

**KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

**SAYISAL DELİLLERDE PASİF DOĞRULAMA YÖNTEMLERİNE
DAYALI SAHTECİLİK TESPİTİ**

DOKTORA TEZİ

Bil. Yük. Müh. Beste ÜSTÜBİOĞLU

**NİSAN 2018
TRABZON**



KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünce

Unvanı Verilmesi İçin Kabul Edilen Tezdir.

Tezin Enstitüye Verildiği Tarih : / /

Tezin Savunma Tarihi : / /

Tez Danışmanı :

Trabzon

KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Bilgisayar Mühendisliği Anabilim Dalında
Beste ÜSTÜBİOĞLU Tarafından Hazırlanan

SAYISAL DELİLLERDE PASİF DOĞRULAMA YÖNTEMLERİNE DAYALI
SAHTECİLİK TESPİTİ

başlıklı bu çalışma, Enstitü Yönetim Kurulunun 20 /03 /2018 gün ve 1745 sayılı
kararıyla oluşturulan jüri tarafından yapılan sınavda
DOKTORA TEZİ
olarak kabul edilmiştir.

Jüri Üyeleri

Başkan : Prof. Dr. Tülay YILDIRIM

Üye : Prof. Dr. Vasıf NABİYEV

Üye : Prof. Dr. Temel KAYIKÇIOĞLU

Üye : Doç. Dr. Sedat AKLEYLEK

Üye : Dr. Öğr. Üyesi Güzin ULUTAŞ



Prof. Dr. Sadettin KORKMAZ

Enstitü Müdürü

ÖNSÖZ

Sayısal görüntüler ve videolar, sosyal amaçlı kullanımlarının dışında, adli incelemelerde suça ya da suçsuzluğa delil teşkil edecek şekilde yargı makamlarına sunulmaktadır. Günümüzde geliştirilen çok sayıda görüntü/video düzenleme yazılımıyla ilgili sayısal ortamlar üzerinde, çeşitli sahtecilik işlemleri oldukça kolay bir şekilde yapılabilir hale gelmesi sebebiyle sayısal görüntü ve videoların özellikle yargı alanında kanıt olarak kullanımından önce herhangi bir şekilde değişikliğe tabi tutulup tutulmadığının tespiti önem arz etmektedir. Bu tez çalışmasında, yeni yaklaşımlar ortaya konularak görüntü/video sahteciliklerinin tespitini gerçekleyen pasif görüntü/video doğrulama sisteminin tasarımı gerçekleştirilmiştir.

Bu çalışmada danışmanlığımı üstlenen değerli hocam Prof. Dr. Vasıf V. NABİYEV'e ilgi, alaka ve yardımlarından dolayı teşekkürü bir borç bilirim. Akademisyenlik hayatım boyunca bana her türlü desteği sağlayan hocam ve proje yürütücümüz Dr. Öğr. Üyesi Güzin ULUTAŞ'a, proje süresince bilgi ve tecrübelerinin esirgemeyen proje danışmanımız Prof. Dr. Mustafa ULUTAŞ'a, fikirlerinden yararlandığım tez jürime çok teşekkür ederim. Tez çalışmamı 115E214 kodlu 3001 projesi kapsamında destekleyen TÜBİTAK'a ayrıca teşekkür ederim. Doktora eğitimimin başlamasında ve her aşamasında yanımda olan eşime, hoşgörülerinden ötürü sevgili aileme çok teşekkür ederim.

Beste ÜSTÜBİOĞLU
Trabzon 2018

TEZ ETİK BEYANNAMESİ

Doktora Tezi olarak sunduđum “Sayısal Delillerde Pasif Doğrulama Yöntemlerine Dayalı Sahtecilik Tespiti” başlıklı bu çalışmayı baştan sona kadar danışmanım Prof. Dr. Vasıf NABİYEY’in sorumluluğunda tamamladıđımı, verileri/örnekleri kendim topladıđımı, deneyleri/analizleri ilgili laboratuvarlarda yaptıđımı/yaptırdıđımı, başka kaynaklardan aldıđım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiđimi, çalışma sürecinde bilimsel araştırma ve etik kurallara uygun olarak davrandıđımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiđimi beyan ederim.

09/04/2018

Beste ÜSTÜBİOĞLU

İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ	III
TEZ ETİK BEYANNAMESİ.....	IV
İÇİNDEKİLER.....	V
ÖZET	VIII
SUMMARY	IX
ŞEKİLLER DİZİNİ.....	XI
TABLolar DİZİNİ.....	XVII
SEMBOLLER DİZİNİ	XVIII
1. GENEL BİLGİLER.....	1
1.1. Giriş.....	1
1.1.1. Görüntü Üzerinde Yapılan Sahtecilikler	2
1.1.2. Video Üzerinde Yapılan Sahtecilikler.....	6
1.2. Tezin Kapsamı.....	10
1.3. Kopyala-Yapıştır Sahteciliği Tespiti	11
1.3.1. Blok Tabanlı Teknikler	14
1.3.1.1. Ön İşlem	14
1.3.1.2. Alt Bloklara Ayrıştırma.....	14
1.3.1.3. Özellik Çıkarma	15
1.3.1.4. Özellik Eşleştirme	24
1.3.1.5. Sahtecilik Tespiti.....	25
1.3.2. Anahtar Nokta Tabanlı Teknikler.....	25
1.3.2.1. Ölçekten Bağımsız Özellik Dönüşümü-SIFT.....	29
1.3.2.2. Hızlandırılmış Dayanıklı Özellikler-SURF	33
1.4. Video Üzerinde Yapılan Sahtecilikler.....	37
1.4.1. İkili veya Çoklu Sıkıştırma Tespiti ile Belirlenen Sahtecilikler.....	37
1.4.2. Çerçeveler Arası Yapılan Sahtecilikler	41
1.4.2.1. Çerçeve Tekrarlama Sahteciliği	41
1.4.3. Bölge Tabanlı Sahtecilikler	47
2. YAPILAN ÇALIŞMALAR	52
2.1. Görüntü Üzerinde Yapılan Çalışmalar	56
2.1.1. CLD ve Renk Momentlerine Dayalı Kopyala-Yapıştır Sahteciliği Tespiti	56

2.1.1.1.	Renk Momentleri.....	57
2.1.1.2.	Renk Düzeni Tanımlayıcısı-CLD.....	58
2.1.2.	Eşik Değerinin Otomatik Olarak Belirlendiği Kopyala-Yapıştır Sahteciliği Tespiti.....	62
2.1.2.1.	AKD Faz.....	63
2.1.2.2.	AKD Uzayında Enerji Olasılığı	64
2.1.3.	LBP-AKD ‘ye Dayalı Kopyala-Yapıştır Sahteciliği Tespiti.....	73
2.1.4.	LPQ ‘ya Dayalı Kopyala-Yapıştır Sahteciliği Tespiti.....	76
2.1.4.1.	Yerel Faz Nicemleyicisi-LPQ	76
2.1.5.	Renkli SIFT Yöntemine Dayalı Kopyala-Yapıştır Sahteciliği Tespiti.....	80
2.1.6.	LPQ ve SIFT Yöntemlerine Dayalı Anahtar Nokta Tabanlı Kopyala-Yapıştır Sahteciliğinin Tespiti.....	82
2.1.7.	Pasif Görüntü Doğrulama Yöntemlerine Dayalı Medikal Görüntülerde Sahtecilik Tespiti.....	84
2.1.7.1.	LBPROT Operatörü	84
2.2.	Video Üzerinde Yapılan Çalışmalar.....	87
2.2.1.	LDB’ye Dayalı Çerçeve Tekrarlama Sahteciliği Tespiti	87
2.2.1.1.	Yerel Fark İkiliği-LDB	88
2.2.2.	İkili Özelliklere Dayalı Çerçeve Tekrarlama\Yansıma Sahteciliği Tespiti.....	96
2.2.3.	BoW Modeline Dayalı Çerçeve Tekrarlama Sahteciliği Tespiti.....	104
2.2.3.1.	BoW Modeli	106
2.2.4.	Videolarda Bölge Tabanlı Sahtecilik Tespiti	120
2.2.4.1.	NSCT Dönüşümü	122
2.2.5.	Pasif Video Doğrulama Sistemi Yazılımı	127
3.	BULGULAR VE İRDELEME.....	128
3.1.	Veritabanı	128
3.2.	Kullanılan Ölçütler	132
3.3.	CLD ve Renk Momentlerine Dayalı Kopyala-Yapıştır Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar	134
3.4.	Eşik Değerinin Otomatik Olarak Belirlendiği Kopyala-yapıştır Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar	142
3.5.	LBP-AKD’ye Dayalı Kopyala-Yapıştır Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar	151
3.6.	LPQ’ya Dayalı Kopyala-Yapıştır Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar	155
3.7.	Renkli SIFT Yöntemine Dayalı Kopyala-Yapıştır Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar	159

3.8.	LPQ ve SIFT Yöntemlerine Dayalı Anahtar Nokta Tabanlı Kopyala-Yapıştır Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar	162
3.9.	Pasif Görüntü Doğrulama Yöntemlerine Dayalı Medikal Görüntülerde Sahtecilik Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar	167
3.10.	LDB'ye Dayalı Çerçeve Tekrarlama Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar	178
3.11.	İkili Özelliklere Dayalı Çerçeve Yansıma\Tekrarlama Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar	180
3.12.	BoW Modeline Dayalı Çerçeve Tekrarlama Sahteciliğinin Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar	184
3.13.	Videolarda Bölge Tabanlı Sahtecilik Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar	195
4.	SONUÇLAR.....	202
5.	ÖNERİLER	206
6.	KAYNAKLAR.....	208
ÖZGEÇMİŞ		

Doktora Tezi

ÖZET

SAYISAL DELİLLERDE PASİF DOĞRULAMA YÖNTEMLERİNE DAYALI
SAHTECİLİK TESPİTİ

Beste ÜSTÜBİOĞLU

Karadeniz Teknik Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı
Danışman: Prof.Dr. Vasıf NABİYEV
2018, 199 (Tez Sayfa)

Tez kapsamında görüntü/video sahteciliklerinin tespitindeki problemler irdelenmiş ve bu problemlere çözüm getirecek şekilde yeni görüntü/video sahtecilikleri tespit algoritmaları önerilmiştir. Görüntüde yapılan sahteciliklerin tespitinde görüntü bloklarından özellik çıkarımında literatürde ilk kez Renk Momentleri ile birlikte Renk Düzeni Tanımlayıcısı ve Yerel İkili Örüntüler ile birlikte Ayırık Kosinüs Dönüşümünün kullanımı, literatürdeki diğer yöntemlere kıyasla tanılama performansını oldukça yükseltmiştir. Bloklardan çıkarılan özelliklerin benzerlik araştırmasında, Benford'un genelleştirilmiş yasası ile görüntünün sıkıştırma geçmişinden yararlanılarak eşik değerinin görüntü/video içerikleri ve boyutundan bağımsız bir şekilde dinamik olarak belirlenmesi yeni bir yaklaşım olarak literatüre kazandırılmıştır. Görüntüde anahtar nokta çıkarımından önce doku çıkarılması ile düz geçişsiz bölgelerde dahi yapılan sahtecilikler tespit edilerek anahtar nokta tabanlı pasif görüntü doğrulama yöntemleri iyileştirilmiştir. Videoda çerçeveler arasında yapılan sahteciliklerin tespitinde çerçevelerden özellik çıkarımında literatürde ilk olarak Yerel Fark İkili ve Kelime Torbası yaklaşımlarından faydalanılarak literatürdeki diğer yöntemlere kıyasla yüksek başarı oranı ve düşük çalışma zamanı elde edilmiştir. Bunun dışında çerçevelerden ikili özellikler çıkarılarak Çerçeve Tekrarlama sahteciliği yanında literatürde göz önünde bulundurulmayan Çerçeve Yansıma sahteciliği de tespit edilmiştir. Tez kapsamında videoda Çerçeve Tekrarlama sahteciliği dışında video çerçevelerindeki bulanık kenarların tespiti ile bölgesel tabanlı sahtecilik de tespit edilmiştir.

Anahtar Kelimeler: Pasif Yöntemler, Kopyala-yapıştır Sahteciliği, Çerçeve Tekrarlama Sahteciliği, Videolarda Bölge Tabanlı Sahtecilik.



PhD. Thesis

SUMMARY

DIGITAL EVIDENCE FORGERY DETECTION BASED ON PASSIVE AUTHENTICATION METHODS

Beste ÜSTÜBİOĞLU

Karadeniz Technical University
The Graduate School of Natural and Applied Sciences
Computer Engineering Graduate Program
Supervisor: Prof.Dr. Vasıf NABİYEV
2018, 199 Pages

In the scope of the thesis, the problem of image/video forgery detection has been examined and the new image/video forgery detection algorithms has been proposed in order to solve these problems. The use of Colour Moments together with Colour Layout Descriptor and Discrete Cosinus Transformation together with Local Binary Pattern in feature extraction from image blocks for detection of image forgery has greatly improved detection accuracy compared to other methods in the literature. The use of image compression history with Benford's generalized law to dynamically determine the threshold value independently of the image / video content in the block similarity research have been credited with the literature as a new approach. Keypoint-based passive image authentication methods have been improved by detecting forgery in smooth regions through texture extraction prior to the key point extraction in the image. In detection the forgery intra frames, the higher success rate and the low running time were obtained in comparison with the other methods in the literature by using the Local Difference Binary and Bag of Words approaches in feature extraction from the frames. Apart from this, not only Frame Duplication forgery but also Frame Mirroring forgery which is not considered in the literature has also been detected by extracting binary features from the frames. In the thesis, besides Frame Duplication forgery on video, detection of blurred edges in video frames and region based forgery have also been identified.

Keywords: Passive methods, Copy-move forgery, Frame duplication forgery, Region based forgery.

