göstermektedir.

Çizelge 5.12: 8x8 blok boyutunda, kümeleme yöntemi ile referans blokların ağırlık bilgisinin 3 ve 5 bit ile temsil edildiği PSNR sonuçlarının, ağırlık bilgisinin ondalıklı olarak ifade edildiği PSNR sonuçlarıyla kıyaslaması

Video Setleri	PSNR	3 Bit			5 Bit			Ondalıklı	
		1 Ref.	2 Ref.	3 Ref.	2 Ref.	3 Ref.	2 Ref.	3 Ref.	
<i>Four People</i>	<i>Maksimum</i>	40,38	41,01	41,32	41,04	41,39	41,04	41,45	
	<i>Minimum</i>	36,57	36,99	37,14	36,99	37,15	36,99	37,18	
	<i>Ortalama</i>	39,51	40,22	40,52	40,24	40,59	40,24	40,66	
<i>Kristen & Sara</i>	<i>Maksimum</i>	41,20	41,94	42,31	41,95	42,38	41,96	42,45	
	<i>Minimum</i>	38,93	39,64	39,78	39,66	39,96	39,67	40,02	
	<i>Ortalama</i>	40,59	41,42	41,79	41,45	41,88	41,45	41,95	
<i>Mob. Cal.</i>	<i>Maksimum</i>	34,18	34,65	34,79	34,66	34,86	34,67	34,91	
	<i>Minimum</i>	25,79	27,82	28,75	27,83	28,86	27,83	28,91	
	<i>Ortalama</i>	29,59	31,07	31,59	31,09	31,71	31,09	31,78	
<i>Parkrun</i>	<i>Maksimum</i>	26,95	28,20	28,49	28,21	28,61	28,21	28,65	
	<i>Minimum</i>	22,04	24,05	24,76	24,05	24,81	24,05	24,85	
	<i>Ortalama</i>	24,31	25,87	26,43	25,87	26,51	25,87	26,55	
<i>Shields</i>	<i>Maksimum</i>	33,45	34,28	34,59	34,29	34,67	34,29	34,72	
	<i>Minimum</i>	29,81	31,44	32,33	31,45	32,42	31,46	32,46	
	<i>Ortalama</i>	31,60	33,14	33,61	33,15	33,70	33,15	33,75	
<i>Stockholm</i>	<i>Maksimum</i>	32,60	33,62	33,99	33,62	34,07	33,62	34,10	
	<i>Minimum</i>	30,87	32,20	32,72	32,21	32,78	32,23	32,84	
	<i>Ortalama</i>	31,48	32,87	33,28	32,88	33,35	32,89	33,40	

16x16 Boyutlu Bloklar İle Benzetim Sonuçları

Çizelge 5.13'te, bu kısımda elde edilen PSNR değerleri gösterilmektedir. 16x16 blok boyutunda, PSNR değerleri diğer iki blok boyutuna kıyasla genel olarak daha düşük gözükmektedir. Bu durum, büyük blok boyutlarının hareket ve detay tahminindeki etkinliğinin daha az olduğunu göstermektedir.

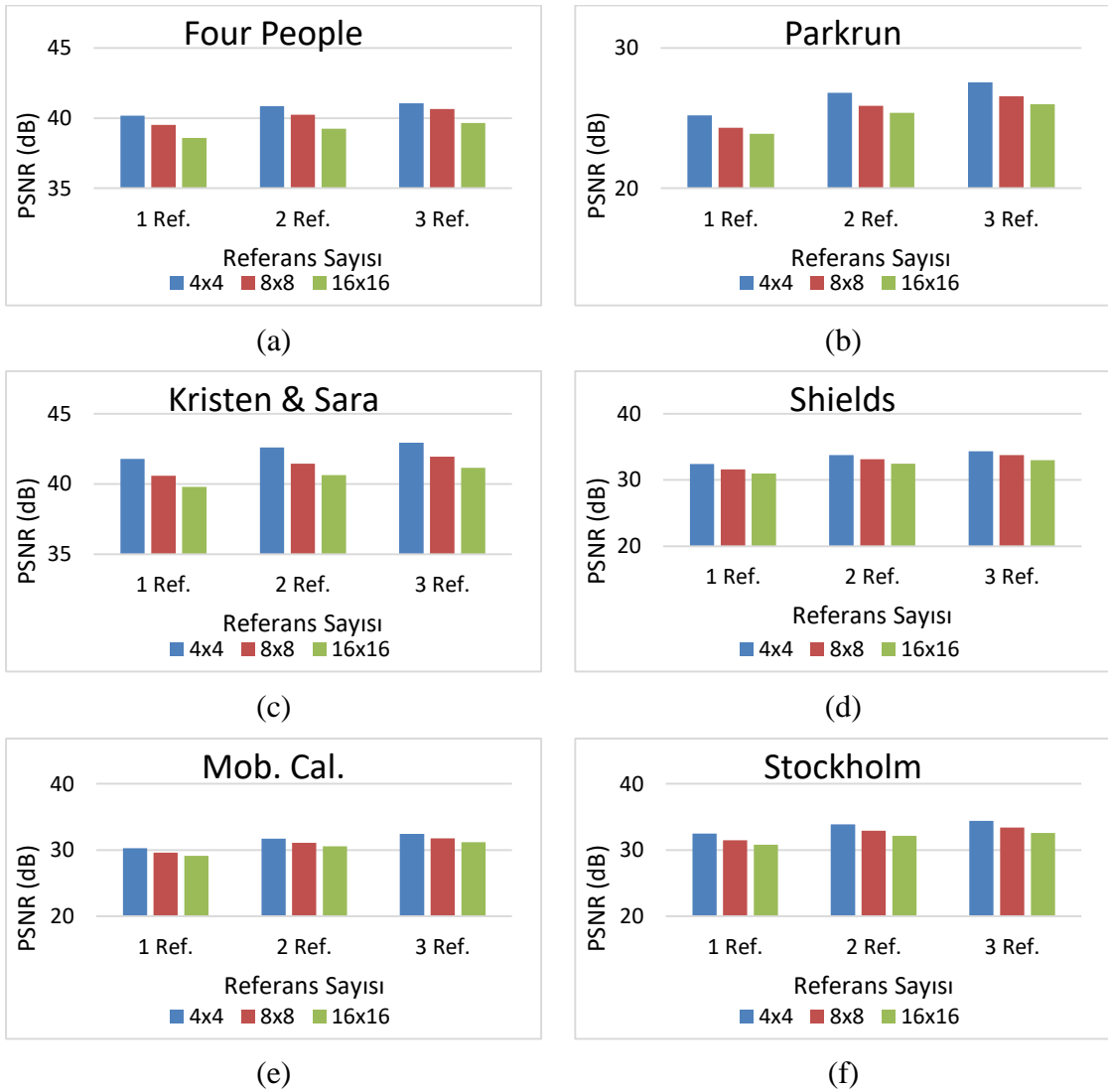
16x16 boyutunda da, özellikle *Kristen & Sara* ve *Mob. Cal.* setlerinde, 5 bit ağırlıklandırma, 3 bit ve ondalıklı ağırlıklandırmaya göre daha iyi PSNR değerleri sağlamaktadır.

Çizelge 5.13: 16x16 blok boyutunda, kümeleme yöntemi ile referans blokların ağırlık bilgisinin 3 ve 5 bit ile temsil edildiği PSNR sonuçlarının, ağırlık bilgisinin ondalıklı olarak ifade edildiği PSNR sonuçlarıyla kıyaslaması

Video Setleri	PSNR	3 Bit			5 Bit			Ondalıklı	
		1 Ref.	2 Ref.	3 Ref.	2 Ref.	3 Ref.	2 Ref.	3 Ref.	
<i>Four People</i>	<i>Maksimum</i>	39,37	40,04	40,35	40,05	40,43	40,06	40,48	
	<i>Minimum</i>	35,76	36,23	36,42	36,23	36,46	36,23	36,49	
	<i>Ortalama</i>	38,59	39,23	39,53	39,25	39,60	39,25	39,65	
<i>Kristen & Sara</i>	<i>Maksimum</i>	40,59	41,30	41,60	41,32	41,70	41,32	41,78	
	<i>Minimum</i>	38,20	38,76	39,02	38,77	39,10	38,77	39,14	
	<i>Ortalama</i>	39,80	40,62	41,00	40,64	41,09	40,64	41,16	
<i>Mob. Cal.</i>	<i>Maksimum</i>	33,62	33,93	34,05	33,94	34,10	33,94	34,13	
	<i>Minimum</i>	25,36	27,45	27,95	27,47	28,08	27,47	28,12	
	<i>Ortalama</i>	29,11	30,55	31,07	30,57	31,16	30,57	31,19	
<i>Parkrun</i>	<i>Maksimum</i>	26,49	27,57	27,79	27,58	27,88	27,58	27,91	
	<i>Minimum</i>	21,51	23,52	24,30	23,52	24,35	23,52	24,37	
	<i>Ortalama</i>	23,88	25,37	25,91	25,37	25,97	25,37	26,00	
<i>Shields</i>	<i>Maksimum</i>	32,62	33,32	33,66	33,33	33,74	33,33	33,77	
	<i>Minimum</i>	29,05	30,67	31,67	30,67	31,75	30,67	31,78	
	<i>Ortalama</i>	30,96	32,45	32,86	32,46	32,95	32,46	32,99	
<i>Stockholm</i>	<i>Maksimum</i>	32,00	32,86	33,19	32,88	33,24	32,88	33,26	
	<i>Minimum</i>	30,23	31,59	32,05	31,60	32,11	31,64	32,13	
	<i>Ortalama</i>	30,79	32,16	32,50	32,16	32,56	32,16	32,59	

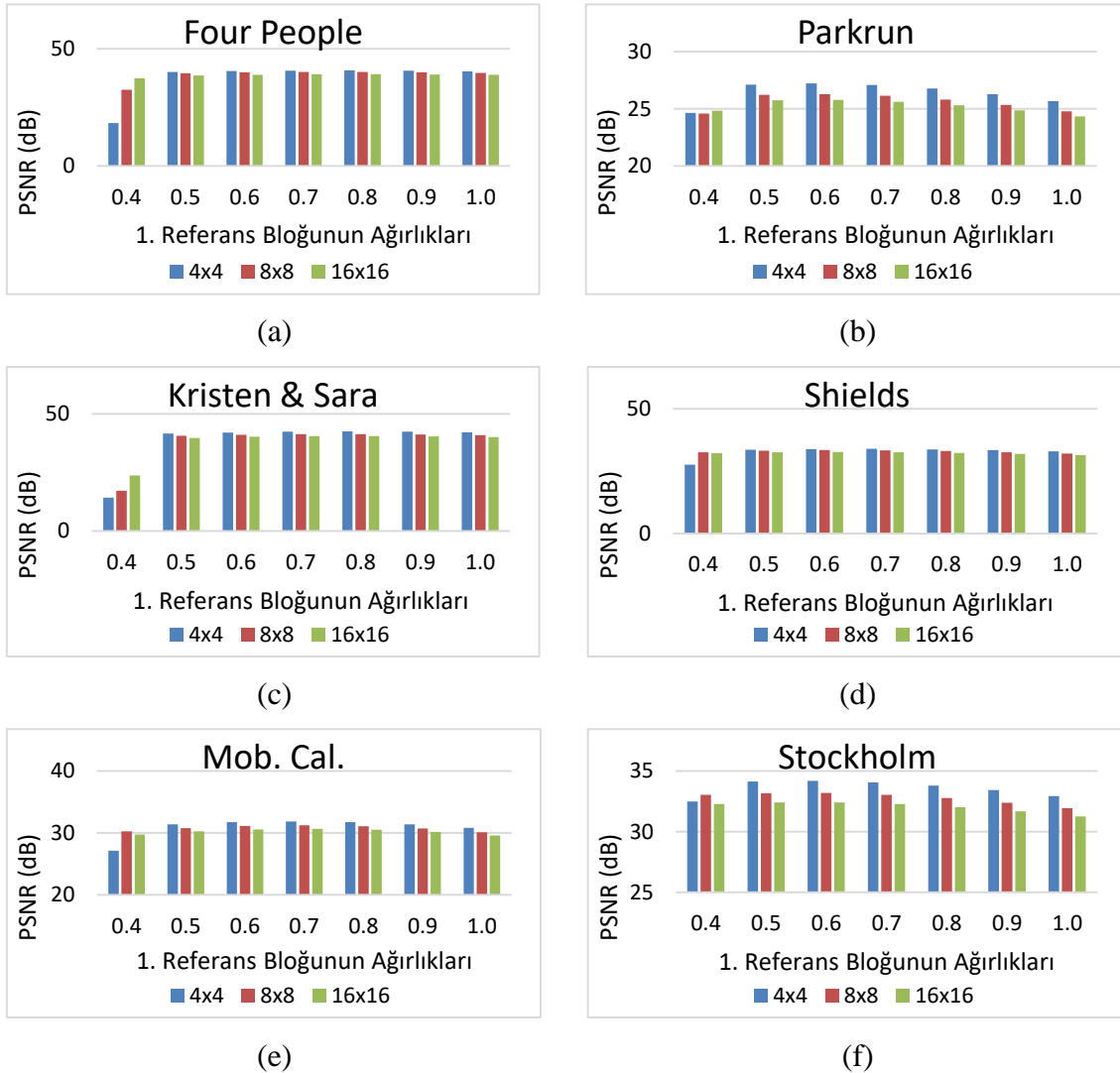
5.2.2. Karşılaştırma

Bulgular blok boyutları açısından incelenecek olursa, dinamik ağırlıklandırmada 4x4 blok boyutu ile yapılan benzetim sonuçlarında PSNR değerinde en iyi artış gözlemlenmektedir. 8x8'lik ve 16x16'lık bloklar, ortalama PSNR değeri açısından azaldığını göstermektedir. Referans bloğu sayısındaki artış ile PSNR değerlerinde de artış görülmektedir. Şekil 5.5 incelendiğinde, en yüksek PSNR değerlerine *Four People* ve *Kristen & Sara* video setlerinde ulaşıldığı görülmektedir. Geri kalan setlerle farkları, sabit ve daha az karmaşık bir arka plana sahip olmasıdır. Küçük blok boyutlarının, hesaplama süresini arttırması beklense de kalitede artış sağladığı sonucunu değiştirmemektedir.



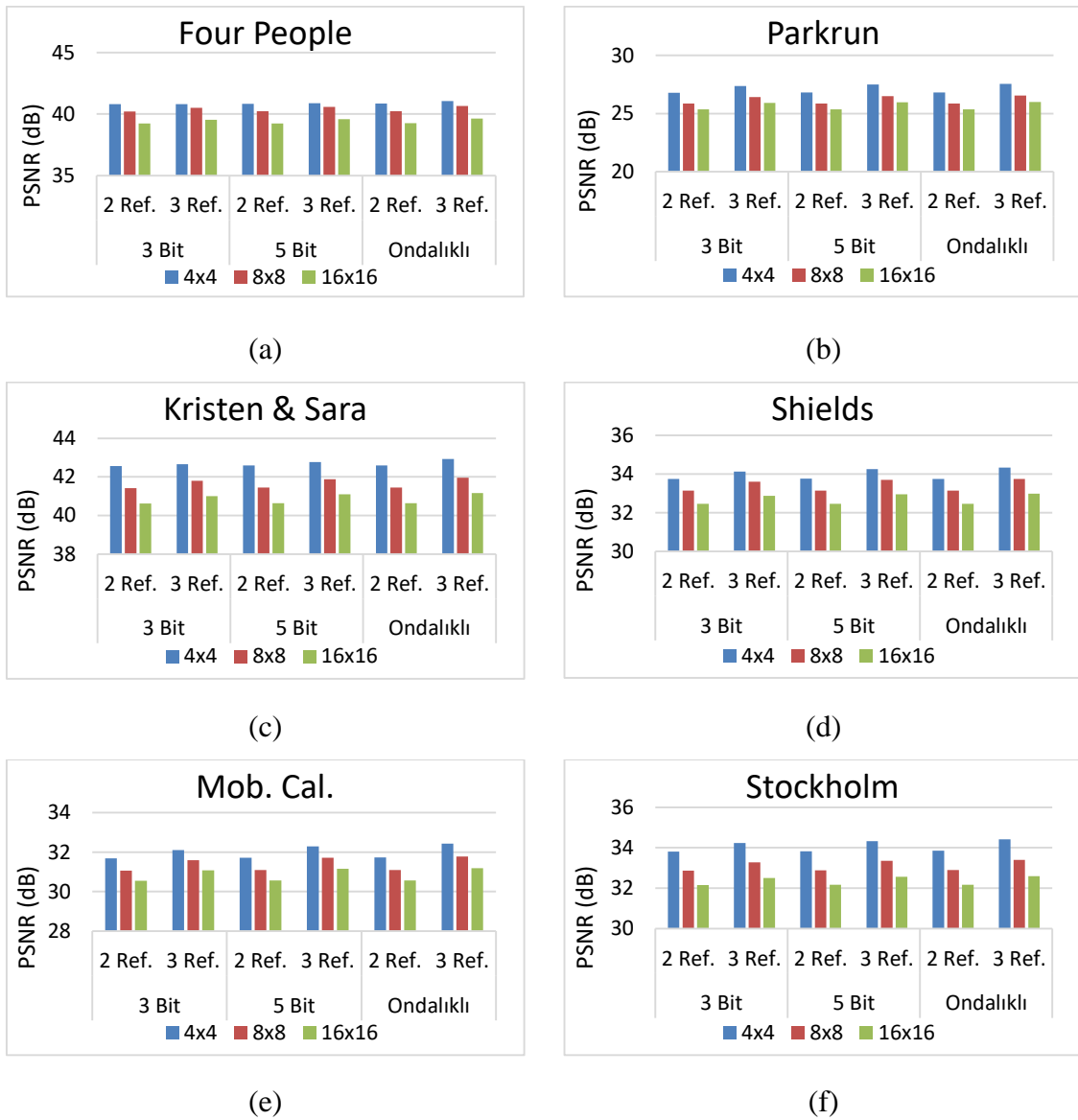
Şekil 5.5: Tüm setler için dinamik ağırlıklandırma ile elde edilen sonuçlar, (a) Four People, (b) Parkrun, (c) Kristen & Sara, (d) Shields, (e) Mob. Cal., (f) Stockholm

Statik ağırlıklandırma ise PSNR kazanımlarının birinci ağırlığın sabit tutulduğu değere ve veri setine göre değişiklik gösterdiği gözlemlenmektedir. Şekil 5.6 incelendiğinde çoğu video setinde birinci ağırlık 0.4'e sabitlendiğinde en düşük PSNR değerlerini elde ettikleri görülmektedir. Mob. Cal., Parkrun ve Stockholm setlerinde hangi ağırlıkta en iyi PSNR sonuçları alındığı daha belirgin bir şekilde gözükmemektedir. Bu kısımdaki bulgularda blok boyutunun küçülmesinin, bazı video setlerinde PSNR artışı sağlamadığı elde edilmektedir. Her blok boyutu için birinci referansın ağırlığı belirli bir noktaya kadar arttıkça PSNR değerlerinde genel olarak bir artış gözükmemektedir. Fakat o belirli noktalar geçildiğinde ağırlık arttıkça PSNR değerleri azalmaktadır.



Şekil 5.6: Statik ağırlıklandırma için elde edilen sonuçlar, (a) Four People, (b) Parkrun, (c) Kristen & Sara, (d) Shields, (e) Mob. Cal., (f) Stockholm

İndekslenirilmiş ağırlıklandırma, Şekil 5.7 incelendiğinde 2 referans kullanımı tüm ağırlık gösterim türleri için birbirlerine çok yakın sonuçlar vermektedir. İki referans kullanılmak istenildiğinde 3 bitlik ağırlık setleri çok daha avantajlı olacaktır. Üç referans kullanımında ise kalite açısından ondalıklı gösterim, veri sıkıştırması ve kalite gözetimi bir arada ise 5 bitlik ağırlık setleri kullanımı uygun olduğu görülmektedir. Gönderilecek ekstra bit sayısı azaltılarak veri iletimi verimliliği artırılmak istendiğinde, ağırlıkların 5 bit ile ifade edilmesi daha dengeli bir kodlama sunmakta ve kabul edilebilir kalite seviyeleri sağlamaktadır.



Şekil 5.7: İndekslenirilmiş ağırlıklandırma için elde edilen sonuçlar, (a) Four People, (b) Parkrun, (c) Kristen & Sara, (d) Shields, (e) Mob. Cal., (f) Stockholm

5.3. JM 19.1 Ortamında Yapılan Benzetimler

Önerilen yöntem kısmında bulunan, dinamik ağırlıklandırma yöntemi her bir video seti için 4 farklı nicemleme parametresi çalıştırılmıştır. Nicemleme parametresi JVET ortak test koşullarında belirtilen {22, 27, 32, 37} değerleri olarak seçilmiştir. Bu değerler, farklı bit hızlarında ayrı PSNR değerleri elde etmemizi sağlamıştır. Bit hızı-PSNR ilişkileri de bu kısımda grafikler aracılığı ile sunulmaktadır.