ŞEKİLLER DİZİNİ

Sayfa No

Şekil 1.1.	Sayısal delillerde sahtecilik tespitinde kullanılan yöntemlerin sınıflandırılması	2
Şekil 1.2.	Görüntü üzerinde yapılan sahtecilik yöntemlerinin sınıflandırılması	3
Şekil 1.3.	(a)(b) Görüntü birleştirme sahteciliği ile oluşturulmuş sahte görüntüler	4
Şekil 1.4.	Görüntü rötuşlama sahteciliğine örnek	4
Şekil 1.5.	(a) Orijinal görüntü (b) Kopyala-yapıştır sahteciliği yapılmış görüntü.....	5
Şekil 1.6.	Video üzerinde yapılan sahtecilik yöntemlerinin sınıflandırılması	7
Şekil 1.7.	Sol taraftaki çerçeveler (1-4) orijinal çerçeveler, sağ taraftaki çerçeveler(1-4) top figürünün kopyala-yapıştır sahteciliği ile kapatıldığı sahte çerçeveler.....	8
Şekil 1.8.	Sol taraftaki çerçeveler (1-4) orijinal çerçeveler, Sağ taraftaki çerçeveler (1-4) başka bir videodan alınan kuğu görüntüsünün video birleştirme sahteciliği ile eklendiği sahte çerçeveler	9
Şekil 1.9.	(a)Orijinal video dizisi (b) Çerçeve ekleme (c) Çerçeve silme (d) Çerçeve tekrarlama sahtecilikleri.....	9
Şekil 1.10.	(a) Orijinal video (b) Sahte video	10
Şekil 1.11.	Tez içeriğinin genel yapısı.....	11
Şekil 1.12.	(a)(b) Orijinal görüntü (c) bölge gizleme (d) bölge tekrarlama amacıyla oluşturulmuş sahte görüntüler.....	12
Şekil 1.13.	Kopyala-yapıştır sahteciliğinde uygulanan görüntü işleme operasyonları.....	13
Şekil 1.14.	Kopyala-yapıştır sahteciliği tespit algoritmalarının genel akış diyagramı	13
Şekil 1.15.	4x4 büyüklüğünde örtüşen karesel ve çembersel bloklara ayrıştırma	15
Şekil 1.16.	[59]'da önerilen blok tabanlı yöntemin akış diyagramı.....	16
Şekil 1.17.	Zigzag Tarama	17
Şekil 1.18.	[45] çalışmasındaki karesel blok ayrıştırma	19
Şekil 1.19.	[54] çalışmasındaki çembersel blok ayrıştırma	20
Şekil 1.20.	[32] çalışmasındaki görüntü alt bloklarının bölümlenmesi	20
Şekil 1.21.	[70] çalışmasında önerilen yöntemin akış diyagramı	22
Şekil 1.22.	Ölçek uzayında DoG görüntülerinin oluşturulması [98]	31
Şekil 1.23.	DoG görüntülerinde piksel komşuluklarına bakılarak maksimum ve minimum noktaların belirlenmesi [98]	31
Şekil 1.24.	Görüntü gradyanı ve anahtar nokta tanımlayıcılar [98].....	33
Şekil 1.25.	Kısmi ve yaklaşık türevler [99]	34
Şekil 1.26.	Orijinal görüntü ve tümlev görüntüsü.....	36

Şekil 1.27. Tümler görüntü	36
Şekil 1.28. (a) Orijinal (b) Sahte video	43
Şekil 1.29. (a) Orijinal (b) Sahte video	43
Şekil 1.30. Literatürde önerilmiş Çerçeve Tekrarlama Sahteciliği Tespit Yöntemlerinin genel akış diyagramı.....	44
Şekil 2.1. Görüntüden CLD vektörünün elde edilmesi.....	59
Şekil 2.2. Önerilen yöntemin akış diyagramı.....	60
Şekil 2.3. (a) Orijinal görüntü (b) Genlik görüntüsü (c) Faz görüntüsü	64
Şekil 2.4. Önerilen Özellik Çıkarma Yöntemi Aşamaları	67
Şekil 2.5. Sıkıştırılmamış ve QF=50 faktör ile sıkıştırılmış versiyonunun JPEG katsayılarının ilk rakam dağılımları. (Log-log ölçeği kullanılmıştır).....	72
Şekil 2.6. Bir piksel için 5x5'lik bölgede LPQ etiketinin elde edilmesi.....	78
Şekil 2.7. Önerilen yöntemin akış diyagramı.....	81
Şekil 2.8. (a) Orijinal görüntü (b) Sahte görüntü (c) Doku Görüntüsü.....	83
Şekil 2.9. (a) SIFT sonucu (b) LPQ+SIFT sonucu	83
Şekil 2.10. LBP operatörü.....	85
Şekil 2.11. 8 komşuluklu R yarıçaplı çembersel komşuluklar (a) LBP(8,1) (b) LBP(8,2) (c) LBP(8,3).....	86
Şekil 2.12. Önerilen yöntemin akış diyagramı.....	86
Şekil 2.13. (a) Medikal görüntü (b) Oynanmış medikal görüntü (c) LBPROT görüntü	87
Şekil 2.14. Önerilen yöntemin akış diyagramı.....	91
Şekil 2.15. Özellik vektörünün hesaplanması	92
Şekil 2.16. Sahtecilik Uzaklığı Belirleme algoritması akış diyagramı	94
Şekil 2.17. (a) Orijinal (b) Ayna yansıması ile oluşturulmuş sahte video	98
Şekil 2.18. Önerilen yöntemin akış diyagramı.....	99
Şekil 2.19. (a) Özellik çıkarma yönteminin genel yapısı (b) Önerilen ikili özellik çıkarma yönteminin yansımadan bağımsızlığın gösterimi	101
Şekil 2.20. (a)(b)(c) Video 3, Video5 ve Video 15 için tPSNR değerinin belirlenmesi (d) tPSNR değerinin belirlenmesinde kullanılan algoritma adımları	103
Şekil 2.21. Sistemin Genel Yapısı	110
Şekil 2.22. (a) (b) Kopyalanan ve yapıştırılan çerçevelerdeki anahtar noktaları.....	111
Şekil 2.23. (a)(b) Kopyalanan ve yapıştırılan çerçevelerdeki dayanıklı anahtar noktalarının seçilmesi	114
Şekil 2.24. (a) Aday Listesi (b) FR Listesi (c) Uzaklık Histogramı (d) Güncellenmiş FR listesi	119
Şekil 2.25. Seviyelere göre sekiz yönde elde edilen görüntüler	121
Şekil 2.26. NSCT dönüşümüne bir örnek	124

Şekil 2.27. Bulanıklaştırmadan önce ve sonraki kenar tipleri yoğunlukları	126
Şekil 2.28. (a)(b)(c) Orijinal (d)(e)(f) Bulanık kenar tiplerine göre mutlak maksimum frekans katsayısı dağılımları	126
Şekil 3.1. (a) (b) (c) 32x32, 64x64 ve 96x96 boyutlarında kopyalama yapılmış sahte görüntüler (d) (e) (f) Önerilen yöntemin doğruluk oranları ve yanlış negatif değerleri	135
Şekil 3.2. (a) (b) (c) Düzgün olmayan bölgelerin kopyalanması oluşan sahte görüntüler (d) (e) (f) Önerilen yöntemin doğruluk oranları ve yanlış negatif değerleri	135
Şekil 3.3. (a) (b) (c) Çoklu kopyalama yapılmış sahte görüntüler (d) (e) (f) Önerilen yöntemin doğruluk oranları ve yanlış negatif değerleri.....	136
Şekil 3.4. Gauss Bulanıklaştırma, $w = 5 \times 5, \sigma = 1$ (b) AWGN (Beyaz Gauss Gürültüsü) (SNR = 25dB) (c) JPEG qf=65 (d) Gauss Bulanıklaştırma $w = 5 \times 5, \sigma = 3$ (e) AWGN (SNR=30dB) (f) JPEG qf=75.....	137
Şekil 3.5. 25, 30, 35, 40 dB AWGN için (a) doğruluk oranları (b) yanlış negatif değerleri	138
Şekil 3.6. Pencere büyüklüğü 3, 5 ve standart sapma $\sigma=0.5, 1$ Gauss bulanıklaştırması sonucu (a) Doğruluk oranları (b) Yanlış negatif değerleri. 138	
Şekil 3.7. Kalite faktörü q=50,70,80,90'da JPEG sıkıştırma sonucu elde edilen (a) Doğruluk oranları (b) Yanlış negatif değerleri	139
Şekil 3.8. SNR=15, 25, 30, 35, 40 dB AWGN sonucu yöntemlerin doğruluk oranları (b) Yanlış negatif değerleri.....	140
Şekil 3.9. (a) Gauss bulanıklaştırma sonucu yöntemlerin doğruluk oranları (b) Yanlış negatif değerleri.....	140
Şekil 3.10. Kalite faktörü 40, 50, 70, 80, 90 iken JPEG sıkıştırma sonucu elde edilen Doğruluk oranları (b) Yanlış negatif değerleri	141
Şekil 3.11. (a) Kalite faktörü 75 ile sıkıştırılmış görüntü (b) En iyi yeniden sıkıştırılmış beş görüntünün ilk rakam dağılımları.....	144
Şekil 3.12. (a) (b) (c) 32x32, 64x64 ve 96x96 büyüklüğünde kopyala-yapıştır yapılmış sahte görüntüler (d) (e) (f) Yöntemin görsel sonuçları	145
Şekil 3.13. (a)(b)(c) Sahte görüntüler (d)(e)(f) Yöntemin görsel sonuçları.....	146
Şekil 3.14. (a)(b)(c) Sahte görüntüler (d)(e)(f) Yöntemin görsel sonuçları.....	147
Şekil 3.15. (a) (b) Kalite faktörü QF=30 ve 70 olan görüntüler (c) (d) $w=5, \sigma = 1$ ve 3 ile Gauss bulanıklaştırması uygulanmış görüntüler (e) (f) SNR = 25 ve 30 dB'lik gürültü eklenmiş görüntüler	148
Şekil 3.16. JPEG sıkıştırma uygulandığında yöntemin [12, 14, 33, 35, 71, 73] çalışmaları ile (a) doğruluk oranına göre (b) Yanlış negatif değerlerine göre kıyaslanması	149

Şekil 3.17. Görüntülere AWGN uygulanması sonucunda önerilen yöntem ile [12, 14, 33, 35, 71, 73] çalışmalarının (a) doğruluk oranına göre (b) Yanlış negatif değerlerine göre kıyaslanması	150
Şekil 3.18. Görüntülere Gauss Bulanıklaştırma uygulandığında önerilen yöntemle [12, 14, 33, 35, 71, 73] çalışmalarının (a) doğruluk oranına göre (b) Yanlış negatif değerlerine göre karşılaştırılması.....	151
Şekil 3.19. (a) Orijinal görüntü (b) Sahte görüntü (c) Maske (d) Tespit sonucu.....	152
Şekil 3.20. (a) Orijinal görüntü (b) Sahte görüntü (c) Maske (d) Tespit sonucu.....	152
Şekil 3.21. (a) Orijinal görüntü (b) Sahte görüntü (c) Maske (d) Tespit sonucu.....	153
Şekil 3.22. Çeşitli kalite faktörlerinde elde edilen tespit sonuçları.....	153
Şekil 3.23. (a) Orijinal görüntü, (b) Sahte görüntü, (c) Maske görüntüsü, (d) Algoritmanın tespit sonucu.....	155
Şekil 3.24. (a) Orijinal görüntü(b) Sahte görüntü(c) Maske görüntüsü(d) Algoritmanın tespit sonucu	156
Şekil 3.25. Farklı bulanıklaştırma seviyelerinde elde edilen program çıktıları	157
Şekil 3.26. (a) Doğruluk oranları (b) Yanlış negatif değerleri.....	157
Şekil 3.27. Önerilen yöntemin farklı bulanıklaştırma seviyelerinde diğer yöntemlerle (a) Doğruluk oranına (b) Yanlış Negatif değerine göre kıyaslanması.....	158
Şekil 3.28. (a) SIFT sonucunda (b) Renkli SIFT sonucunda eşleşen noktalar	159
Şekil 3.29. (a) 90 derece (b)180 derece (c)270 derece döndürme sonucunda renkli SIFT ile(d) 90 derece(e) 180 derece (f) 270 derece döndürme sonucunda SIFT ile eşleşen noktalar	160
Şekil 3.30. (a) Ölçek 0.3 (b) Ölçek 0.5 (c) Ölçek 1.5 sonucunda renkli SIFT ile (d) Ölçek 0.3 (b) Ölçek 0.5 (c) Ölçek 1.5 sonucunda SIFT ile elde edilmiş eşleşen noktalar.....	161
Şekil 3.31. (a) KF=90 (b) KF=50 (c) KF=30 ile kaydedilmiş sahte görüntülerin renkli SIFT sonucunda (d) KF=90 (e) KF=50 (f) KF=30 ile kaydedilmiş görüntülerin SIFT sonucunda eşleşen noktaları	162
Şekil 3.32. (a) Orijinal Görüntü (b) Sahte görüntü (c) SIFT yöntemi sonucu (Anahtar nokta: 2 Eşleşen nokta: 0)(d) Önerilen yöntem sonucu(Anahtar nokta: 11600 Eşleşen anahtar nokta: 412).....	163
Şekil 3.33. (a) Orijinal görüntü (b) Sahte görüntü (c) SIFT yöntemi sonucu (Anahtar nokta: 2520 Eşleşen anahtar nokta: 359)(d) Önerilen yöntem sonucu(Anahtar nokta: 15922 Eşleşen anahtar nokta: 852)	164
Şekil 3.34. Orijinal görüntü (b) Sahte görüntü (c) SIFT sonucu (Anahtar nokta: 1163 Eşleşen anahtar nokta: 637)(d) Önerilen yöntem sonucu (Anahtar nokta: 5153 Eşleşen anahtar nokta: 1789).....	165
Şekil 3.35. Bulanıklaştırma sonucu DR değerleri.....	165
Şekil 3.36. Gürültü sonucu DR değerleri.....	166
Şekil 3.37. JPEG sıkıştırma sonucu DR değerleri	166