5.3.1. 1080p Video Setleri ile Yapılan Benzetimler

Aerial, *Boat*, *Crowd Run* ve *Dinner Scene* setleri için Şekil 5.8'de Y, U ve V bileşenleri için bit hızı PSNR arasındaki ilişki verilmektedir. *Aerial*, *Boat* ve *Crowd Run* setleri için Y bileşeni grafikleri incelendiğinde, bu setlerde, bit hızı arttıkça PSNR değerlerinin genel olarak arttığı görülmektedir. Bu durum, daha fazla bit hızının, özellikle Y bileşeninde, daha iyi görüntü kalitesine izin verdiğini göstermektedir. Önerilen yöntem ve JM 19.1 yöntemi arasında PSNR değerleri benzerlik göstermekte, ancak önerilen yöntem bazı bit hızlarında daha yüksek PSNR değerleri sunduğu görülmektedir. U bileşeni grafikleri incelendiğinde, JM 19.1 yöntemi, düşük bit hızlarında önerilen yönteme göre daha yüksek PSNR değerleri sağlamaktadır. Ancak, bazı yüksek bit hızlarında önerilen yöntemin PSNR değerleri JM 19.1'i geçmektedir. V bileşeni grafikleri incelendiğinde, JM 19.1 yöntemi genellikle önerilen yönteme göre daha yüksek PSNR değerleri sunmaktadır. *Dinner Scene* setinde Y bileşeni için, önerilen yöntem, yüksek bit hızlarında JM 19.1 yöntemine kıyasla önemli ölçüde daha yüksek PSNR değerleri elde etmektedir. Bu, önerilen yöntemin yüksek bit hızlarında luma bilgisini daha iyi koruduğunu göstermektedir. U ve V bileşenleri de göz önünde bulundurulduğunda önerilen yöntem yüksek bit hızlarında JM 19.1'e göre daha iyi performans göstermektedir.

Önerilen yöntemin, özellikle yüksek bit hızlarında JM 19.1 yöntemine göre bu dört video seti için daha yüksek PSNR değerleri sunduğu görülmektedir. Bu, önerilen yöntemin sıkıştırma performansının belirli koşullarda JM 19.1'den daha üstün olabileceğini göstermektedir. Ancak, bazı düşük bit hızlarında JM 19.1'in daha iyi sonuçlar verdiği de göz ardı edilememektedir. Bu nedenle, en uygun yöntemi seçerken hedeflenen bit hızının ve görüntü kalitesinin dikkate alınması gerekmektedir.

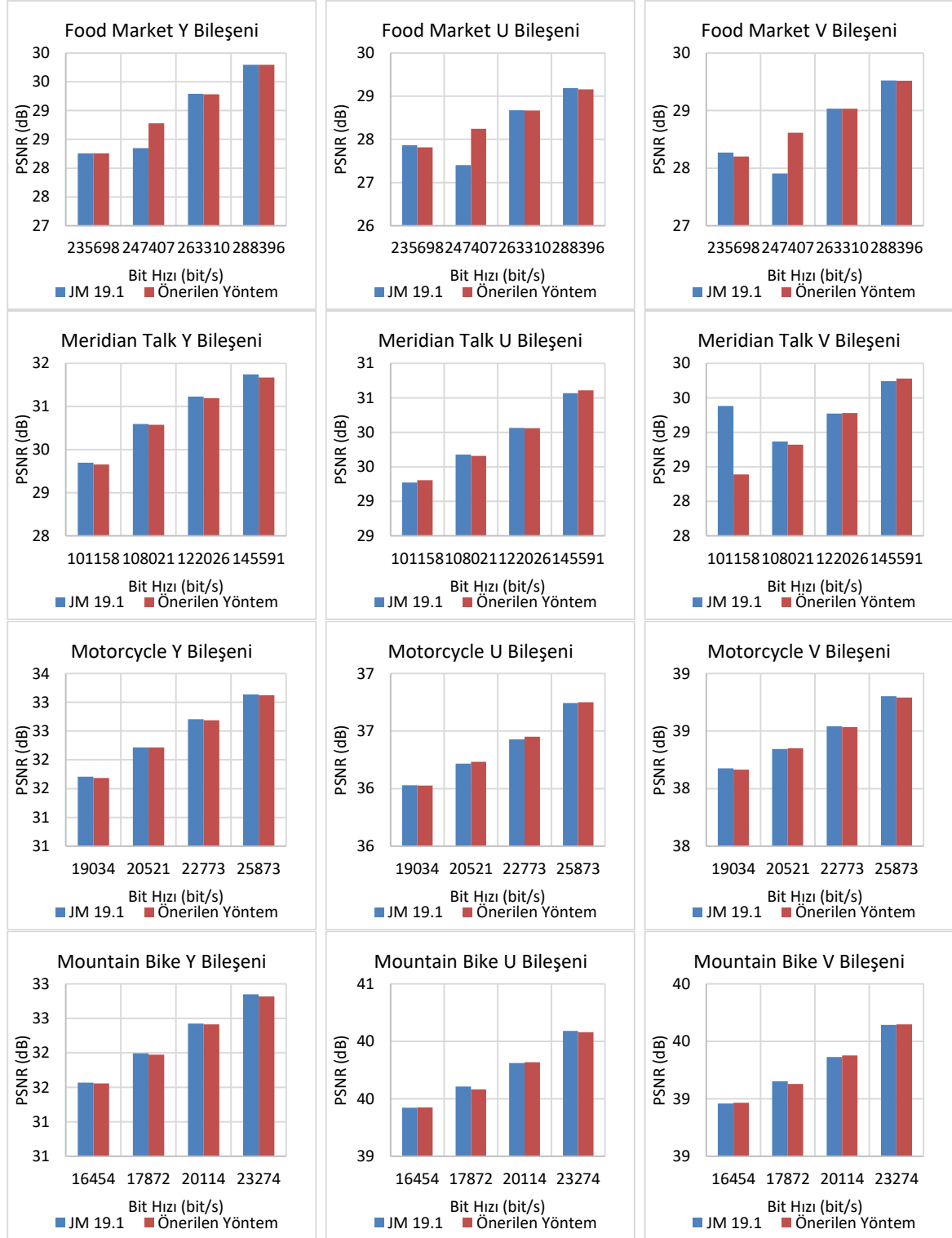


Şekil 5.8: Aerial, Boat, Crowd Run ve Dinner Scene video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları

Food Market, *Meridian Talk*, *Motorcycle* ve *Mountain Bike* setleri için Şekil 5.9'da Y, U ve V bileşenleri için bit hızı ve PSNR arasındaki ilişki verilmektedir. *Food Market* setinde Y bileşeni sonuçları incelendiğinde, en düşük bit hızında JM 19.1, önerilen yöntemden daha yüksek PSNR değerleri vermekte iken, bit hızı arttıkça önerilen yöntemin performansının JM 19.1'i geçtiği gözlemlenmektedir. U bileşeni için, düşük bit hızlarında önerilen yöntem, JM 19.1 ile benzer PSNR değerleri sağlamakta; daha yüksek bit hızlarında ise JM 19.1 yöntemi önerilen yöntemden daha iyi performans göstermektedir. V bileşeninde ise JM 19.1 genel olarak önerilen yöntemden daha iyi PSNR değerler elde etmektedir. Önerilen yöntem sadece en düşük bit hızında JM 19.1'i geçmektedir. *Meridian Talk* setinde, tüm bit hızlarında JM 19.1, Y bileşeni için önerilen yöntemden daha yüksek PSNR değerleri sağlamaktadır. U bileşeni için de benzer bir eğilim gözlemlenmekte ve JM 19.1'in önerilen yöntemden daha iyi performans sergilediği görülmektedir. V bileşeni için de sonuçlar JM 19.1'in önerilen yöntemden iyi olduğunu göstermektedir. *Motorcycle* setinde Y bileşeni grafiklerinde, düşük bit hızlarında JM 19.1, önerilen yöntemden daha yüksek PSNR sunmakta iken, yüksek bit hızlarında önerilen yöntemin daha iyi sonuçlar verdiği görülmektedir. U bileşeninde, özellikle yüksek bit hızlarında JM 19.1, önerilen yöntemle göre daha iyi sonuçlar vermektedir. V bileşeninde, önerilen yöntem, düşük bit hızlarında JM 19.1'i geçmekte; fakat bit hızı arttıkça, JM 19.1 yönteminin PSNR değerleri önerilen yöntemi geride bırakmaktadır. *Mountain Bike* setinde Y ve U bileşenleri JM 19.1 yöntemi, tüm bit hızlarında önerilen yöntemden daha yüksek PSNR değerleri elde etmektedir. V bileşeninde ise önerilen yöntem, düşük bit hızlarında JM 19.1 yöntemine yakın veya daha iyi PSNR değerleri sağlamakta; yüksek bit hızlarında ise JM 19.1 daha iyi sonuçlar vermektedir.

Bu dört video seti ele alındığında, JM 19.1 yöntemi, özellikle U ve V bileşenlerinde çoğunlukla önerilen yöntemden daha iyi PSNR değerleri sağlamaktadır. Bu, bu video setlerinde renk bilgilerinin korunmasında JM 19.1'in daha iyi olduğunu göstermektedir.

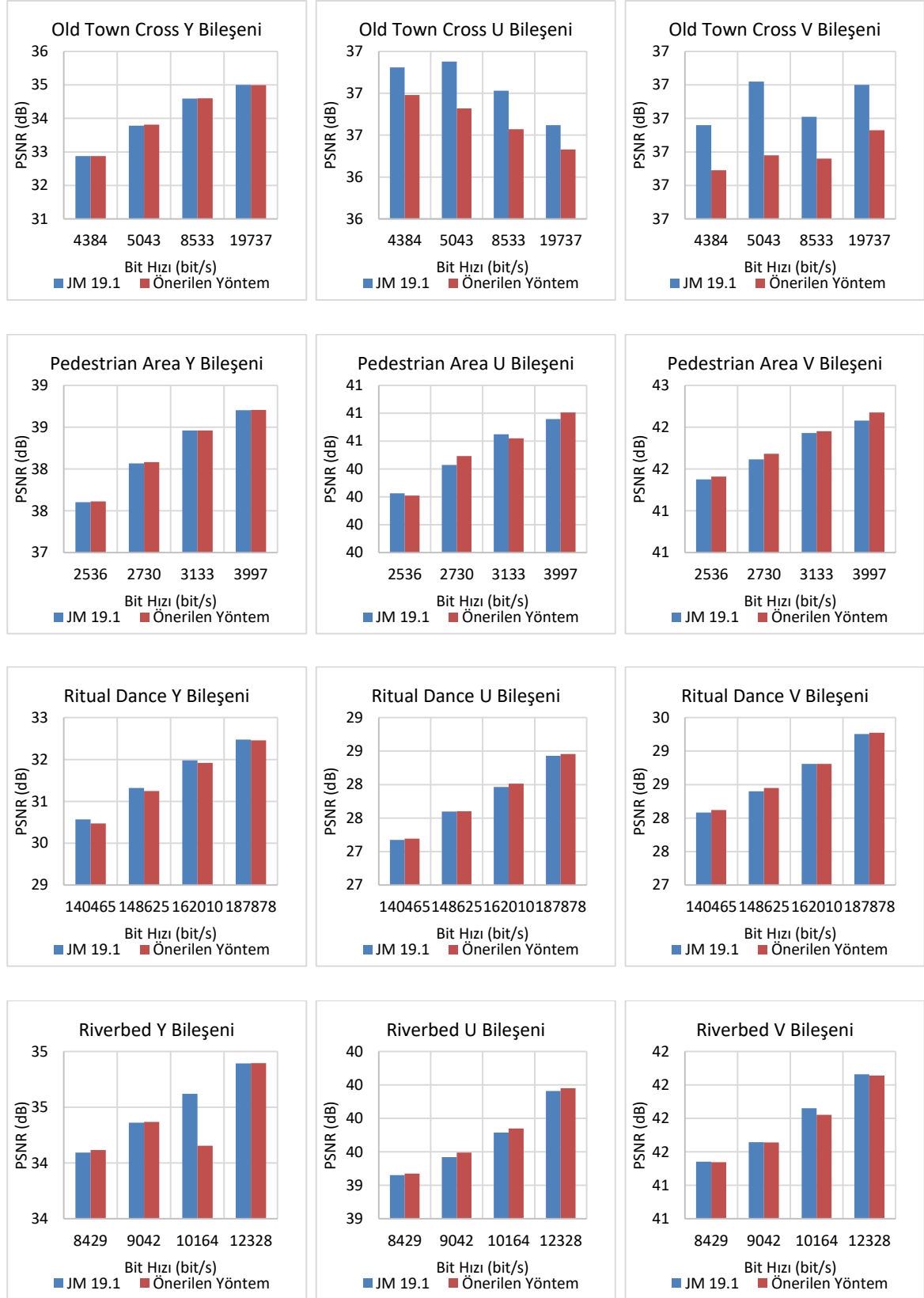
Önerilen yöntem, bazı durumlarda düşük bit hızlarında iyi performans gösterse de, genellikle yüksek bit hızlarında JM 19.1'in gerisinde kalmaktadır. Ancak, Y bileşeni için bazı video setlerinde yüksek bit hızlarında önerilen yöntem lehine de bir eğilim gözlemlenmektedir. Önerilen yöntem belirli koşullar altında Y bileşeni için iyi sonuçlar verebileceği sonucuna ulaşılmaktadır.



Şekil 5.9: Food Market, Meridian Talk, Motorcycle ve Mountain Bike video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları

Old Town Cross, *Pedestrian Area*, *Ritual Dance* ve *Riverbed* setleri için Şekil 5.10'da Y, U ve V bileşenleri için bit hızı ve PSNR arasındaki ilişki verilmektedir. *Old Town Cross* setinde Y bileşeni ile ilgili grafiklere bakıldığında, düşük bit hızlarında JM 19.1 yöntemi genellikle daha yüksek PSNR sunmakta, ancak bit hızı arttıkça önerilen yöntemin performansı iyileşmekte ve bazı noktalarda JM 19.1'i geçmektedir. U bileşeni değerlendirildiğinde, JM 19.1'in genellikle daha yüksek PSNR değerleri sağlamakta olduğu, yüksek bit hızlarında önerilen yöntemin geride kaldığı görülmektedir. V bileşeninde ise önerilen yöntem ve JM 19.1 arasında daha yakın sonuçlar olduğu görülmektedir. Düşük ve yüksek bit hızlarında önerilen yöntemin biraz daha iyi performans gösterdiği durumlar da gözükmektedir. *Pedestrian Area* setinde, düşük bit hızlarında Y bileşeni sonuçlarında önerilen yöntem, daha yüksek PSNR değerleri sunmakta ama bit hızı arttıkça bu fark azalmaktadır. U bileşeninde JM 19.1 yöntemin, tüm bit hızlarında önerilen yöntemden daha yüksek PSNR değerleri sağladığı görülmektedir. V bileşeninde, her iki yöntem de benzer PSNR değerleri sağlamakta, özellikle yüksek bit hızlarında aralarındaki fark çok az olmaktadır. *Ritual Dance* video setinde, Y bileşeni için, JM 19.1 yöntemi, özellikle daha yüksek bit hızlarında önerilen yöntemden daha yüksek PSNR sunmaktadır. U bileşeni için, düşük bit hızlarında önerilen yöntem hafif bir avantaj göstermekte, ama bit hızı arttıkça JM 19.1 yönteminin PSNR değerleri de öne geçmektedir. JM 19.1 genellikle önerilen yönteme göre V bileşeninde daha yüksek PSNR değerleri elde etmektedir. *Riverbed* video seti, Y bileşeni sonuçları, özellikle daha düşük bit hızlarında önerilen yöntemin, JM 19.1'den daha yüksek PSNR değerleri sağladığını göstermektedir. U bileşeninde, düşük bit hızlarında önerilen yöntem, JM 19.1'e göre daha iyi performans sergilemektedir. V bileşeninde ise, JM 19.1 ve önerilen yöntem arasında PSNR değerleri benzer gözükmekte, ancak yüksek bit hızlarında JM 19.1 biraz daha iyi bir performans sunduğu görülmektedir.

Genellikle, JM 19.1 yöntemi düşük bit hızlarında iyi performans göstermekte iken, önerilen yöntem bazı setlerde yüksek bit hızlarında daha iyi sonuçlar vermektedir. U ve V bileşenleri için JM 19.1 genellikle daha yüksek PSNR değerleri sunmakta, bu da renk bileşenlerinin sıkıştırılmasında daha etkili olduğunu göstermektedir. Her iki yöntemin de performansı, video içeriğine ve kullanılan bit hızlarına göre değişiklik göstermektedir. Dolayısıyla, belirli bir kullanım durumu veya hedeflenen kalite için uygun kodlama yönteminin seçilmesi gerektiğini göstermektedir.



Şekil 5.10: Old Town Cross, Pedestrian Area, Ritual Dance ve Riverbed video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları

Rush Field Cut, Skater, Toddler Fountain ve Trees & Grass setleri için Şekil 5.11'de Y, U ve V bileşenleri için bit hızı ve PSNR arasındaki ilişki verilmektedir. *Rush Field Cut* setinin Y bileşeni ele alındığında, önerilen yöntem genellikle JM 19.1'e benzer PSNR değerleri sunmaktadır. U ve V bileşenleri için, tüm bit hızlarında JM 19.1'in, önerilen yöntemden daha iyi performans gösterdiği görülmektedir. *Skater* video setinde, Y bileşeni için JM 19.1 yöntemi, önerilen yöntemle göre her bit hızında daha iyi PSNR değerleri vermektedir. U bileşeni incelendiğinde ise tam aksine, önerilen yöntem tüm bit hızlarında daha iyi PSNR değerleri vermektedir. V bileşeninde ise bit hızlarının çoğunda önerilen yöntem daha yüksek PSNR değerlerine sahip görülmektedir. U ve V bileşenleri göz önünde bulundurulduğunda, bu set için renk bilgisinin önerilen yöntemle daha iyi kodlandığı görülmektedir. *Toddler Fountain* setinde, önerilen yöntem ve JM 19.1 arasında çok az fark olduğu görülmektedir. U ve V bileşenleri göz önüne alındığında, önerilen yöntem sadece en düşük bit hızında daha iyi PSNR sonuçları vermektedir. Geri kalan yerlerde JM 19.1 daha yüksek PSNR değerleri elde etmektedir. *Trees & Grass* setinde Y bileşeni için, JM 19.1, özellikle düşük bit hızlarında daha yüksek PSNR değerleri sağlamaktadır. U ve V bileşenleri göz önünde bulundurulduğunda, bu set için renk bilgisinin JM 19.1 ile daha iyi kodlandığı görülmektedir.

Her video seti ve bileşen için JM 19.1 ve önerilen yöntemin performansı farklılık göstermektedir. Y bileşeni için genellikle iki yöntem birbirlerine yakın PSNR değerleri sağlamakta iken, U ve V bileşenleri için JM 19.1 daha iyi sonuçlar vermektedir. Önerilen yöntem bazı setlerde düşük bit hızlarında avantajlıyken, JM 19.1 genellikle yüksek bit hızlarında daha iyi sonuçlar vermektedir.

Tunnel Flag, Vertical Bees, Vertical Carnaby ve Walking In Street setleri için Şekil 5.12'de Y, U ve V bileşenleri için bit hızı ve PSNR arasındaki ilişki verilmektedir. *Tunnel Flag* setinin Y bileşeni ele alındığında, JM 19.1 referans yazılımın, önerilen yöntemden biraz daha iyi PSNR değerlerine sahip olduğu görülmektedir. U bileşeni incelendiğinde ise önerilen yöntemin, JM 19.1'in PSNR değerlerinin altında kaldığı gözlemlenmektedir. V bileşeninde ise sadece bir bit hızı değerinde önerilen yöntemin PSNR değeri yüksek gözükmekte olup, geri kalan noktalarda PSNR değeri JM 19.1'den düşük çıkmaktadır.



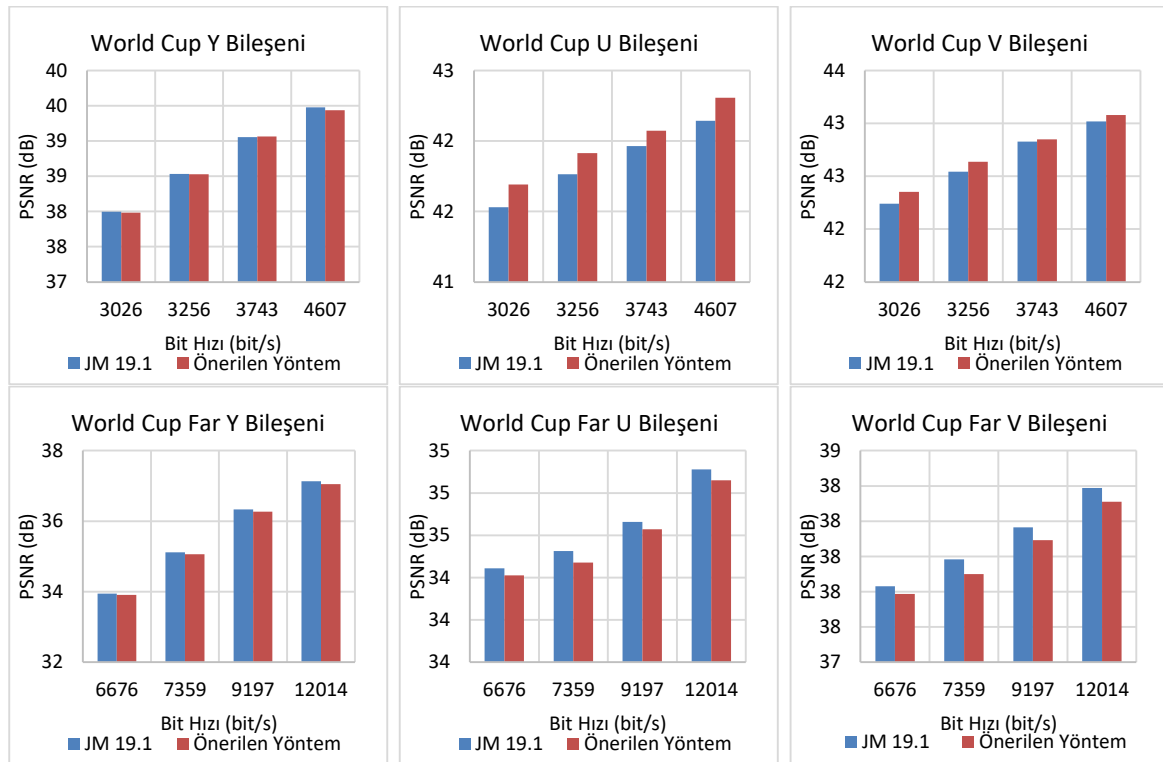
Şekil 5.11: Rush Field Cuts, Skater, Toddler Fountain ve Trees & Grass video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları



Şekil 5.12: Tunnel Flag, Vertical Bees, Vertical Carnaby ve Walking In Street video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları

Vertical Bees video seti için, önerilen yöntemin, en düşük bit hızı hariç Y bileşeninde JM 19.1'den daha yüksek PSNR değerlerine sahip olduğu gözükmektedir. U ve V bileşenleri incelendiğinde ise JM 19.1 yazılımının daha iyi PSNR değerlerine ulaştığı görülmektedir. *Vertical Carnaby* setinde, Y bileşeni için önerilen yöntem en düşük bit hızı noktası haricinde PSNR değeri açısından JM 19.1'in önünde gözükmektedir. U bileşeni içinse, sadece bir bit hızında önerilen yöntem geride kalmakta iken, diğer bit hızı değerlerinde önerilen yöntemin, JM 19.1 referans yazılımından yüksek PSNR değerlerinde olduğu görülmektedir. V bileşeninde de benzer bir durumun söz konusu olduğu anlaşılmaktadır. *Walking In Street* setinde, Y bileşenindeki PSNR değerleri yakın olsa da JM 19.1'in daha iyi sonuç vermiş olduğunu göstermektedir. U ve V bileşenlerinde sadece bir bit hızı noktasında önerilen yöntemin öne geçtiği görülmektedir. Bazı durumlar için, veya hedeflenen kalite için uygun kodlama yönteminin seçilmesi gerektiğini göstermektedir.

World Cup ve *World Cup Far* setleri için Şekil 5.13'te Y, U ve V bileşenleri için bit hızı ve PSNR arasındaki ilişki verilmektedir.



Şekil 5.13: World Cup ve World Cup Far video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları

World Cup video seti Y bileşeni sonuçları incelendiğinde, JM 19.1 en düşük ve en yüksek bit hızı noktalarında PSNR değeri olarak önerilen yöntemden daha iyi değerlere ulaştığı görülmektedir. Fakat U ve V bileşenleri incelendiğinde, önerilen yöntemin JM 19.1'den çok daha iyi sonuçlar verdiği görülmektedir. Bu video seti için ten bilgisinin önerilen yöntem ile daha iyi kodlanacağı görülmektedir. *World Cup Far* video seti incelendiğinde ise Y, U ve V bileşenlerin hepsinde JM 19.1 referans yazılımının daha iyi sonuç verdiği görülmektedir. Pozitif değerler, referans yöntemine göre artış, negatif değerler ise azalış anlamına gelmektedir.

Çizelge 5.14'te 1080p çözünürlükteki video setlerinin JM 19.1 referans yazılımı ve önerilen yöntem ile çalıştırılmaları verilmiştir. ΔT , hesaplama süresi farkını yüzde olarak göstermektedir.

Çizelge 5.14: 1080p çözünürlükteki video setler için önerilen yöntem ve JM 19.1 arasındaki farklar

Video Setleri	$\Delta T(\%)$	Y		U		V	
		BDBR (%)	BDPSNR (dB)	BDBR (%)	BDPSNR (dB)	BDBR (%)	BDPSNR (dB)
<i>Aerial</i>	2,07	-0,15	0,02	0,16	1,06	-0,04	0,00
<i>Boat</i>	1,21	-0,69	0,09	0,58	-0,05	0,56	-0,05
<i>CrowdRun</i>	23,53	1,43	-0,05	3,32	-0,02	3,79	-0,03
<i>DinnerSceneCropped</i>	1,79	0,29	-0,02	-0,47	0,03	-0,28	0,02
<i>FoodMarket</i>	1,07	-3,17	0,04	0,78	0,10	0,85	0,08
<i>MeridianTalk</i>	2,10	1,55	-0,09	0,74	-0,02	5,21	-0,15
<i>Motorcycle</i>	1,72	0,95	-0,04	-0,12	0,00	0,87	-0,02
<i>MountainBike</i>	1,65	1,25	-0,05	0,98	-0,02	0,68	-0,01
<i>OldTownCross</i>	24,92	-0,26	0,02	-59,33	-0,04	804,31	-0,02
<i>PedestrianArea</i>	1,83	0,92	-0,03	-0,79	-0,05	-1,35	0,01
<i>RitualDance</i>	2,86	1,29	-0,10	-0,23	0,02	0,05	-0,01
<i>Riverbed</i>	2,80	-32,70	-0,33	-1,06	0,01	2,25	-0,04
<i>RushFieldCuts</i>	5,80	1,19	-0,05	8,88	-0,07	13,32	-0,08
<i>Skater227</i>	1,71	1,07	-0,10	-0,76	0,04	0,02	0,00
<i>ToddlerFountain</i>	1,50	0,01	0,00	0,61	-0,07	0,51	-0,05
<i>TreesAndGrass</i>	1,90	2,13	-0,09	5,16	-0,06	65,77	-0,27
<i>TunnelFlag</i>	2,03	-0,12	0,00	0,26	-0,02	0,07	-0,02
<i>VerticalBees</i>	23,77	-1,64	0,08	21,54	-0,15	13,51	-0,08
<i>VerticalCarnaby</i>	3,55	0,47	-0,03	0,96	-0,03	0,05	-0,01
<i>WalkingInStreet</i>	-0,62	2,45	-0,09	2,62	-0,08	1,78	-0,09
<i>WorldCup</i>	0,81	2,01	-0,06	-7,31	0,10	-1,36	0,01
<i>WorldCupFar</i>	-2,60	2,21	-0,11	5,28	-0,05	9,73	-0,08

BD oranı ise iki yöntem arasındaki bit hızı farkını yüzde olarak göstermektedir. Negatif değerler, referans yönteme kıyasla daha düşük bit hızı anlamına gelmekte ve bu genellikle daha iyi bir verimlilik olarak değerlendirilmektedir. BD-PSNR(dB) değeri iki yöntem arasındaki kalite farkını desibel cinsinden göstermektedir. Pozitif değerler, referans yönteme göre daha yüksek kalite anlamına gelirken, negatif değerler daha düşük kaliteyi işaret etmektedir.

Riverbed Y bileşeni için BD oranı %-32,697 ve BD-PSNR -0,331 dB olarak gösterilmiş, bu da büyük bir verimlilik kazancı ancak hafif bir kalite kaybını işaret etmektedir. *Old Town Cross* ve *Trees & Grass* video setleri için U ve V bileşenlerinde BD oranlarında ciddi artışlar göze çarpmaktadır. Özellikle *Old Town Cross*'ta U bileşeni için BD-Rate %-59,33, V bileşeni için ise %804,31 şeklinde, bu çok büyük oranlarda bit hızı düşüşüne işaret etmektedir. *Trees & Grass* ve *Rush Field Cuts* V bileşeninde BD-PSNR'lerde büyük artışlar gözükmemektedir. *Trees & Grass* için 65,773 dB ve *Rush Field Cuts* için 13,506 dB gibi değerler göze çarpmaktadır, bu da bazı durumlarda belirgin kalite iyileştirmeleri olduğunu göstermektedir.

5.3.2. 720p Video Setleri ile Yapılan Benzetimler

Controlled Burn, *Driving POV*, *Johnny* ve *Kristen & Sara* setleri için Şekil 5.14'te Y, U ve V bileşenleri için bit hızı ve PSNR arasındaki ilişki verilmektedir. *Controlled Burn* setinin Y bileşeni değerlendirilecek olursa, önerilen yöntemin ve JM 19.1'in birbirlerine çok yakın değerler vermekte olduğu görülmektedir. U ve V bileşenlerine bakıldığında ise, JM 19.1 daha iyi sonuçlar vermektedir. *Driving POV* video setinde ise, Y bileşeni için önerilen yöntem en düşük bit hızı için daha iyi sonuç vermekte iken diğer durumlarda benzer sonuçlar üretmektedirler. U ve V bileşenleri için ise JM 19.1 yazılımının daha iyi sonuç verdiği görülmektedir. *Johnny* video setinde, Y bileşeni için JM 19.1 düşük bit hızlarında daha iyi sonuç vermekte iken yüksek bit hızlarında önerilen yöntem daha iyi PSNR değerleri döndürmektedir. U ve V bileşenlerinde sadece bir noktada önerilen yöntem iyi değer vermekte iken geri kalan bit hızlarında JM 19.1 iyi sonuçlar vermektedir. *Kristen & Sara* video setlerinde Y bileşeni için bit hızı yükseldikçe önerilen yöntemin JM 19.1'i geçtiği görülmektedir. U ve V bileşenleri için ise JM 19.1'in bir durum hariç daha yüksek PSNR değerleri verdiği görülmektedir.



Şekil 5.14: Controlled Burn, Driving POV, Johnny ve Kristen & Sara video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları

Roller Coaster, Vidyo3, Vidyo4 ve West Wind Easy setleri için Şekil 5.15'te Y, U ve V bileşenleri için bit hızı ve PSNR arasındaki ilişki verilmektedir.



Şekil 5.15: Roller Coaster, Vidyo3, Vidyo4 ve West Wind Easy video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları

Roller Coaster video setinde Y bileşeni için elde edilen sonuçlar incelendiğinde, JM 19.1 ve önerilen yöntem yakın PSNR değerleri vermektedir. U bileşeni incelendiğinde, en düşük bit hızında önerilen yöntem iyi sonuç vermekte iken diğer bit hızlarında JM 19.1 öne geçmektedir. V bileşeninde ise düşük bit hızlarında JM 19.1 ile yakın bir başarımla göstermekte olan önerilen yöntemin bit hızı arttıkça geri kaldığı görülmektedir. *Vidyo3* video setinde, önerilen yöntemin Y bileşeninde bit hızı arttıkça PSNR değerinin JM 19.1'in önüne geçtiği görülmektedir. U ve V bileşenlerinde ise genel itibari ile JM 19.1 daha iyi sonuç vermektedir. *Vidyo4* video setinde Y, U ve V bileşenlerinde önerilen yöntemin birkaç durum dışında daha iyi sonuçlar verdiği görülmektedir. *West Wind Easy* video setinde Y bileşeni için JM 19.1 daha iyi sonuçlar vermekte iken, U ve V bileşenlerinde önerilen yöntemin PSNR değerleri daha yüksek değerlere sahip olmaktadır.

Çizelge 5.15'te 720p çözünürlükteki video setlerinin JM 19.1 referans yazılımı ve önerilen yöntem ile çalıştırılmaları verilmiştir.