Şekil 3.38. R değerlerine göre elde edilen DR'ler	167
Şekil 3.39. (a) Medikal görüntü (b) Oynanmış medikal görüntü (c) (d) Senaryo 1 sonucu: SIFT tabanlı yöntem (Anahtar nokta: 2059 Eşleşen anahtar nokta: 43) (i) (i) Önerilen yöntem sonucu (Anahtar nokta: 3791 Eşleşen anahtar nokta: 121).....	169
Şekil 3.40. (a) Oynanmış medikal görüntü (b) (c) Senaryo 1 sonucu: SIFT tabanlı yöntem (Anahtar nokta: 470 Eşleşen anahtar nokta: 14) (d) (e) Senaryo 2 sonucu: SURF tabanlı (Anahtar nokta: 769 Eşleşen anahtar nokta: 16) (f) (g) Senaryo 3 sonucu: LBPROT+SURF tabanlı (Anahtar nokta: 2667 Eşleşen anahtar nokta: 3) (h) (i) Önerilen Yöntem (Anahtar nokta: 4868 Eşleşen anahtar nokta: 57)	170
Şekil 3.41. (a) Medikal görüntü (b) Oynanmış medikal görüntü (c) (d) Senaryo 1 sonucu: SIFT tabanlı yöntem (Anahtar nokta: 531 Eşleşen anahtar nokta: 0) (e) (f) Senaryo 2 sonucu: SURF tabanlı (Anahtar nokta: 691 Eşleşen anahtar nokta: 4) (g) (h) Senaryo 3 sonucu: LBPROT+SURF tabanlı (Anahtar nokta: 2197 Eşleşen anahtar nokta: 0) (i) (j) Önerilen yöntem (Anahtar nokta: 3958 Eşleşen anahtar nokta: 111).....	171
Şekil 3.42. (a) Medikal görüntü (b) Oynanmış medikal görüntü (c) (d) Senaryo 1 sonucu: SIT tabanlı yöntem (Anahtar nokta: 1559 Eşleşen anahtar nokta: 32) (e) (f) Senaryo 2 sonucu: SURF tabanlı (Anahtar nokta: 2325 Eşleşen anahtar nokta: 59) (g) (h) Senaryo 3 (Anahtar nokta: 5967 Eşleşen anahtar nokta: 2) (i) (j) Önerilen yöntem (Anahtar nokta: 10668 Eşleşen anahtar nokta: 260).....	172
Şekil 3.43. (a) Oynanmış medikal görüntü (b) (c) Senaryo 1 sonucu: SIFT tabanlı yöntem (Anahtar nokta: 267 Eşleşen anahtar nokta: 27) (d) (e) Senaryo 2 sonucu: SURF tabanlı (Anahtar nokta: 583 Eşleşen anahtar nokta: 50) (f) (g) Senaryo 3 sonucu: LBPROT+SURF tabanlı (Anahtar nokta: 1864 Eşleşen anahtar nokta: 26) (h) (i) Önerilen yöntem (Anahtar nokta: 2485 Eşleşen anahtar nokta: 84).....	173
Şekil 3.44. (a) Medikal görüntü (b) Oynanmış medikal görüntü (c) (d) Senaryo 1 sonucu: SIFT tabanlı yöntem (Anahtar nokta: 784 Eşleşen anahtar nokta: 130) (e) (f) Senaryo 2 sonucu: SURF tabanlı (Anahtar nokta: 1469 Eşleşen anahtar nokta: 166) (g) (h) Senaryo 3 sonucu: LBPROT+SURF tabanlı (Anahtar nokta: 3043 Eşleşen anahtar nokta: 20) (i) (j) Önerilen yöntem (Anahtar nokta: 4730 Eşleşen anahtar nokta: 194).....	174
Şekil 3.45. Ataksız tıbbi görüntüler için ortalama DR değerleri	175
Şekil 3.46. Gürültü eklenmiş medikal görüntüler için ortalama DR değerleri	176
Şekil 3.47. Bulanıklaştırma uygulanmış medikal görüntüler için ortalama DR değerleri	177
Şekil 3.48. Döndürme sonucunda medikal görüntüler için elde edilen DR değerleri	177
Şekil 3.49. Ölçekleme sonucunda medikal görüntüler için elde edilen DR değerleri	178
Şekil 3.50. Önerilen yöntemin diğer çalışmalarla (a) PR (b) RR (c) DA değerlerine göre karşılaştırılması.....	179
Şekil 3.51. Duyarlılık ve Seçicilik eğrilerine t_d değerinin etkisi	181

Şekil 3.52. (a)(b)(c) Çerçeve Tekrarlama, Çerçeve Yansıma ve hem Çerçeve Yansıma hemde Çerçeve Tekrarlama sahteciliklerinde yöntemin performans değerlendirmesi(d)(e)(f) Yöntemin diğer çalışmalarla kıyaslanması	182
Şekil 3.53. Sahtecilik işlemi sonrasında uygulanan yeniden sıkıştırma işlemi sonucunda önerilen yöntemin tespit sonuçları.....	183
Şekil 3.54. Test videoları için elde edilen histogramlar ve belirlenen ölçek değerleri.....	186
Şekil 3.55. Seçilen video için tanılama performansına ölçek değerinin etkisi	187
Şekil 3.56. Seçilen videolar için olması gereken ve hesaplanan PSNR değerleri	188
Şekil 3.57. (a)Video 2 (b) Video 5 için tanılama performansı sonuçları.....	189
Şekil 3.58. Önerilen yöntemden farklı kalite faktörlerinde elde edilen PR, RR ve DA değerleri	191
Şekil 3.59. Video içeriğine karşı önerilen yöntemin dayanıklılığı	191
Şekil 3.60. th değerinin seçilmesi	192
Şekil 3.61. Önerilen yöntemin literatürdeki benzer çalışmalarla (a) PR(b)RR(c)DA açısından kıyaslanması	194
Şekil 3.62. (a) Merkezi orijinal arka planı bulanık görüntü (b) Bulanık kenarları etiketlemiş sonuç görüntüsü	195
Şekil 3.63. Görüntü birleştirme sahteciliğine bir örnek (a) Test görüntüsü (b) Sahte bölgenin alındığı görüntü (c) Sahte görüntü (d) Tespit sonucu.....	196
Şekil 3.64. (a) Referans çerçeve (b) Bulanık kenar tespit sonucu (c) Son işlem sonrası kenar tespit sonucu	197
Şekil 3.65. (a) Sahte referans çerçeve (b) Bulanık kenar tespit sonucu (c) Son işlem sonrası kenar tespit sonucu	198
Şekil 3.66. En yakın noktaların birleştirilmesine ilişkin görsel sonuçlar	198
Şekil 3.67. Ayna Simetrik Bölgesel Tekrarlama Atağı.....	200
Şekil 3.68. Video Birleştirme atağına karşı dayanıklılık testi	201

TABLÖLAR DİZİNİ

Sayfa No

Tablo 2.1. Birinci momentlere göre sınıflandırma.....	60
Tablo 2.2. Düz ve karmaşık blokların ve JPEG sıkıştırılmış hallerinin özellik vektörleri.....	69
Tablo 2.3. Düz ve karmaşık bloklar ve sıkıştırılmış hallerinin önerilen yöntem sonucunda çıkarılan özellik vektörleri.....	70
Tablo 3.1. Test videolarının özellikleri.....	130
Tablo 3.2. Test videolarının özellikleri.....	130
Tablo 3.3. Test videoları özellikleri.....	132
Tablo 3.4. Blokların sınıflandırılmasının koşum zamanına etkisi	142
Tablo 3.5. En iyi beş uydurma için model parametreleri sonuçları	143
Tablo 3.6. Yöntemlerin JPEG sıkıştırma sonucunda karşılaştırılması.....	154
Tablo 3.7. Yöntemlerin Gauss bulanıklaştırma sonucunda karşılaştırılması	154
Tablo 3.8. Yöntemlerin Beyaz Gauss gürültüsü sonucunda karşılaştırılması	154
Tablo 3.9. Çalışma zamanlarının kıyaslanması	180
Tablo 3.10.Önerilen yöntemin diğer çalışmalarla kıyaslanması	184
Tablo 3.11.Önerilen tanımlayıcı çıkarma algoritmasının etkinlik testi	189
Tablo 3.12. Önerilen yöntemin benzer çalışmalarla çalışma zamanı açısından kıyaslanması	193
Tablo 3.13.Test veri tabanı üzerinde elde edilen sonuçlar	199

SEMBOLLER DİZİNİ

DWT	: Ayrık Dalgacık Dönüşümü (Discrete Dalgacık Transform)
AKD	: Ayrık Kosinüs Dönüşümü (Discrete Cosine Transform)
TBA	: Temel Bileşen Analizinden (Principal Component Analysis)
SVD	: Tekil Değer Ayrıştırması (Singular Value Decomposition)
FMD	: Fourier-Mellin Dönüşümü (Fourier Mellin Transform)
LPFD	: Log-Polar Fourier Dönüşümünü (Log-Polar Fourier Transform)
SNR	: Tepe Gürültü Oranını (Signal Noise Ratio)
KTBA	: Çekirdek TBA (Kernel TBA)
SIFT	: Ölçekten Bağımsız Özellik Dönüşümü (Scale Invariant Feature Transform)
SURF	: Hızlandırılmış Dayanıklı Özellikler (Speeded Up Robust Features)
ORB	: Yönlü FAST ve Döndürülmüş BRIEF (Oriented FAST and Rotated Brief)
SLIC	: Basit Doğrusal İteratif Bölütleme (Simple Linear Iterative Clustering)
DoG	: Gausslar farkı (Difference of Gaussian)
GOP	: Resimler grubu
FLD	: Fisher Doğrusal Ayrıştırıcı Analizi (Fisher's Linear Discriminant Analysis)
HD	: Histogram farkı
LBP	: Yerel İkili Örüntüler (Local Binary Patterns)
BBVD	: Blok-İçi Parlaklık Varyans Tanımlayıcısı
GMM	: Gauss Karışım Modeli (Gaussian Mixture Model)
NLF	: Gürültü seviyesi fonksiyonu
HoG	: Yönlü Meyillerin Histogramı (Histogram of Gradient)
k-SVD	: k-Tekil Değer Ayrıştırması
LBPROT	: Rotasyondan bağımsız Yerel İkili Örüntüler
LDB	: Yerel Fark İkiliğini (Local Difference Binary)
PSNR	: Tepe Sinyal Gürültü Oranı
BoW	: Sözcük Torbası
DFT	: Ayrık Fourier Dönüşümü (Discrete Fourier Transform)
PSF	: Nokta Yayılım Fonksiyonu (Point Spread Function)
DoF	: Sahtecilik uzaklığı
CT	: Contourlet dönüşümü (Contourlet Transform)

LP	: Laplacian piramidi
NSCT	: Alt Örneklenmeyen Contourlet dönüşümü (Non-subsampled Contourlet Transform)
NSPFB	: Alt örneklenmeyen piramit filtre bankası (Non-subsampled pyramid filter bank)
DFB	: Yönlü filtre bankaları (Directional Filter Bank)
SNR	: Tepe Gürültü Oranı
DR	: Tespit oranı (Detection Rate)
PR	: Duyarlılık (Precision Rate)
RR	: Seçicilik (Recall Rate)
DA	: Tanılama doğruluğu (Detection Accuracy)
SSE	: Uyum Derecesi

1. GENEL BİLGİLER

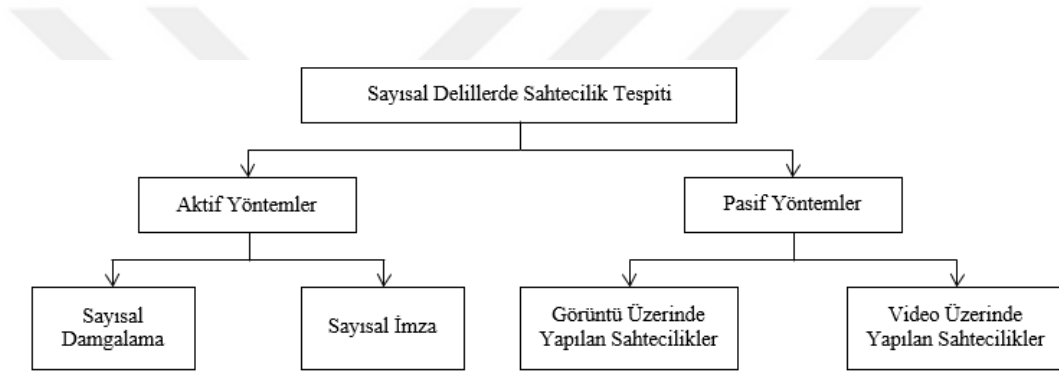
1.1. Giriş

Sayısal görüntü ve videoların, günümüzde birçok alanda kullanımı mevcuttur. Medikal bilgilerin tutulması, askeri bilgilerin depolanması ve adli süreçte delil olarak kullanımları, bu alanlara örnek olarak gösterilebilir. Özellikle adli delil olarak sayısal ortamların kullanımına son zamanlarda sıklıkla rastlanılmaktadır. Adli bir olayda, suçun aydınlatılmasında ve failinin tespitinde kullanılan en önemli mekanizma delillendirmedir. Herhangi bir suç kapsamında değerlendirilen deliller üçe ayrılmaktadır [1]. Bunlar; sayısal deliller, fiziksel öğeler ve veri nesneleridir. Bu delil tiplerinden en çok sayısal delillere ihtiyaç duyulmaktadır. Sayısal delil olarak kullanılan sayısal ortamlar, görüntü ve videolardan oluşmaktadır.

Son yıllardaki sensör ve depolama teknolojilerindeki hızlı gelişme sayesinde kameralar, fotoğraf makineleri ve hatta cep telefonlarıyla bile görüntü ve video kayıtları her an ve her yerde alınabilmektedir. Sayısal video üreticileri Premiere, Photoshop, VirtualDub, GIMP gibi ticari ve açık kaynak birçok yazılımı, görüntü/videoları iyileştirebilmek amacıyla kullanmaktadır. Görüntü/video düzenleme araçlarının kolay kullanılabilir olması, tecrübesiz kullanıcıları bile videolar üzerinde işlem yapabilir hale getirmektedir. Kullanıcılar parlaklığı arttırmak, lüminans veya krominans gürültüsünü yok etmek, titreşimi azaltmak gibi çeşitli amaçlarla görüntü/videoların görsel karakteristiklerini iyileştirmek isteyebilir. Kötü niyetli kullanıcılar ise bir nesneyi saklamak ya da tekrar etmek, gerçekleşen bir hareketi olmamış gibi göstermek veya tekrarlamak gibi çeşitli amaçlarla görüntü/video düzenleme araçlarını kullanabilmektedir. Bu sebepten ötürü sayısal delillerin hukuki yönden kabul edilebilirlikleri konusunda bazı sıkıntılar ile karşılaşılmaktadır. Bu açıdan bakıldığında delil teşkil edecek olan içeriğin mahkeme esnasında gerçek delil özelliği gösterebilmesi için, ilk alındığı andan itibaren değişmediğinin doğrulanması büyük önem arz etmektedir.

Son yıllarda görüntü/videoların doğruluğunu ve değiştirilmediğini ispatlamak için çeşitli yöntemler araştırmacılar tarafından geliştirilmektedir. Şekil 1.1.'de görüldüğü üzere literatürdeki görüntü/video sahteciliği tespit yöntemleri aktif ve pasif doğrulama yöntemleri olmak üzere ikiye ayrılmaktadır. Aktif doğrulama yöntemleri de sayısal

damgalama ve sayısal imza olmak üzere iki alt sınıfa ayrılmaktadır. Damgalama yönteminde; görüntü/video içerisine saklanan damga aracılığıyla, görüntünün bütünlüğünün korunduğu doğrulanabilmektedir. Görüntünün değiştirilmesi durumunda, yerleştirilen damga doğru bir şekilde yapılandırılmayacaktır. Fakat genel amaçlı kullanılan birçok görüntü yakalama aygıtı, görüntüyü alma ve saklama sırasında olağan JPEG başlığı dışında, görüntünün bütünlüğünü koruyacak bir damga yerleştirecek yetenekte değildir. Sayısal imza yönteminde ise imza olarak görüntü/videodan çıkarılan özellikler kullanılır. Görüntü/video doğrulama aşamasında aynı yöntem kullanılarak bu imza yeniden üretilir ve karşılaştırma yoluyla içerik doğrulaması yapılır. Yöntemin dezavantajı damgalama yöntemi ile benzerdir.

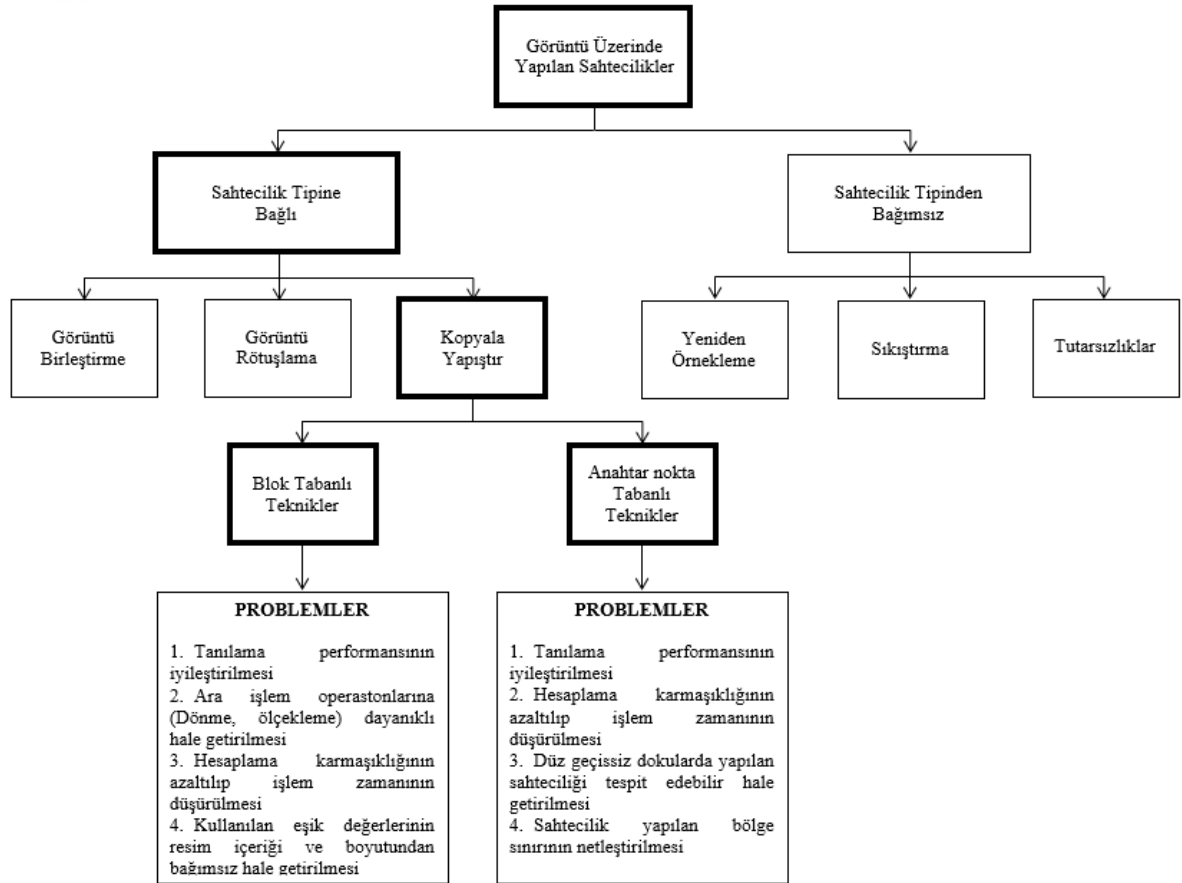


Şekil 1.1. Sayısal delillerde sahtecilik tespitinde kullanılan yöntemlerin sınıflandırılması

Sayısal ortamlarda adli inceleme yöntemleri olarak da bilinen pasif doğrulama yöntemleri, görüntü/video ile ilgili herhangi bir ön bilgi bulunmadığı ve önceden görüntünün bütünlüğünü korumak üzere hiçbir önlemin alınmadığı durumlarda dahi görüntü/video doğrulama sürecidir. Pasif yöntemler görüntü/videodan çıkarılan özelliklere göre doğrulamayı gerçekleştirmektedir. Bu durum da pasif doğrulama yöntemlerini pek çok durum için kabul edilebilir tek çözüm yolu haline getirmektedir. Pasif yöntemler de Şekil 1.1.'de görüldüğü üzere görüntü ve video üzerinde yapılan sahtecilikleri tespit eden yöntemler olmak üzere ikiye ayrılmaktadır.

1.1.1. Görüntü Üzerinde Yapılan Sahtecilikler

Günümüzde geliştirilen çok sayıda görüntü düzenleme yazılımıyla görüntü üzerinde, çeşitli sahtecilik işlemleri oldukça kolay bir şekilde yapılmaktadır. Görüntü üzerinde yapılan sahtecilikler Şekil 1.2.'de görüldüğü üzere sahtecilik tipine bağlı ve sahtecilik tipinden bağımsız yöntemler olmak üzere iki alt gruba ayrılmaktadır. Sahtecilik tipinden bağımsız yöntemler, görüntünün kasti bozulması sonrası gerçekleştirilen yeniden boyutlandırma, farklı bir JPEG kalite faktörü ile sıkıştırıp kaydetme, ortam aydınlatmasındaki tutarsızlıklar gibi çeşitli işlemlere ait izleri kullanarak doğrulamayı sağlamaktadır. Sahtecilik tipine bağlı yöntemler ise görüntü birleştirme, görüntü rötuşlama ve kopyala-yapıştır sahteciliği olarak üç alt sınıf halinde değerlendirilir.



Şekil 1.2. Görüntü üzerinde yapılan sahtecilik yöntemlerinin sınıflandırılması

Görüntü birleştirme sahteciliğinde farklı görüntülerden elde edilen bölgeler tek bir görüntü oluşturacak şekilde birleştirilmektedir. Şekil 1.3.(a) ve 1.3.(b)'de görüntü birleştirme ile oluşturulmuş sahte görüntüler verilmektedir.



Şekil 1.3. (a)(b) Görüntü birleştirme sahteciliği ile oluşturulmuş sahte görüntüler

Görüntü birleştirme sahteciliği tespit yöntemleri bölge tabanlı ve sınır tabanlı tespit yöntemleri olarak ikiye ayrılmaktadır. Sınır tabanlı yöntemler birleşme sınırlarındaki değişimleri algılamaktadırlar [2,3]. Bölge tabanlı yöntemler ise, sahtecilik tespit edebilmek için orijinal ve birleştirmiş görüntülerden oluşturulan generatif modelde tutarlılık kontrolü yapmaktadırlar [4-7].

Görüntü rötuşlama sahteciliği ise diğer görüntü sahtecilik yöntemlerine göre orijinal görüntüden en az uzaklaştıran sahtecilik çeşididir. Rötuşlama görüntü içeriğini önemli ölçüde değiştirmez ancak orijinal görüntünün bazı özelliklerini iyileştirir veya azaltır. Bu sahteciliğe birçok magazin dergisinde görüntünün belirli özelliklerini iyileştirip daha çekici hale getirmek için başvurulmaktadır. Şekil 1.4.'te yaşlı kadın yüzü üzerinde görüntü rötuşlama yapılarak kadın için daha genç bir görünüm oluşturulmuştur.



Şekil 1.4. Görüntü rötuşlama sahteciliğine örnek

Kopyala-yapıştır sahteciliği, görüntü üzerindeki bir bölgenin yine aynı görüntü üzerindeki başka bir bölge ile saklanmasına ya da çoğaltılmasına dayalı bir yöntemdir. Amaç görüntü üzerinde bir bölgeyi gizlemek ya da çoğaltmaktır. Şekil 1.5.(a)'da orijinal adli görüntüde, duvardaki telefon görüntüsü yine duvardan bir bölgenin alınıp üzerine yapıştırılması ile gizlenmiş ve Şekil 1.5.(b)'deki sahte görüntü oluşturulmuştur.

Kopyala-yapıştır sahteciliğinin gerek kolaylığı gerekse de etkinliği bu yöntemi, herhangi bir görüntü içeriğini değiştirmede kullanılan en yaygın yöntem haline getirmiştir. Kopyala-yapıştır sahteciliğinde gizlenen ya da çoğaltılan bölgelerin içeriği yine aynı görüntü içerisinden elde edildiği için, bu bölgelere ait renk paleti ve dinamik aralığı görüntünün geri kalanı ile uyumlu olacaktır. Bununla birlikte, çoğu modifikasyon işleminin ardından, gerçekleştirilen eylemi gizlemek amacıyla kötü niyetli kişi tarafından görüntü üzerine çeşitli döndürme, ölçekleme, bulanıklaştırma, sıkıştırma vb. operasyonlar uygulanabilir. Bu işlemler de kopyala-yapıştır sahteciliğinin tespitini zorlaştırmaktadır.



Şekil 1.5. (a) Orijinal görüntü (b) Kopyala-yapıştır sahteciliği yapılmış görüntü

Literatürde kopyala-yapıştır sahteciliğinin tespiti için önerilen yöntemler blok tabanlı ve anahtar nokta tabanlı teknikler olarak iki sınıfta toplanmaktadır. Blok tabanlı yöntemde görüntü örtüşen bloklara ayrılmakta ve her bir blok için özellik çıkarılmaktadır. Blokların benzerliğine bakılarak kopyala-yapıştır yapılmış bölgeler işaretlenmektedir. Anahtar nokta tabanlı yöntemlerde ise görüntü üzerinde anahtar nokta çıkarılmakta ve eşleşen noktaları içerisine alan bölgeler, görüntü üzerinde kopyala-yapıştır yapılmış bölgelere karşılık düşmektedir. Gerek blok gerekse de anahtar nokta tabanlı çalışmalarda gerçekleştirilmek istenen hedefler kısaca aşağıdaki şekilde özetlenebilir.