Çizelge 5.15: 720p çözünürlükteki video setler için önerilen yöntem ve JM 19.1 arasındaki farklar

Video Setleri	$\Delta T(\%)$	Y		U		V	
		BDBR (%)	BDPSNR (dB)	BDBR (%)	BDPSNR (dB)	BDBR (%)	BDPSNR (dB)
<i>ControlledBurn</i>	25,94	0,33	-0,02	24,38	-0,05	46,03	-0,06
<i>DrivingPOV</i>	0,59	-0,85	0,08	1,33	-0,11	1,11	-0,09
<i>Johnny</i>	25,73	-0,43	0,01	2,92	0,01	10,73	-0,06
<i>KristenAndSara</i>	25,64	-0,40	0,04	10,51	-0,06	4,10	-0,01
<i>RollerCoaster</i>	0,34	-0,05	0,02	1,22	-0,05	0,40	-0,04
<i>Vidyo3</i>	17,80	-0,26	-0,08	31,97	-0,07	6,09	-0,09
<i>Vidyo4</i>	26,05	-0,58	0,02	5,83	0,01	-3,46	0,01
<i>WestWindEasy</i>	21,21	2,73	-0,11	-2,93	0,05	-5,03	-0,03

Controlled Burn video seti Y bileşeni için BD oranının %0,33 değeriyle arttığı, ancak PSNR'ın -0,019 dB değeri ile hemen hemen sabit kaldığı görülmektedir. U ve V bileşenleri için BD-oranı oldukça yüksek artışlar göstermekte, bu da çok fazla bit hızı artışına işaret etmektedir, ancak PSNR değerlerinde çok az bir düşüş görülmektedir. *Driving POV* video setinde ise tüm bileşenlerde negatif BD-oran değerleri gözlemlenmekte ki bu daha düşük bit hızı ile kodlama yapıldığını göstermektedir. Y bileşeni için BD-PSNR bir miktar artmakta,

U ve V için ise az da olsa düşüş gözlenmektedir. *Johnny* ve *Kristen & Sara* video setinde de ΔT oldukça yüksek çıkmış ve kodlama süresinde büyük bir artış olduğunu göstermektedir. Bununla birlikte, her iki set için de Y bileşeninde BD-oranı değerleri negatif gözükmekte, bu da verimliliğin arttığını göstermektedir. U bileşenleri için ve V bileşeni için oldukça yüksek BD-oran artışları görülmektedir. *West Wind Easy* Y bileşeninde BD-oranı pozitif değer göstermekte, yani bit hızı artmaktadır. U ve V bileşenleri için BD-oranı oldukça büyük oranlarda azalmakta ve bu da bit hızının düştüğü anlamına gelmektedir.

5.3.3.360p Video Setleri ile Yapılan Benzetimler

Çizelge 5.16'da 360p çözünürlükteki video setlerinin JM 19.1 referans yazılımı ve önerilen yöntem ile çalıştırılmaları verilmiştir.

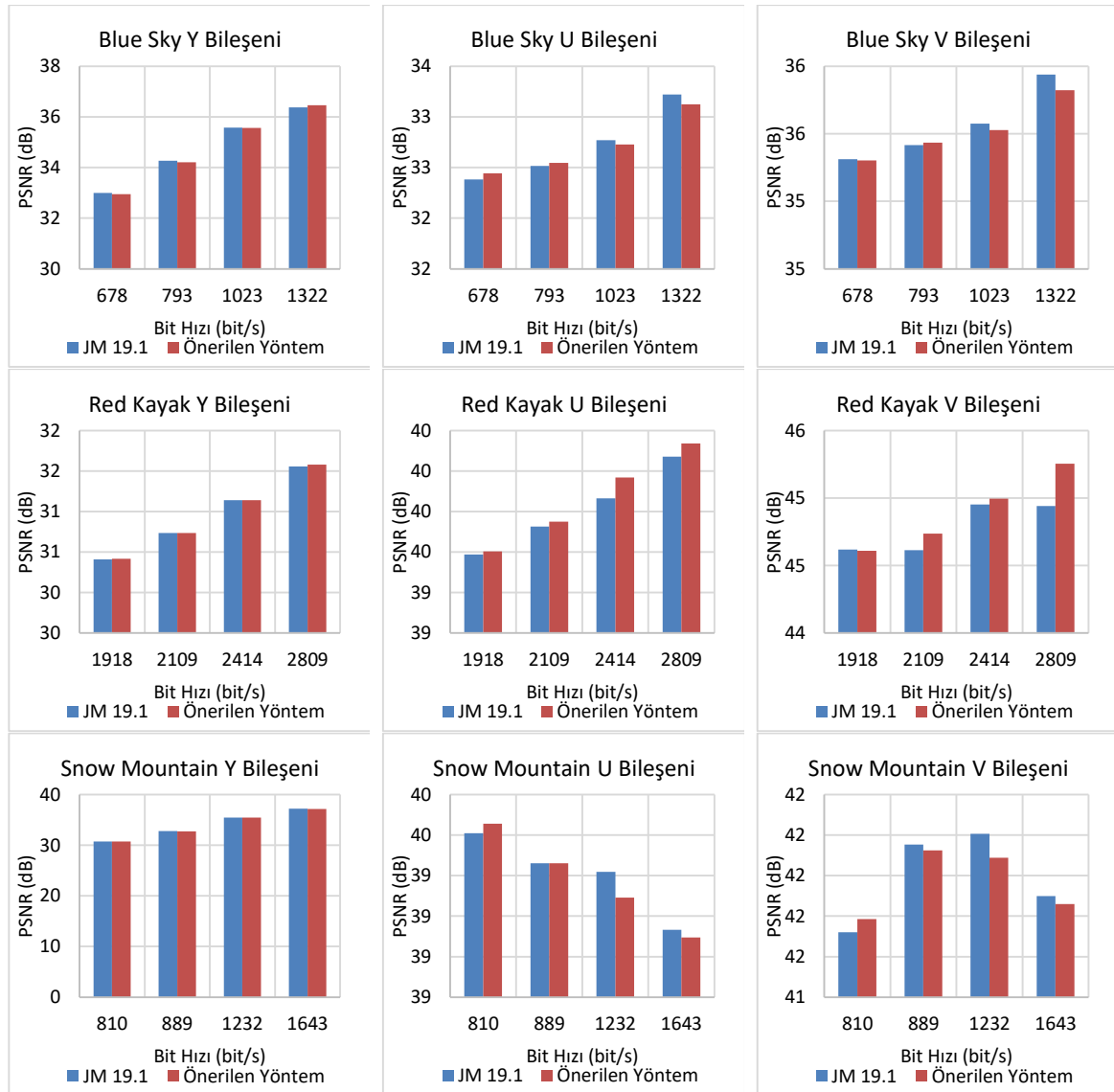
Çizelge 5.16: 360p çözünürlükteki video setler için önerilen yöntem ve JM 19.1 arasındaki farklar

Video Setleri	$\Delta T(\%)$	Y		U		V	
		BDBR (%)	BDPSNR (dB)	BDBR (%)	BDPSNR (dB)	BDBR (%)	BDPSNR (dB)
<i>BlueSky</i>	7,89	1,03	-0,05	3,88	-0,03	9,32	-0,04
<i>RedKayak</i>	7,95	0,07	0,00	-4,65	0,06	-40,84	0,09
<i>SnowMountain</i>	25,64	0,56	0,11	-32,41	-0,10	26,32	-0,07
<i>SpeedBag</i>	1,84	-1,43	0,06	2,51	-0,07	1,80	-0,04
<i>Stockholm</i>	16,88	1,68	-0,08	-28,95	0,05	-7,89	0,05
<i>TouchdownPass</i>	6,00	-0,74	0,02	-0,77	-0,04	-18,01	0,00

Snow Mountain ve *Stockholm* video setleri için ΔT 'nin çok fazla arttığı görülmekte, bu daha uzun kodlama süresini göstermektedir. *Speed Bag* video seti ise en düşük ΔT değerine sahip gözükmektedir. Video setlerinin Y bileşenindeki sonuçları incelendiğinde *Speed Bag* ve *Touchdown Pass* video setlerinde önerilen yöntemin daha başarılı olduğu görülmektedir. U bileşeni için sonuçlar incelendiğinde ise önerilen yöntemin başarısının önemli bir ölçüde arttığı gözükmektedir. Özellikle %-32,41 ve %-28,95 ile *Snow Mountain* ve *Stockholm* video setlerinde önerilen yöntemin iyi sonuçlar verdiği görülmektedir. V bileşeninde ise *Red Kayak* ve *Touchdown Pass* setlerinde önerilen yöntemin başarılı olduğu

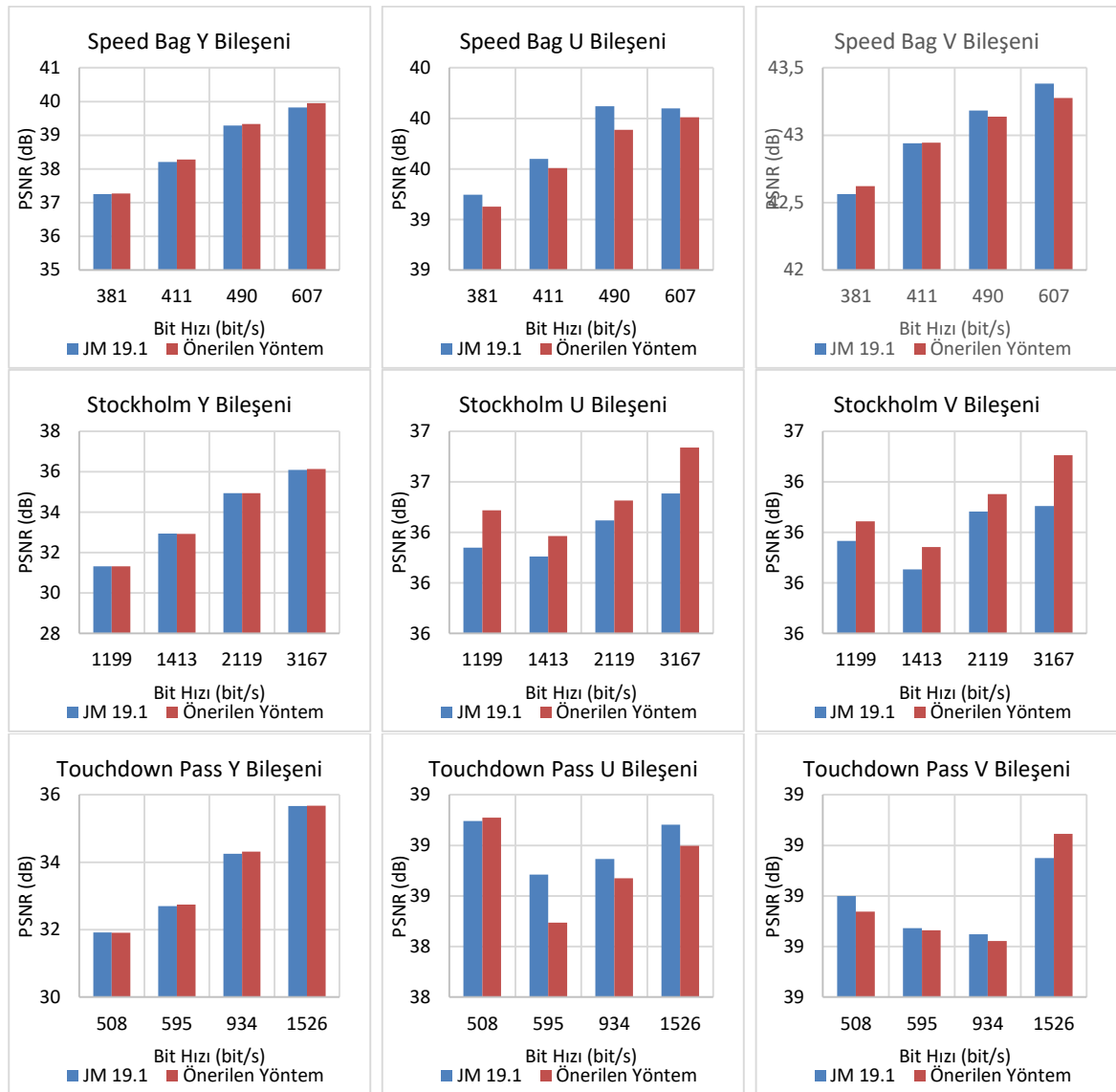
görülmektedir. Bu bileşende JM 19.1'in %26,32 BD oranı değeri ile *Snow Mountain* setinde çok daha iyi sonuç aldığı görülmektedir.

Blue Sky, *Red Kayak* ve *Snow Mountain* setleri için Şekil 5.16'da Y, U ve V bileşenleri için bit hızı ve PSNR arasındaki ilişki verilmektedir. *Blue Sky* video setinde, Y bileşeni incelendiğinde, bit hızı arttıkça önerilen yöntemin PSNR değerlerinin JM 19.1 referans yazılımını geçtiği görülmektedir. U ve V bileşenlerinde ise, önerilen yöntem düşük bit hızlarında daha iyi sonuç vermekte iken bit hızı arttıkça JM 19.1'in öne geçtiği gözlemlenmektedir.



Şekil 5.16: *Blue Sky*, *Red Kayak* ve *Snow Mountain* video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları

Red Kayak seti ele alındığında, Y bileşenindeki sonuçlar bit hızı arttıkça önerilen yöntemin JM 19.1'i geçtiğini göstermektedir. U ve V bileşenlerinde de bu durum gözlemlenmekte iken ek olarak önerilen yöntemin çok daha iyi sonuçlar verdiği görülmektedir. *Snow Mountain* video setinde Y bileşeni için iki yöntemde neredeyse birebir aynı başarıyı göstermektedir. U ve V bileşenlerinde PSNR ve bit hızı arasındaki ilişkinin önceki video setlerinde karşılaşılandan çok farklı olduğu görülmektedir. Bunun da sebebinin videodaki dağ görüntüsünün çok durağan olduğu aklı gelmektedir. *Speed Bag*, *Stockholm* ve *Touchdown Pass* setleri için Şekil 5.17'de Y, U ve V bileşenleri için bit hızı ve PSNR arasındaki ilişki verilmektedir.



Şekil 5.17: Speed Bag, Stockholm ve Touchdown Pass video setleri için Y, U ve V bileşenlerinin önerilen yöntem ve JM 19.1 yazılımı ile kodlanmasının sonuçları

Speed Bag video setinde Y bileşenindeki sonuçlar değerlendirildiğinde, önerilen yöntemin tüm bit hızlarında JM 19.1'e göre yüksek PSNR değerleri ürettiği görülmektedir. U ve V bileşenleri incelendiğinde ise JM 19.1 referans yazılımının daha başarılı olduğu görülmektedir. *Stockholm* seti ele alındığında, Y bileşenindeki sonuçlar bit hızı arttıkça önerilen yöntemin JM 19.1'i geçtiğini göstermektedir. U ve V bileşenlerinde de bu durum gözlemlenmekte iken ek olarak önerilen yöntemin çok daha iyi sonuçlar verdiği görülmektedir. *Touchdown Pass* seti ele alındığında, Y bileşenindeki sonuçlar bit hızı arttıkça önerilen yöntemin JM 19.1'i geçtiğini göstermektedir. U ve V bileşenleri incelendiğinde sadece birer noktada önerilen yöntemin daha iyi sonuç verdiği gözlemlenmektedir.

6. SONUÇ VE ÖNERİLER

Dijital video iletimi, günümüzde ağ trafiğinin önemli bir bölümünü oluşturmakta ve sürekli olarak artmaktadır. Yüksek kalitede video içeriğinin paylaşılması, canlı yayınların gerçek zamanlı olarak aktarılması ve çevrimiçi video platformlarının popüleritesindeki artış, video trafiğinin sürekli büyümesine neden olmaktadır. Bu durum araştırmacıların video kodlama standartlarını sürekli olarak güncellemelerini ve yeni standartlar oluşturmalarını zorunlu kılmaktadır. Bu çalışmanın amacı, video kodlama standartlarının geliştirilmesi ve iyileştirilmesinde önemli bir adım olan hareket kestirimindeki çoklu referans bloklarının kullanımı üzerine odaklanmaktır.

Bu tez çalışmasında, video uç noktalarındaki artan hesaplama gücü ile birlikte, video kalitesini bir miktar karmaşıklık maliyetiyle artırma olasılığı deneysel olarak irdelenmiştir. Bununla birlikte, birden fazla ağırlıklı referans bloğunun kullanımının video kalite gücünü önemli ölçüde iyileştirdiği deneysel olarak gösterilmiştir. Çoklu referans tekniği, blok başına ortalama olarak 1,5 dB'lik PSNR artışı sağlarken, aynı zamanda blok başına 3-5 bit artışına neden olmuştur. Bu, MPEG sıkıştırma şemalarında bile temel algoritmalarla kabul edilebilir bir düzeydedir. Aynı zamanda yöntemin eski ve yeni MPEG algoritmalarına/yazılımlarına uyarlanması da kolaydır. Her iki yaklaşımda video kalitesini artırırken daha fazla hesaplama kaynağı ve biraz artmış bit hızı maliyetiyle beraber gelmektedir.

H.264 standardının referans yazılımı JM 19.1 üzerinde çift yönlü öngörü aşamasına müdahil olup en küçük kareler yöntemi ile referans bloklarının ağırlıklarının hesaplanmasının sonuçları test edilmiştir. Kullanılan veri setleri dahilinde planlanan deneysel çalışmaların çıktıları sonucunda, önerilen yöntemin sıkıştırma performansının belirli koşullarda JM 19.1'den daha üstün olabileceği gösterilmektedir. Bu nedenle, en uygun yöntemi seçerken hedeflenen bit hızının ve görüntü kalitesinin dikkate alınması gerekmektedir.

Gelecek çalışmalarda amaçlanan; sayısal-hareketli video iletimindeki artan talepleri karşılamak ve video kodlama standartlarını daha da geliştirmek için Yüksek Verimli Video Kodlama (HEVC) ve sonrasında VVC (Versatile Video Coding) üzerine odaklanmaktır.

Referans blokların lineer toplamını kullanarak elde edilen önerilen yöntem, mevcut video kodlama standartlarının performansını artırmaya yönelik değerli bir katkı sağlamaktadır. Elde edilen sonuçlar dahilinde, önerilen yöntem daha yüksek kalitede videoların aktarılabilmesini ve depolanabilmesini mümkün kılarak yöntemin video endüstrisine de katkıda bulunması beklenmektedir.

KAYNAKLAR DİZİNİ

- Aoki, H. and Miyamoto, Y., "An H.264 weighted prediction parameter estimation method for fade effects in video scenes," 2008 15th IEEE International Conference on Image Processing, San Diego, CA, USA, 2008, pp. 2112-2115, doi: 10.1109/ICIP.2008.4712204.
- Artuğer, F., Özkaynak, F. (2018). JPEG Sıkıştırma Algoritmasının Dünü, Bugünü ve Geleceği. 30(3):161-167. <https://dergipark.org.tr/tr/download/article-file/539391>
- Asikuzzaman, M., Ahmmed, A., Pickering, M.R. and Sikora, T., "Edge Oriented Hierarchical Motion Estimation For Video Coding," 2020 IEEE International Conference on Image Processing (ICIP), Abu Dhabi, United Arab Emirates, 2020, pp. 1221-1225, doi: 10.1109/ICIP40778.2020.9190852.
- Blaser M. et al., "Low-Complexity Geometric Inter-Prediction for Versatile Video Coding," 2019 Picture Coding Symposium (PCS), Ningbo, China, 2019, pp. 1-5, doi: 10.1109/PCS48520.2019.8954504.
- Brand, F., Seiler, J. and Kaup, A., "P-Frame Coding with Generalized Difference: A Novel Conditional Coding Approach," 2022 IEEE International Conference on Image Processing (ICIP), Bordeaux, France, 2022, pp. 1266-1270, doi: 10.1109/ICIP46576.2022.9897294.
- Boyce, J., Suehring, K., Li, X., Seregin, V. (2018). JVET-J1010: JVET common test conditions and software reference configurations.
- Chai, X., Wu, H., Gan, Z., Chen, Y., Hun, D., Zhang, Y. (2021). An Efficient Approach for Encrypting Double Color Images Into A Visually Meaningful Cipher Image Using 2D Compressive Sensing. Info SCI: 556, 305–340. DOI:10.1016/j.ins.2020.10.007
- Chen, C., Han, J. and Xu, Y., "A Hybrid Weighted Compound Motion Compensated Prediction for Video Compression," 2018 Picture Coding Symposium (PCS), San Francisco, CA, USA, 2018, pp. 223-227, doi: 10.1109/PCS.2018.8456241.
- Chen, C., Xiu, X., He, Y. and Ye, Y., "Generalized bi-prediction method for future video coding," 2016 Picture Coding Symposium (PCS), Nuremberg, Germany, 2016, pp. 1-5, doi: 10.1109/PCS.2016.7906342.
- Chen, Y., Zhang, C., Cui, M., Luo, Y., Wu, T., Liang, X. (2022). Joint Compressed Sensing and JPEG Coding Cased Secure Compression Scheme in OFDM-PON, Optics Communications 510. <https://doi.org/10.1016/j.optcom.2022.127901>
- Coding of Audio-Visual Objects-Part 2: Visual, ISO/IEC 14496-2 (MPEG-4 Visual Version 1), ISO/IEC JTC 1, 1999.