1. Tanılama performansının iyileştirilmesi

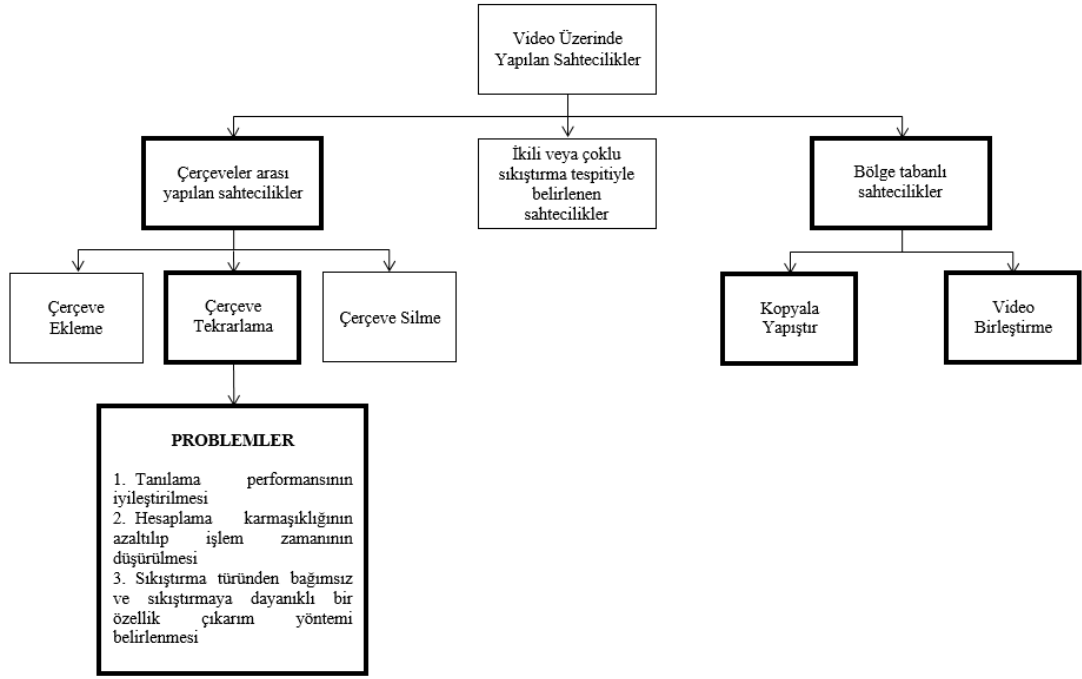
2. Ara ve son işlem operasyonlarına dayanıklı özellik çıkarma algoritmalarının tespit edilmesi
3. Algoritmanın hesaplama karmaşıklığının azaltılıp işlem zamanının düşürülmesi
4. Algoritmada kullanılan eşik değerlerinin resim içeriği ve boyutundan bağımsız hale getirilerek dinamik olarak seçilmesi
5. Seçilen yöntem anahtar nokta tabanlı ise düz geçişsiz dokularda yapılan sahteciliği de tespit edebilecek şekilde iyileştirilmesi
6. Anahtar nokta tabanlı bir yöntem seçilmişse eşleşme sonrasında sahtecilik yapılan bölge sınırının netleştirilmesi

1.1.2. Video Üzerinde Yapılan Sahtecilikler

Videoların olayı izleme imkânı sunma niteliği, adli makamlarca tek bir görüntüye göre daha güçlü bir kanıt olarak değerlendirilmektedir. Sahte video üretimi her ne kadar sahte sayısal görüntü üretiminden zor olsa da, gelişmiş video düzenleme yazılımlarıyla oldukça kolay hale gelmiştir. Video üzerinde yapılan sahtecilikler Şekil 1.6.'da görüldüğü üzere; İkili veya çoklu sıkıştırma tespiti, bölge tabanlı değişiklik tespiti ve video çerçeveleri arasında yapılan sahtecilik tespiti olmak üzere üç ana başlıkta toplanmaktadır.

İkili veya çoklu sıkıştırma tespiti sonucunda belirlenen sahteciliklerde videonun iki ya da daha fazla sayıda sıkıştırılıp sıkıştırılmadığı bilgisi araştırılmaktadır. Çünkü sıkıştırılmamış bir video üzerinde oynama yapıldıktan sonra video tekrar sıkıştırılmış formda kaydedilmektedir. Çoklu sıkıştırma da video üzerinde, videoda oynama yapıp yapılmadığının tespitini araştırarak şekilde kalanlar bırakmaktadır. Bu kalanlara bakılarak videonun sahte olup olmadığı belirlenmektedir.

Videolarda bölge tabanlı yapılan sahtecilikler kopyala-yapıştır ve video birleştirme sahtecilikleri olarak ikiye ayrılmaktadır. Kopyala-yapıştır sahteciliği, çerçeve içerisindeki bir bölgenin yine aynı çerçeve içerisindeki başka bir bölgeye yapıştırılması ya da çerçeve gruplarındaki belirli bölgelerin kopyalanarak yine aynı video içerisindeki başka bir çerçeve grubuna yapıştırılması ile oluşturulmaktadır. Bu sahtecilik de kopyalanan bölge ya da bölgeler yine aynı video içerisinden alınmaktadır.



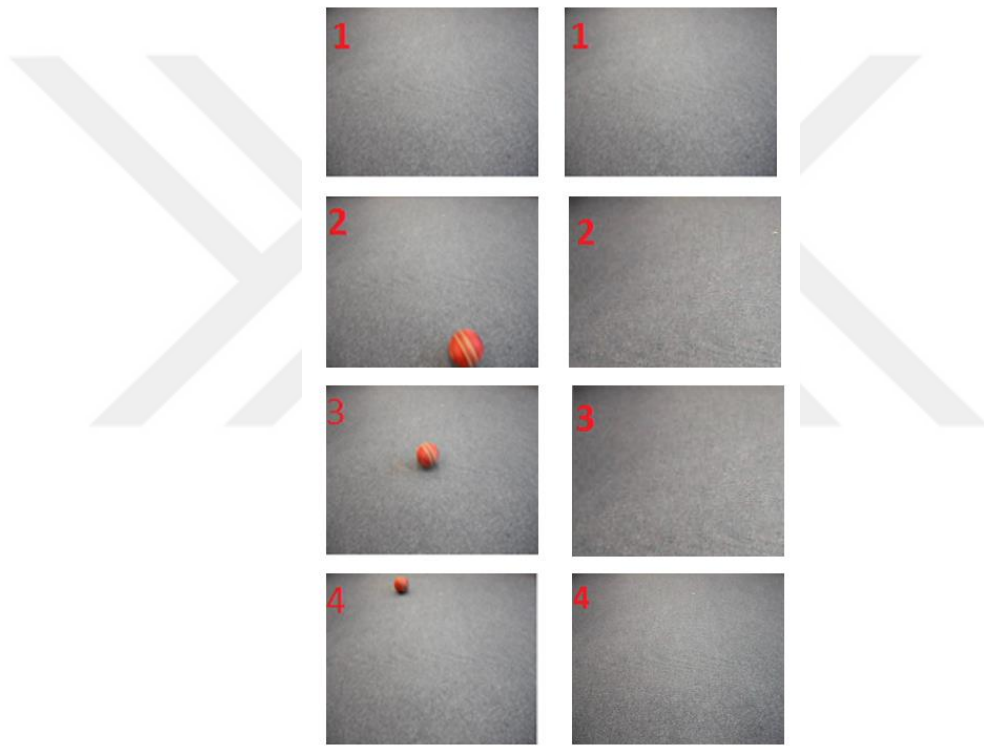
Şekil 1.6. Video üzerinde yapılan sahtecilik yöntemlerinin sınıflandırılması

Şekil 1.7.'de kopyala-yapıştır sahteciliğine örnek verilmiştir. Şekilde yer alan sol taraftaki çerçeveler orijinal videodan alınmışken, sağ taraftaki çerçeveler ise değişimin ardından oluşturulmuştur. Şekilden de gözlemlenebileceği gibi, değiştirilen çerçevelerde hareketli top figürü arka plan görüntüsünden bir bölgenin alınıp onun üzerine yapıştırılması ile kapatılmıştır.

Videolarda bölge tabanlı yapılan sahteciliklerden Video Birleştirme sahteciliğinde ise, videoda yer alan çerçeve ya da çerçeveler içerisindeki bir bölge, başka bir videodaki çerçeve ya da çerçevelerden alınan bölge/bölgeler ile yer değiştirilmektedir. Şekil 1.8.'de videolarda birleştirme sahteciliğine bir örnek verilmiştir. Örnekte sol taraftaki video çerçeveleri orijinal iken, sağ taraftaki çerçeveler kuğu görüntüsünün başka bir video çerçevesinden alınıp orijinal çerçevelere yapıştırılması sonucu elde edilen sahte çerçevelerdir. Videolarda yapılan bu bölgesel sahteciliklerin görsel izlerini kapatmak adına görüntüde olduğu gibi yapıştırılan bölgelere geometrik dönüşümler ve rötuşlamalar uygulanması videolardaki bölgesel sahtecilik tespitini oldukça zor hale getirmektedir. Literatürde bu alanda önerilmiş çalışmaların çoğunda geometrik dönüşümler göz önünde bulundurulmamıştır.

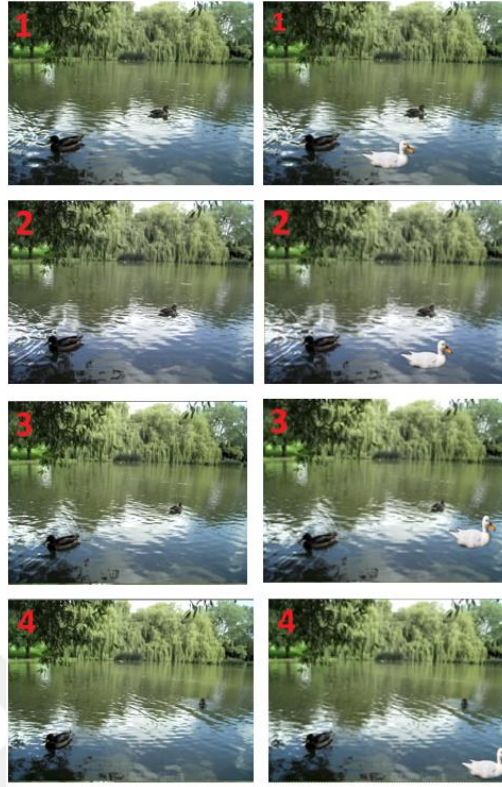
Video çerçeveleri arasında yapılan sahtecilikler; çerçeve ekleme, çerçeve silme ve çerçeve tekrarlama sahteciliği olarak üçe ayrılmaktadır. Çerçeve Ekleme sahteciliği Şekil

1.9.(b)'de görüldüğü gibi videonun herhangi bir kısmına farklı ya da aynı videodan alınan çerçeve grubunun eklenmesi ile oluşturulmaktadır. Kırmızı çerçeveler farklı videodan alınan çerçeveleri göstermektedir. Çerçeve Silme sahteciliği, videodaki belirli bir çerçeve grubunun silinmesiyle oluşturulmaktadır. Şekil 1.9.(c)'de silinen çerçeve grubu kesikli olarak verilmektedir. Çerçeve Tekrarlama sahteciliği ise video içerisindeki belirli bir çerçeve grubunun kopyalanarak yine aynı video içerisinde başka bir yere yapıştırılmasıyla oluşturulmaktadır. Şekil 1.9.(d)'den de görüldüğü üzere çerçeve tekrarlama sahteciliği oluşturulurken 2, 3 ve 4. çerçeveler kopyalanıp 7. çerçeveden sonra yapıştırılmıştır.

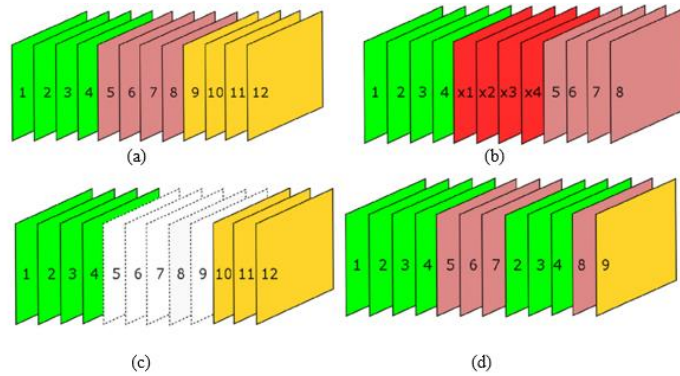


Şekil 1.7.Sol taraftaki çerçeveler (1-4) orijinal çerçeveler, sağ taraftaki çerçeveler(1-4) top figürünün kopyala-yapıştır sahteciliği ile kapatıldığı sahte çerçeveler

Video çerçeveleri arasında yapılan sahteciliklerin en yaygın ve gerçekleştirilmesi kolay olanı çerçeve tekrarlama sahteciliğidir. Çerçeve Tekrarlama sahteciliğinde, video içerisindeki bir çerçeve grubu kopyalanarak yine aynı video içerisindeki başka bir yere yapıştırılmasıyla ya bir hareket kapanmakta ya da bir hareket tekrar edilmektedir. Bu tür sahtecilik yönteminin, günümüzdeki modern video düzenleme yazılımları ile gerçekleştirilebilmesi kolay iken, insan gözü tarafından algılanabilmesi oldukça zordur.

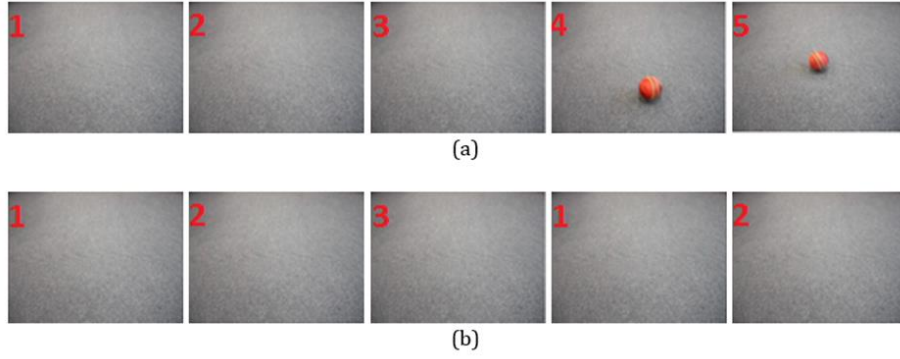


Şekil 1.8. Sol taraftaki çerçeveler (1-4) orijinal çerçeveler, Sağ taraftaki çerçeveler (1-4) başka bir videodan alınan kuğu görüntüsünün video birleştirme sahteciliği ile eklendiği sahte çerçeveler



Şekil 1.9. (a)Orijinal video dizisi (b) Çerçeve ekleme (c) Çerçeve silme (d) Çerçeve tekrarlama sahtecilikleri

Şekil 1.10.'da çerçeve tekrarlama sahteciliğine bir örnek verilmiştir. Şekil 1.10.(a) ve 1.10.(b) sırasıyla orijinal ve sahte video sıralarını göstermektedir. Orijinal videodaki ilk iki çerçeve yine aynı videodaki son iki çerçevenin üzerine yerleştirilerek top figürü yok edilmiş ve Şekil 1.10(b)'de verilmiş olan sahte video elde edilmiştir.