KAYNAKLAR DİZİNİ

- Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to About 1.5 Mbit/s-Part 2: Video, ISO/IEC 11172-2 (MPEG-1), ISO/IEC JTC 1, 1993.
- Çimtay, Y. (2012). Doğrusal Olmayan Hareket Kestirimi İle Video Çerçeve Ara Değerlendirilmesi [Yüksek Lisans Tezi, Eskişehir Osmangazi Üniversitesi Fen Bilimleri Enstitüsü]. YÖK Ulusal Tez Merkezi.
- Deng, Z., Zhang, K., and Zhang, L., "Geometry Partitioning with Motion Vector Difference for Video Coding," 2022 IEEE International Conference on Image Processing (ICIP), Bordeaux, France, 2022, pp. 2271-2275, doi: 10.1109/ICIP46576.2022.9898040.
- Eisert, P., Wiegand, T., Girod, B. (2000). Model-Aided Coding: A New Approach to Incorporate Facial Animation Into Motion-Compensated Video Coding. IEEE Transactions on Circuits and Systems for Video Technology, 10(3): 344-358. <https://doi.org/10.1109/76.836279>
- Erabadda, B., Mallikarachchi, T., Kulupana, G. and Fernando, A., "Improving HEVC Coding Efficiency Using Virtual Long-Term Reference Pictures," 2020 IEEE 9th Global Conference on Consumer Electronics (GCCE), Kobe, Japan, 2020, pp. 401-402, doi: 10.1109/GCCE50665.2020.9291917.
- Fan, K., Wang, R., Li, G., Gao, W. (2019). Efficient Prediction Methods With Enhanced Spatial-Temporal Correlation for HEVC. IEEE Transactions on Circuits and Systems for Video Technology, 29(12): 3716-3728. <https://doi.org/10.1109/TCSVT.2018.2885002>
- Flierl M. and Girod, B., "Generalized B pictures and the draft H.264/AVC video-compression standard," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, no. 7, pp. 587-597, July 2003, doi: 10.1109/TCSVT.2003.814963.
- Glantz, A., Krutz, A., Sikora, T. (2010). Adaptive Global Motion Temporal Prediction for Video Coding. 28th Picture Coding Symposium, 202-205. <https://doi.org/10.1109/PCS.2010.5702463>
- HoangVan, X. and Nguyen, H. -H., "Enhancing Quality for VVC Compressed Videos with Multi-Frame Quality Enhancement Model," 2020 International Conference on Advanced Technologies for Communications (ATC), Nha Trang, Vietnam, 2020, pp. 172-176, doi: 10.1109/ATC50776.2020.9255448.
- Huang, X., Zhang, Q., Zhao, X., Zhang, W., Zhang, Y., Gan, Y. (2017). Fast Inter-Prediction Mode Decision Algorithm for HEVC. Signal, Image and Video Processing, 11(1): 33-40. <https://doi.org/10.1007/s11760-016-0887-4>

KAYNAKLAR DİZİNİ

- Itani, Y., Sekiguchi, S., Yamada, Y. (2010). Adaptive Direct Vector Derivation for Video Coding. 28th Picture Coding Symposium, 190-193. <https://doi.org/10.1109/PCS.2010.5702458>
- ITU-T and ISO/IEC. (2020). High Efficiency Video Coding, document ITU-T Rec. H.265 and ISO/IEC 23008-2 (HEVC).
- Jiwani, L. K. (2018). Least Square Intra-Prediction Mode Prediction Based Video Compression. 2018 Conference on Information and Communication Technology (CICT), 1-6. <https://doi.org/10.1109/INFOCOMTECH.2018.8722371>
- Keller, Y., Averbuch, A. (2003). Fast Gradient Methods Based on Global Motion Estimation for Video Compression. IEEE Transactions on Circuits and Systems for Video Technology, 13(4):300-309. <https://doi.org/10.1109/TCSVT.2003.811360>
- Kim, S. H., Kim, Y. K., Lee, S.-U. (2002). Adaptive Multiple Reference Frame Based Scalable Video Coding Algorithm. Proceedings. International Conference on Image Processing, 2, II-II. <https://doi.org/10.1109/ICIP.2002.1039880>
- Kudo, S., Kitahara, M., Shimizu, A. (2016). Motion Vector Prediction Methods Considering Prediction Continuity in HEVC. 2016 Picture Coding Symposium (PCS), 1-5.
- Lainema, J., Bossen, F., Han, W.-J., Min, J., Uğur K. (2012). Intra Coding of the HEVC Standard, IEEE Transactions on Circuits and Systems for Video Technology, 22,(12):1792-1801. DOI: 10.1109/TCSVT.2012.2221525
- Li, B., Han, J. and Xu, Y., "An Efficient Scheme of Multi-Hypothesis Motion Compensated Prediction for Video Coding Applications," 2022 IEEE International Conference on Image Processing (ICIP), Bordeaux, France, 2022, pp. 2231-2235, doi: 10.1109/ICIP46576.2022.9897276.
- Lin, J.-L., Chen, Y.-W., Tsai, Y.-P., Huang, Y.-W., Lei, S. (2011). Motion Vector Coding Techniques for HEVC. 2011 IEEE 13th International Workshop on Multimedia Signal Processing, 1-6. <https://doi.org/10.1109/MMSP.2011.6093817>
- Ma, C., Liu, D., Peng, X., Wu, F., Li, H. and Wang, T., "Reference Clip for Inter Prediction in Video Coding," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 29, no. 1, pp. 130-143, Jan. 2019, doi: 10.1109/TCSVT.2017.2769884.
- McCann, K. D., Mrak, M., Weerakkody, R., Baroncini, V., Tan, T. K., Sullivan, G. J., Ramzan, N., Ohm, J.-R. (2014). HEVC Subjective Video Quality Test Results. International Broadcasting Convention (IBC) 2014 Conference, 9.3-9.3. <https://doi.org/10.1049/ib.2014.0029>

KAYNAKLAR DİZİNİ

- Mesut, A. (2006). Veri Sıkıştırma Yeni Yöntemler [Doktora Tezi, Trakya Üniversitesi Fen Bilimleri Enstitüsü]. YÖK Ulusal Tez Merkezi.
- Murakami, T., Saito, S., Komatsu, Y., Nakamura, K., Yokoyama, T. (2010). Enhancement of H.264/AVC for Higher Coding Efficiency Using Motion Estimation Between Reference Frames. *IEEE Transactions on Consumer Electronics*, 56(2):925-929. <https://doi.org/10.1109/TCE.2010.5506021>
- Nasır, S. (2007). 2-Boyutlu Hareket Kestirimi ve Video Çerçeve Aradeğerlenmesinde Kullanılması [Yıldız Teknik Üniversitesi Tezi, Fen Bilimleri Enstitüsü]. YÖK Ulusal Tez Merkezi.
- Rhee, C., Lee, K., Kim, T., Lee, H.-J. (2012). A survey of fast mode decision algorithms for inter-prediction and their applications to high efficiency video coding, *IEEE Transactions on Consumer Electronics*, vol. 58, no. 4.
- Rojas, L., Garcia, L.-I., Rojas, C. G. (2006). Low Cost Visual Event Detection over JPEG Flows. 16 - 16. 10.1109/CONIELECOMP.2006.38.
- Samet, A., Litayem, N., Zouch, W., Ben Ayed, M. A., Masmoudi, N. (2015). New Horizontal Diamond Search Motion Estimation Algorithm For H.264/AVC.
- Sampaio, F., Bampi, S., Grellert, M., Agostini, L., Mattos, J. (2012). Motion Vectors Merging: Low Complexity Prediction Unit Decision Heuristic for the Inter-Prediction of HEVC Encoders. 2012 IEEE International Conference on Multimedia and Expo, 657-662. <https://doi.org/10.1109/ICME.2012.37>
- Shen, Y., Zhang, D., Huang, C. and Li, J., "Adaptive weighted prediction in video coding," 2004 IEEE International Conference on Multimedia and Expo (ICME) (IEEE Cat. No.04TH8763), Taipei, 2004, pp. 427-430 Vol.1, doi: 10.1109/ICME.2004.1394220.
- Steinbach, E., Wiegand, T., & Girod, B. (1999). Using multiple global motion models for improved block-based video coding. *Proceedings 1999 International Conference on Image Processing (Cat. 99CH36348)*, 2, 56-60 vol.2. <https://doi.org/10.1109/ICIP.1999.822854>
- Su, Y., Sun, M.-T. (2006). Fast Multiple Reference Frame Motion Estimation for H.264/AVC. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(3):447-452. <https://doi.org/10.1109/TCSVT.2006.869970>
- Sullivan, G.-J., Ohm, J.-R., Han, W.-J., Wiegand, T. (2012). Overview of the High Efficiency Video Coding (HEVC) Standard, *IEEE transactions on circuits and systems for video technology*, 22(12). DOI: 10.1109/TCSVT.2012.2221191

KAYNAKLAR DİZİNİ

- Sun, Y. Chen, F., Wang, L. and Pu, S., "Angular Weighted Prediction for Next-Generation Video Coding Standard," 2021 IEEE International Conference on Multimedia and Expo (ICME), Shenzhen, China, 2021, pp. 1-6,
- Suzuki, Y., Boon, C. S. (2010). An Improved Low Delay Inter Frame Coding Using Template Matching Averaging. 28th Picture Coding Symposium, 370-373.
- Tao, J., Li, S., Zhang, X., Wang, Z. (2019). Towards robust image steganography, IEEE Trans. Circuits Syst. Video Technol. 29(2):594–600, DOI: 10.1109/TCSVT.2018.2881118
- Veena, S. K., & Rao, M. K. (2021). Handling Tradeoff Between Visual Perception and Compression in Video Codec Using Optimal Intra-Frame Prediction. Wireless Personal Communications, 119(2):1515-1540.
- Video Codec for Audiovisual Services at px64 kbit/s, ITU-T Rec. H.261, version 1: Nov. 1990, version 2: Mar. 1993.
- Video Coding for Low Bit Rate Communication, ITU-T Rec. H.263, Nov. 1995.
- Wang, Y., Fan, X., Jia, C., Zhao, D., Gao, W. (2018). Neural Network Based Inter Prediction for HEVC. 2018 IEEE International Conference on Multimedia and Expo (ICME), 1-6. <https://doi.org/10.1109/ICME.2018.8486600>
- Winken, M., Bartnik, C., Schwarz, H., Marpe, D. and Wiegand, T., "Weighted Multi-Hypothesis Inter Prediction for Video Coding," 2019 Picture Coding Symposium (PCS), Ningbo, China, 2019, pp. 1-5, doi: 10.1109/PCS48520.2019.8954505.
- Xiong, J., Li, H., Wu, Q., Meng, F. (2014). A Fast HEVC Inter CU Selection Method Based on Pyramid Motion Divergence. IEEE Transactions on Multimedia, 16(2):559-564.
- Zhang, J., Li, B., Li, H. (2016). An Efficient Fast Mode Decision Method for Inter Prediction in HEVC. IEEE Transactions on Circuits and Systems for Video Technology, 26(8): 1502-1515. <https://doi.org/10.1109/TCSVT.2015.2461991>
- Zhang, J., Li, B., Li, H. (2016). An Efficient Fast Mode Decision Method for Inter Prediction in HEVC. IEEE Transactions on Circuits and Systems for Video Technology, 26(8): 1502-1515. <https://doi.org/10.1109/TCSVT.2015.2461991>
- Zhang, S., Wang, Y., Kang, J., & Li, H. (2008). A New Approach to Fast Multiple Reference Frame Motion Estimation for H.264. 2008 International Symposium on Computer Science and Computational Technology, (2):254-258.
- Zhao, X., Lei, Z., Norkin, A., Daede, T., Tourapis, A. (2021). AOM common test conditions v2. 0. Alliance for Open Media, Codec Working Group Output Document.