Şekil 1.10.(a) Orijinal video (b) Sahte video

Sahte videonun oluşturulmasının ardından kaydedilmesi esnasında yeniden kodlama işlemi gerçekleştirilmektedir. Yeniden kodlama işlemi ise video üzerinde fark edilmeyen kalanlara sebep olacağı için, kopyalanıp tekrarlanan çerçeveler birebir aynı özellik göstermeyecektir. Yeniden kodlamadan kaynaklanan bu bozulmalar ise Çerçeve Tekrarlama sahteciliğinin tespitini zorlaştırmaktadır.

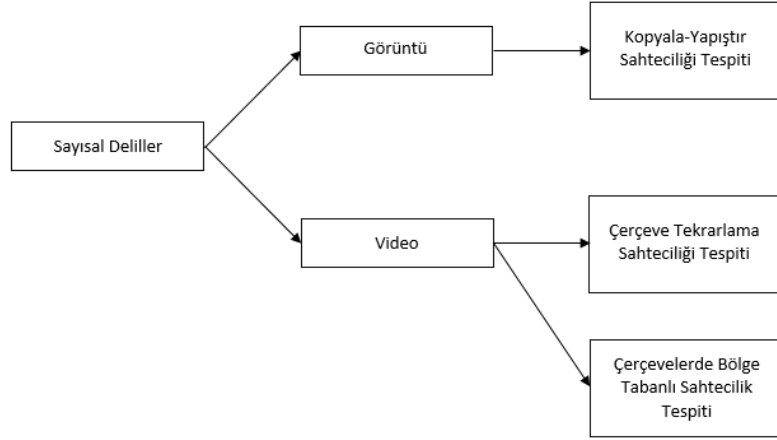
Literatürde çerçeve tekrarlama sahteciliğinin tespitine yönelik çalışmalar mevcuttur. Bu çalışmalarda önerilen yöntemlerin algoritması benzer adımlar içermektedir. Önerilen algoritmalarda öncelikle video örtüşen alt gruplara ayrıştırılmaktadır. Sonrasında bu alt gruplardan özellik çıkarılmakta ve çıkarılan özelliklere göre aday sahte çerçeve grupları belirlenmektedir. Aday çerçeve gruplarının gerçekte sahte grup olup olmadığını belirlemek için ise bu gruplar arasında çeşitli yöntemlerle uzaysal benzerliğe bakılmaktadır.

Bu çalışmalardaki gerçekleştirilmek istenen hedefler kısaca aşağıdaki şekilde özetlenebilir.

1. Tanılama performansını iyileştirmek
2. Hesaplama karmaşıklığını düşürüp işlem zamanını azaltmak
3. Sıkıştırma türünden bağımsız, sıkıştırmaya dayanıklı bir özellik çıkarım yöntemi belirleyebilmektir.

1.2. Tezin Kapsamı

Tez kapsamında sayısal delil niteliğinde olan görüntü ve videoların doğrulanmasını gerçekleştiren, pasif video doğrulama yöntemlerine dayalı bir sahtecilik tespit sisteminin tasarımı gerçekleştirilmiştir. Bu sistem görüntüde kopyala-yapıştır, videoda ise çerçeve tekrarlama ve bölge tabanlı sahtecilik tespitini gerçekleştirmektedir.



Şekil 1.11. Tez içeriğinin genel yapısı

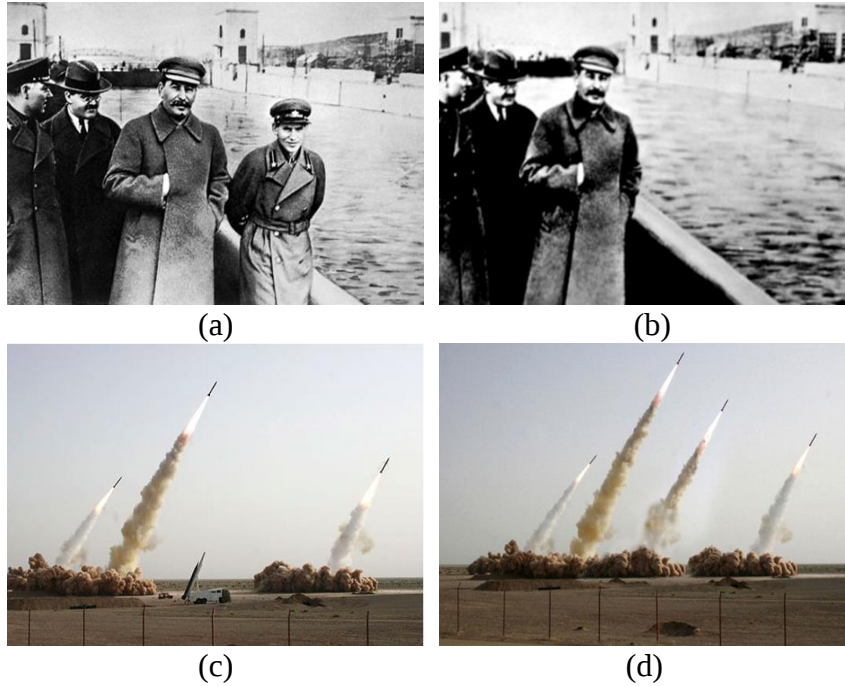
Tezin genel bilgiler bölümünde Şekil 1.11.'de verilen her bir sahtecilik tespiti için ayrı ayrı literatür taraması ve literatürde kullanılmış yöntemlerin kısaca tanımlamaları yapılacaktır. Yapılan çalışmalar bölümünde ise bu sahteciliklerin tespiti için gerçekleştirdiğimiz çalışmalar deney ve sonuçlardan bağımsız olarak özetlenecek ve görüntü/videoda yapılan sahteciliklerin tespiti alanında önerilen yöntemler açıklanacaktır. Bulgular ve irdeleme bölümünde de elde edilen sayısal ve görsel sonuçlar irdelenip yorumlanacak ve literatürde bu sahteciliklerin tespitine yönelik yapılmış olan diğer çalışmalarla karşılaştırılacaktır.

1.3. Kopyala-Yapıştır Sahteciliği Tespiti

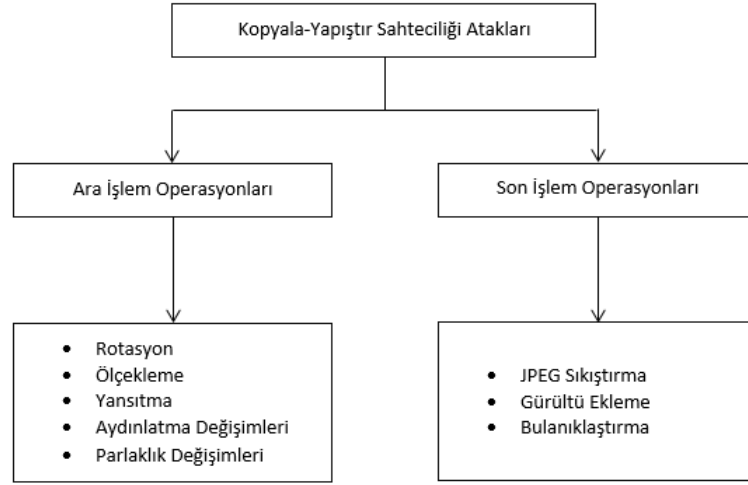
Teknolojideki ilerleyiş kullanılan sayısal ortamların günlük hayatta birçok alanda karşımıza çıkması gerçeğini beraberinde getirmiştir. Sayısal görüntüler, sosyal amaçlı kullanımlarının dışında, adli incelemelerde suça ya da suçsuzluğa delil teşkil edecek şekilde de yargı makamlarına kolaylık sağlamaktadır. Sayısal görüntü gibi ortamların özellikle yargı alanında kanıt olarak kullanımından önce herhangi bir şekilde değişikliğe tabi tutulup tutulmadığının da ispatlanması gerekmektedir. Son yıllarda görüntü modifikasyonunda en çok kullanılan tekniklerden biri kopyala-yapıştır sahteciliğidir. Kopyala-yapıştır sahteciliği görüntü üzerindeki bir bölgenin yine aynı görüntü üzerindeki başka bir bölge ile gizlenmesiyle ya da çoğaltılmasıyla oluşturulmaktadır. Bu sahtecilikteki

amaç ya bir bölgeyi gizlemek ya da çoğaltmaktır. Şekil 1.12.(a)'da orijinal görüntüdeki denize yakın adam, deniz görüntüsü ile kapatılarak 1.12.(b)'deki sahte görüntü, yine 1.12.(c)'deki füzelerden ikincisi kopyalanıp üçüncü bir füze olarak yapıştırılmasıyla 1.12.(d)'deki sahte görüntü oluşturulmuştur.

Gizlenen ya da tekrarlanan bölgelerin içeriği yine aynı görüntü içerisinde alınması sebebiyle bu bölgelere ait renk paleti ve dinamik aralığı görüntünün geri kalanı ile uyumlu olacaktır. Bununla birlikte, çoğu modifikasyon işleminin ardından, gerçekleştirilen eylemi gizlemek amacıyla kötü niyetli kişi tarafından görüntü üzerine çeşitli operasyonlar uygulanabilir. Bu operasyonlar Şekil 1.13.'te görüldüğü gibi ara işlem operasyonları ve son işlem operasyonları olmak üzere ikiye ayrılmaktadır. Ara işlem operasyonları kopyalanmış bölge ile komşuları arasındaki konumsal senkronizasyonu ve homojenliği sağlamak için kullanılmaktadır [8]. Ara işlem operasyonları döndürme, ölçekleme, yansıtma, ışıklandırma değişimleri ve renk değişimleri olabilmektedir. Bazı durumlarda da bu operasyonların iki ya da daha fazlası birlikte görüntüye uygulanabilmektedir. Beyaz gürültü ekleme, JPEG sıkıştırma ya da bulanıklaştırma gibi işlemleri içeren son işlem operasyonları da kopyala-yapıştır sürecinde kenar keskinliği gibi oluşan algılanabilir izleri gizlemek için uygulanmaktadır [9].

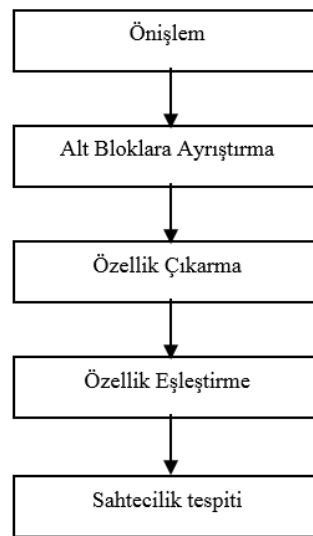


Şekil 1.12. (a)(b) Orijinal görüntü (c) bölge gizleme (d) bölge tekrarlama amacıyla oluşturulmuş sahte görüntüler



Şekil 1.13. Kopyala-yapıştır sahteciliğinde uygulanan görüntü işleme operasyonları

Literatürde Kopyala-yapıştır sahteciliğinin tespiti için yapılmış çalışmalar blok tabanlı ve anahtar nokta tabanlı olarak iki sınıfta incelenebilir. Blok tabanlı çalışmalar görüntüyü blok adı verilen sabit büyüklükte örtüşen veya örtüşmeyen karesel veya çembersel alt bölgelere ayrıştırmaktadır. Anahtar nokta tabanlı yöntemler ise görüntüyü alt bölgelere ayırmadan görüntüyü bütün olarak değerlendirmektedir. Blok tabanlı ve anahtar nokta tabanlı çalışmalar alt bölümlerde detaylandırılacaktır.



Şekil 1.14. Kopyala-yapıştır sahteciliği tespit algoritmalarının genel akış diyagramı

1.3.1. Blok Tabanlı Teknikler

Kopyala-yapıştır sahteciliğinin tespitinde kullanılan blok tabanlı yöntemlerde önerilen algoritmalar benzer yapı göstermekle beraber bu algoritmaların genel akış diyagramı Şekil 1.14.'te verildiği gibidir.

1.3.1.1. Ön İşlem

Yapılan çalışmaların büyük bir çoğunluğunda ön işlem adımı olarak renk uzayları arasında dönüşüm yapılmaktadır. Birçok yöntem [10-36] gri seviye görüntüler üzerinde çalıştığı için kırmızı, yeşil ve mavi kanallar da çalışıp sonucu birleştirmektedir. Gri seviyede çalışmak yerine bazı yöntemlerde RGB renk uzayını YCbCr renk uzayına çevirerek ya sadece Y(parlaklık) kanalı [37-40] ya da diğer renk kanallarından biri üzerinde işlem yapmaktadırlar [41]. Bunun yanında RGB renk uzayı ile Y kanalını [42-44] ve RGB ile gri seviye değerlerini kullanan çalışmalar da mevcuttur [45].

Ön işlem olarak renk uzayı dönüşümden başka Ayrık Dalgacık Dönüşümü (Discrete Dalgacık Transform-DWT) yaparak boyut azaltımı yapan çalışmalar vardır [46-50]. Literatürde kullanılan diğer ön işlem adımları olarak sahtecilik tespit zamanını indirmek için görüntüyü belli bir ölçeğe göre yeniden boyutlandırma [30, 37, 40], alçak geçiren filtre uygulama [17, 25, 37, 51], Gaussian piramidinden geçirme [50, 52-54] gibi ön işlem adımları literatürde yapılan çalışmalar arasında bulunmaktadır.

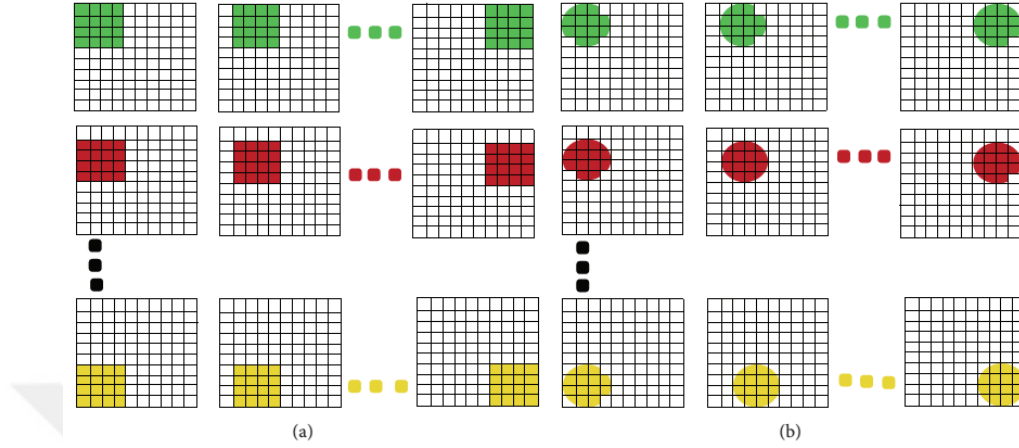
1.3.1.2. Alt Bloklara Ayırıştırma

Görüntü ön işlem aşamasından geçirildikten sonra bloklara ayrıştırılmakta ve kopyalanmış bölge tespiti için karşılaştırma blok seviyesinde yapılmaktadır. Bloklar karesel veya çembersel olabilmektedir.

Karesel Bloklara Bölme: En genel ayırıştırma şekli olup, literatürde birçok çalışmada kullanılmıştır [11-16, 30-37, 55-79]. Görüntünün en üst sol köşesinden en alt sağ köşesine kadar bxb büyüklüğünde bir pencere gezdirilerek, görüntü birbiri ile örtüşen veya örtüşmeyen alt karesel bloklara ayrıştırılmaktadır.

Çembersel Bloklara Bölme: Bazı çalışmalar görüntüyü karesel bloklara ayırıştırma yerine çembersel bloklara ayırştırmıştır [44,54].

Görüntü, Şekil 1.15.'te görüldüğü gibi örtüşen karesel veya çembersel alt bloklara ayrıştırılmaktadır.



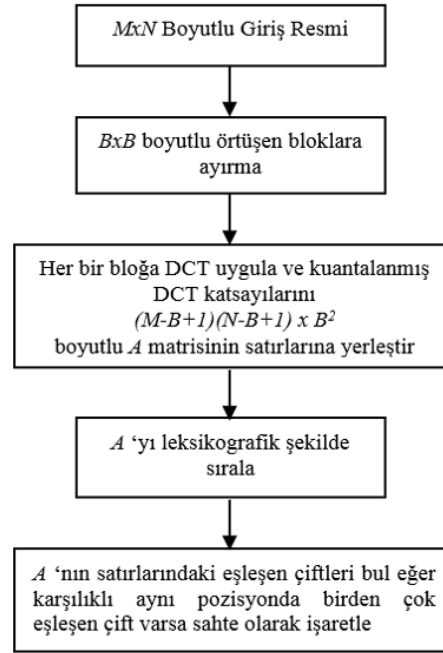
Şekil 1.15. 4x4 büyüklüğünde örtüşen karesel ve çembersel bloklara ayrıştırma

1.3.1.3. Özellik Çıkarma

Kopyala yapıştır yapılmış bölgelerin tespitindeki doğrulama oranı, tamamen görüntüye atak uygulansa bile görüntüde kopyala yapıştır uygulanmış bölgeleri temsil edecek özellikleri benzer bulabilmeye bağlıdır. Literatürde gerçekleştirilmiş olan çalışmalar birbirlerinden, görüntü bloklarını temsil etmede kullandıkları özellik vektörlerinin büyüklüğü ve çıkarılış şekli ile farklılık göstermektedir. Bu nedenle özellik çıkarımı aşamasında literatür özeti yapılırken, özellik vektörünün oluşturulmasında kullanılan algoritmaların türüne göre sınıflandırma yapılacaktır.

Ayrık Kosinüs Dönüşümü (Discrete Cosine Transform-AKD) ve Ayrık Dalgacık Dönüşümü Tabanlı Yöntemler: Kopyala-yapıştır sahteciliğinin tespitinde gerçekleştirilen ilk çalışmada, Fridrich vd. özellik vektörlerinin oluşturulmasında AKD'den faydalanmıştır [59]. Çalışmada Şekil 1.16.'da görüldüğü gibi her bir bloğun AKD'si alınmış ve blok temsiliinde kuantalanmış AKD katsayıları kullanılmıştır. Bu yöntem ile ilgili en büyük dezavantaj tekrarlanan bölgelerin küçük olması durumunda, tespit algoritmasının başarısının düşük olmasıdır.

[49] çalışması ise [59] çalışması ile benzer blok temsili kullanmış olup tek fark [49]'da görüntüye dalgacık dönüşümü uygulandıktan sonra alçak frekans bandına AKD uygulanmaktadır. Bu sayede kıyaslama yapılacak blok sayısı azaltılmıştır.

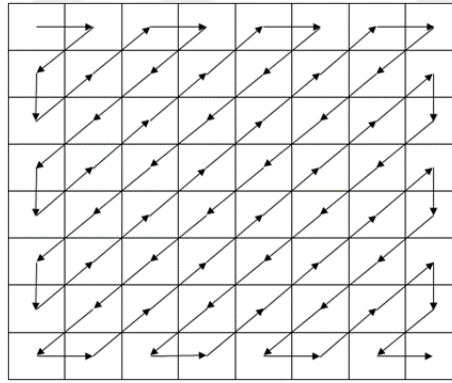


Şekil 1.16. [59]'da önerilen blok tabanlı yöntemin akış diyagramı

Bir diğer çalışmada ise kuantalanan değerler zigzag tarama ile 1-D boyutlu dizi haline getirilmiş ve elde edilen dizinin bir kısmı kesilerek bloğu temsil eden özellik vektörünün boyutu düşürülmüştür [72]. Görüntü üzerinde zigzag tarama Şekil 1.17.'de gösterildiği gibi yapılmaktadır. Yöntem JPEG sıkıştırma ve bulanıklaştırmaya karşı dayanıklıdır. [60] çalışmada ise zigzag ile oluşturulmuş, kuantalanmış AKD katsayılarının 8 tanesi seçilmiş ve yöntem bulanıklaştırmaya ve gürültüye karşı dayanıklı hale getirilmiştir. Bir diğer çalışmada da her bir blok için kuantalanmış katsayılar zigzag tarama ile 1-D boyutlu dizi haline getirildikten sonra DC katsayısını sıfıra yaklaştırmak için alçak frekans bileşenlerinde alt örnekleme yapılmış ve boyut azaltımı için Temel Bileşen Analizinden (Principal Component Analysis-TBA) yararlanılmıştır [29]. Cao vd. çalışmalarında AKD'si alınmış her bir bloğu çember olarak temsil edip çemberi dört eşit parçaya ayırmışlardır [12]. Çalışmada özellik vektörü oluştururken parçalara düşen AKD katsayılarının ortalaması alınarak özellik vektörlerinin boyutunun küçültülmesi hedeflenmiştir. Vektörler arasındaki Öklid uzaklığından ise vektörlerin benzerlik araştırması aşamasında yararlanılmıştır. Önermiş oldukları yöntem, özellik vektörünün boyutunu küçültmenin yanında, bulanıklaştırma ve gürültü ekleme işlemlerine karşı

dayanıklılık göstermektedir. [42] çalışmasında görüntü öncelikli olarak RGB uzayından YUV uzayına dönüştürülmektedir. Her bir blok için R, G, B ve Y kanalları için AKD katsayısı hesaplanmakta ve RGB kanalları için ortalama alınarak 9 özellikli özellik vektörü oluşturulmaktadır. Yöntem JPEG sıkıştırma, bulanıklaştırmaya ve gürültüye karşı dayanıklıdır. Zhao vd. çalışmalarında AKD'si alınıp kuantalanan blokları birbiri ile örtüşmeyen alt bloklara ayrıştırmakta ve her bir alt bloğun Tekil Değer Ayrıştırması (Singular Value Decomposition-SVD) ile en büyük tekil değerini hesaplamaktadır [33]. Tekil değerler birleştirilerek bloğun tamamı için özellik vektörü oluşturulmaktadır. Yöntem JPEG sıkıştırma, Beyaz Gauss gürültüsü ve bulanıklaştırmaya karşı dayanıklıdır.

Wang vd. önerdikleri yöntemde görüntü bloklarından özellik vektörlerinin üretilmesi için DWT ve AKD yöntemlerini beraber kullanmışlardır [58]. Görüntü blokları üzerinde ayrı olarak her iki yöntem uygulandıktan sonra, sonuçta üretilen katsayılar özvektörleri oluşturacak şekilde çarpılır. Bloklar arasındaki benzerlik ise özvektörlere karşılık düşen özdeğerlerin arasındaki uzaklığın ortalaması ve varyansı ile ölçülür. Önerilen yöntem JPEG sıkıştırma ve gürültü ekleme işlemlerine karşı dayanıklı olup, diğer görüntü işleme tekniklerine karşı hassastır.



Şekil 1.17. Zigzag Tarama

Log-Polar ve Fourier Tabanlı Yöntemler: Literatürdeki diğer bir alt grup ise özellik vektörlerinin oluşturulmasında Log-polar ve Fourier dönüşümünden faydalanmaktadır. Bu dönüşümleri uygulamadaki amaç, tespit algoritmalarını görüntü transformasyonlarına karşı dayanıklı hale getirmektir. Diğer bir amaç ise dönüşüm sonrası, tespit performansını etkilemeden özellik vektör boyutunu küçültmektir.

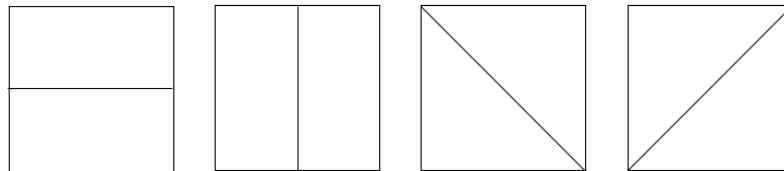
Myna vd. önerdikleri yöntemde DWT kullanmış, görüntünün alçak frekans bandını alarak boyutunu küçülttükten sonra, her seviyede görüntü bloklarını kartezyen

kordinatlardan Log- polar koordinatlara haritalanmışlardır [27]. Faz korelasyonu, görüntü blokları arasındaki benzerliğin bir ölçütü olarak kullanılmaktadır. Yöntem öteleme ve rotasyona karşı dayanıklıdır. [44] çalışmasında ise yöntemi rotasyon, yansıma ve ölçeklemeye karşı dayanıklı hale getirmek için bloklar log-polar koordinatlara haritalanmış ve açı eksenini boyunca elde edilen değerler toplanarak 1-D boyutlu özellik vektörü elde edilmiştir. Bayram vd. yapmış oldukları çalışmada, kopyalanan bölgenin yeniden ölçeklenmesi ya da rotasyona tabi tutulması durumunda dahi tanılama algoritmasının performansını iyileştirebilmek amacıyla farklı bir dönüşüm algoritması uygulamışlardır [14]. Çalışmada Fourier-Mellin Dönüşümü (Fourier Mellin Transform-FMD), görüntü bloklarını temsil etmekte kullanılır. Bunun için öncelikli olarak her bir bloğun Fourier dönüşümü alınmaktadır. Elde edilen Fourier katsayılarının genlikleri log-polar koordinatlara haritalanmakta ve blokların özellik vektörü olarak yarıçap boyunca log-polar değerlerin toplanması sonucu oluşan 1-D boyutlu dizi alınmaktadır. Bloklar arasındaki benzerliği ölçümünde ise Bloom filtrelerinden yararlanılmaktadır. Bu filtrelerin kullanımı işlemsel karmaşıklığı azaltmıştır. Yöntem %10 oranında ölçeklemeye, 20 kalite faktörüne kadar JPEG sıkıştırmaya, 10 dereceden daha az açılar için de döndürmeye karşı dayanıklıdır. Li vd. önermiş oldukları çalışmada, Bayram vd.'nin çalışmasında önerdikleri yöntemi iyileştirmeyi hedeflemişlerdir [77]. Önermiş oldukları yöntemde, Bayram vd. gibi görüntü blokları üzerinde öncelikle Fourier dönüşümü uygulanmakta, ardından üretilen genlik değerleri Log-polar koordinatlara haritalanmaktadır. Ancak bu çalışmada yarıçap boyunca toplama yerine açı boyunca toplama yapılmıştır. Vektörlerin benzerliğinin ölçülmesinde ise özüt fonksiyonlarından faydalanılmaktadır. Deneysel sonuçlarda, yöntemin 90 dereceye kadar dönmeye ve küçük ölçeklemelere dayanıklı olduğu gösterilmiştir. Fakat yöntem gürültü, bulanıklaştırma ve yüksek seviyeli ölçeklendirmelere karşı dayanıksızdır. Wu vd.'nin çalışmasında ise Bayram vd.'nin çalışmasından farklı olarak blokların temsilinde Log-Polar Fourier Dönüşümünü (Log-Polar Fourier Transform-LPFD) kullanmışlardır [38]. Çalışmada görüntü blokları dairesel olarak seçilmiş, bloklar üzerinde Log-Polar dönüşüm uygulanmış ve ardından da iki boyutlu Fourier dönüşümü gerçekleştirilmiştir. Deneysel sonuçlarda yöntemin, döndürme ve ölçeklemelere karşı dayanıklı, diğer görüntü işleme yöntemlerine karşı dayanıksız olduğu gösterilmiştir. Wu vd. gerçekleştirmiş oldukları bir diğer çalışmada ise, önceki çalışmalarındaki eksiklerin üstesinden gelebilmek amacıyla özellik vektörünün temsilinde Log-Polar Fraktal Fourier

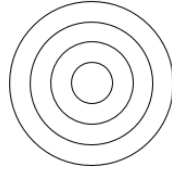
(Log-Polar Fractional Fourier Transform) dönüşümünü kullanmışlardır [73]. Yöntem ölçeklemeye ve rotasyona dayanıklıdır.

Log-polar ve Fourier dönüşümlerinden faydalanan çalışmalar, AKD tabanlı çalışmalara kıyasla daha karmaşıktır. Döndürme ve ölçeklemeye karşı dayanıklı olsalar dahi, genel olarak bulanıklaştırma, sıkıştırma, gürültü ekleme gibi görüntü işleme operasyonlarına karşı dayanıksızdırlar. En önemli dezavantajlarından biri de tespit algoritmalarının kullanmış olduğu eşik değeri sayısındaki fazlalıktır. Eşik değerlerini en uygun değerlere ayarlayabilmek için, önceden birçok testin yapılması gerekmektedir.

Doku Tabanlı Yöntemler: [45] çalışmasında bloklar için iki tür özellik çıkarılmıştır. İlk olarak renkli görüntü bloklarının R, G ve B kanallarının ortalaması özellik olarak alınmıştır. İkinci olarak ise Şekil 1.18.'de görüldüğü gibi Y kanalındaki her bir blok iki parçaya ayrılıp dört yönde bölünmüş ve her parçanın parlaklık değerinin tüm bloğun parlaklık değerine oranı özellik olarak çıkarılmıştır. Yöntem kalite faktörü 30'a kadar JPEG sıkıştırmaya, Tepe Gürültü Oranı (Signal Noise Ratio-SNR) 24dB'ye kadar gürültüye ve Gaussyan bulanıklaştırmasına karşı dayanıklıdır. [54]'te ise görüntü alt blokları çembersel olarak temsil edilmiş ve her bir çember Şekil 1.19.'da görüldüğü gibi eş merkezli dört çembere bölünmüş ve çember içerisindeki piksel parlaklık değeri ortalamaları özellik değeri olarak alınmıştır. Yöntem rotasyona, gürültüye, bulanıklaştırmaya ve 65 kalite faktörüne kadar JPEG sıkıştırmaya dayanıklıdır. Ardizzone vd. sahtecilik tespitinde istatistiksel, Tamura, kenar histogram, Haralick ve Gabor tanımlayıcıları olmak üzere beş tanımlayıcı kullanmış ve çalışma zamanına karşın doğruluk oranı açısından istatistiksel tanımlayıcıların performansının daha iyi olduğu sonucuna varmışlardır [24].



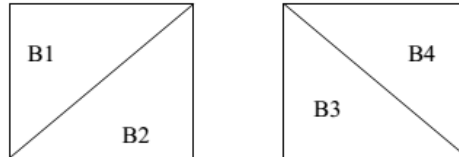
Şekil 1.18. [45] çalışmasındaki karesel blok ayrıştırma



Şekil 1.19. [54] çalışmasındaki çembersel blok ayrıştırma

[32]'de görüntünün her bir alt bloğu Şekil 1.20.'de görüldüğü gibi sağ ve sol köşegenden ayrılmak üzere dört alt bölgeye ayrılmıştır. Her bir alt bölge için hesaplama yapılmış bloğun ortalama değerine bağlı olarak 9 özellikli bir özellik vektörü oluşturulmuştur. [44] çalışmasında ise alt bloklar çembersel olarak temsil edilmiş ve çembersel bölgelerin R, G ve B kanallarının ortalaması ve piksellerin parlaklık değerlerinin entropisi alınarak 4 özellikli bir özellik vektörü oluşturulmuştur.

Gharibi vd. görüntünün alt bloklarından özellik çıkarımı yaparken Gabor özelliklerden yararlanmışlardır. Bunun için her bir bloğun Gabor özelliği bir matriste tutulmuş ve TBA ile özellik vektörlerinin boyutu indirgenmiştir. Gabordan yararlanan bir başka çalışmada ise her bir bloğun özellik çıkarımında Gabor filtresinin farklı ölçekleme faktörleri, rotasyon açıları ve frekansları kullanılmıştır [66].

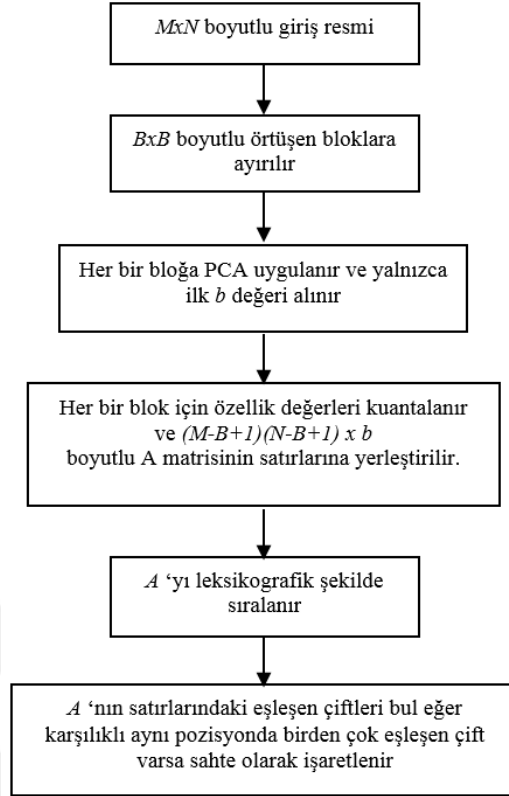


Şekil 1.20. [32] çalışmasındaki görüntü alt bloklarının bölünmesi

[43] çalışmasında ise renkli görüntünün her bir bloğu için R,G ve B renk kanallarının ortalaması, standart sapması ve çarpıklık değerleri alınarak 9 özellikli bir özellik vektörü oluşturulmuştur. Bunun yanında bloğun Y kanalının piksel değerlerinin ortalaması da bu özelliklere eklenerek toplamda 10 elemanlı bir özellik vektörü oluşturulmuştur. Yöntem gürültüye karşı dayanıklıdır. Renk bilgisi kullanan bir diğer çalışmada ise Renk Uyum vektöründen yararlanılmıştır [56]. Çalışmada her bir blok için ilk 64'ü o piksele karşılık gelen uyumlu pikselleri, son 64'ü ise uyumsuz olanları göstermek üzere toplamda 128 elemanlı özellik vektörü oluşturulmuştur. Yöntem Gauss bulanıklaştırmasına karşı dayanıklıdır.

Boyut İndirgemesi Tabanlı Yöntemler: Literatürde kopyala-yapıştır sahteciliği tespiti için yapılan çalışmalar, TBA'yı özellik çıkarma yerine görüntü bloklarından çıkarılan özellik vektörlerinin boyut küçültülmesinde kullanmaktadır. Bu amaçla özellik vektörlerinin tutulduğu özellik matrisinin kovaryans matrisi hesaplanır, kovaryans matrisin özvektörleri aracılığıyla yeni bir taban elde edilir, her bir bloğun bu temel vektörler üzerine projeksiyonu gerçekleştirilerek özellik vektörlerinin boyutu küçültülür. TBA, kopyala-yapıştır sahteciliğinin tespitinde ilk olarak Popescu ve Farid tarafından kullanılmıştır [70]. Şekil 1.21.'de görüldüğü gibi her bir bloğa TBA uygulanarak boyut azaltımı gerçekleştirilmiş ve kuantalanan değerler özellik vektörünü oluşturmuştur. Özellik vektörü boyutlarının küçülmesi, zamansal karmaşıklık açısından yöntemin üstünlük göstermesini sağlamıştır. Aynı zamanda çalışmada yöntemin gürültüye ve kayıplı JPEG sıkıştırmaya karşı dayanıklı olduğu gösterilmiştir.

Mahdian ve Saic, çalışmalarında bulanık moment değişmezlerini kullanarak Popescu'nun çalışmasını iyileştirmeyi hedeflemiştir [68]. Aynı zamanda yöntemlerinde vektörlerin benzerlik analizini yapabilmek amacıyla kd ağacı (k dimensional tree) algoritmasını kullanmışlardır.



Şekil 1.21. [70] çalışmasında önerilen yöntemin akış diyagramı

Zimba ve Xingming ise bu çalışmaya benzer olarak DWT ile TBA'yı kullanan bir algoritma önermişlerdir [80].

Çalışmalarında, görüntü boyutunu öncelikle DWT aracılığıyla boyut açısından küçültmüşler ve ardından daha küçük özellik vektörlerinin kullanımını olanaklı hale getirmişlerdir. Böylece Popescu vd.'nin çalışmasındaki hesapsal karmaşıklığı azaltmışlardır. Gharibi vd.'nin gerçekleştirmiş olduğu çalışmada ise doku bilgisi bir özellik gibi kullanılmıştır [75]. Görüntü blokları üzerinden Gabor filtresi kullanılarak doku bilgisi çıkarılmış, özellik vektörü büyüklüğü ise TBA aracılığıyla küçültülmüştür. Önermiş oldukları yöntem yalnızca JPEG sıkıştırmaya karşı dayanıklılık göstermektedir. TBA'yı tespit algoritmasında kullanan çalışmalar, küçük özellik vektörü boyutuna sahip olmakla beraber, çeşitli görüntü işleme operasyonlarının hepsine birden dayanıklılık gösterememektedirler. [65]'deki çalışmada Yerel Doğrusal Gömme yöntemine (Locally Linear Embedding) dayalı yeni bir teknik geliştirilmiştir. [31]'deki çalışmada ise her bir blok için çekirdek TBA'ya (Kernel TBA-KTBA) ve DWT'ye dayalı özellikler

çıkarılmıştır. KTBA'ya dayalı özellikler gürültüye ve oldukça düşük kalite faktöründeki sıkıştırımlara dayanıklıdır.

Tekil Değer Ayırışımına Dayalı Yöntemler: Tekil değer ayırışması, bir görüntü üzerinden geometrik anlam çıkarmak amacıyla görüntü işlemenin birçok alanında kullanılan, matrisi faktörlerine ayırma yöntemidir. Ölçekleme ve rotasyondan bağımsızlığı nedeni ile literatürde birçok kullanım alanı bulmaktadır. [62] çalışmasında her bir bloğa SVD uygulanarak özellik vektörü oluşturulmuştur. [11]'de de benzer yöntemle özellik vektörü oluşturulmuş olup tek fark ön işlem olarak DWT uygulanmış ve alçak frekans bandı üzerinden SVD gerçekleştirilmiştir. Çıkarılan özellikler kalite faktörü 70'e kadar sıkıştırma seviyesine dayanıklıdır. Xiaobing vd. de çalışmalarında [57], [13] ile aynı şekilde özellik çıkarmış ancak eşleştirme aşamasında bu iki çalışma birbirinden ayrılmaktadır. [50] çalışmasının [13]'den farkı da ön işlem olarak DWT yerine Gaussian Piramidi kullanmıştır. Kang ve Cheng ise iyileştirilmiş SVD'nin kullanımına dayalı yeni bir yöntem önermişlerdir [63]. Fakat çalışmaları çeşitli görüntü işleme operasyonlarına karşı dayanıksızdır. SVD'yi kullanan çalışmalar küçük özellik vektörü boyutuna ve dolayısıyla düşük hesapsal karmaşıklığa sahip olmakla beraber, çeşitli görüntü işleme operasyonlarına karşı dayanıklılık göstermemektedir.

Moment Tabanlı Yöntemler: Son yıllarda bazı çalışmalarda ise değişmez görüntü momentleri, kopyala-yapıştır sahteciliğinin tespitinde kullanılmaktadır. [68]'de her bir bloğun R, G ve B renk kanalları için 24 bulanık değişmez moment hesaplanarak toplamda 72 elemanlı özellik vektörü elde edilmiş ve boyut azaltımı için TBA uygulanmıştır. Çıkarılan özellikler sadece bulanıklaştırmaya değil aynı zamanda JPEG sıkıştırıma, kontrast değişikliklerine ve gürültüye karşı dayanıklıdır. [46] çalışmasının ise [68]'den tek farkı tüm görüntüdeki blokların alınmayıp sadece DWT uygulandıktan sonra alçak frekans bandındaki blokların alınışdır. Bu şekilde çalışmada işlem zamanı indirgenmiş ancak düşük kalite faktöründeki JPEG sıkıştırımlara karşı dayanıklılık sağlanamamıştır. Wang vd. çalışmalarında her bir moment için 4 bulanık momenti birleştirerek afin moment değişmezlerini hesaplamış ve bir matriste tutmuştur [19]. Yöntem afin dönüşümüne ve bulanıklaştırmaya karşı dayanıklıdır. İmamoğlu vd. ise sahtecilik tespiti için bir başka moment çeşidi olan Krawtchouk momentleri kullanmışlardır [69]. Yöntem gürültüye ve bulanıklaştırmaya karşı dayanıklıdır. Yine [8] çalışmasında ise diğer bir moment türü olan Zernike momentlerinden faydalanılmıştır. Her bir bloğun Zernike momenti çıkarılarak elde edilen vektörler genlik değerlerine göre karakterize edilmiştir. Yöntem döndürmeye ve

gürültüye karşı etkilidir. Liu vd. gerçekleştirmiş oldukları çalışmada Hu momentlerinden, kopyalanıp yapıştırılan bölgelerin tespitinde faydalanılmıştır [52]. Dairesel bloklardan Hu momentleri elde eden çalışma, yalnızca ilk dört momenti özellik vektörlerinin temsilinde kullanmıştır. Özellik vektörünün yalnızca dört adet değerle temsil ediliyor olması, yöntemin işlemsel karmaşıklığını oldukça azaltmıştır. Yöntem bulanıklaştırma, JPEG sıkıştırma, gürültü ekleme ve rotasyon işlemlerine karşı dayanıklılık göstermektedir. Ölçekleme ve öteleme işlemlerinde ise tespit algoritmasının performansı düşmektedir. Yöntemle ilgili en büyük problem, dairesele blokların dışında kalan piksellerin, algoritmanın hatalı tanıma oranını artırıyor olmasıdır. [53] çalışmasında ise Fourier momentleri ve histogram değişmez momentleri kullanılarak her bir blok için 1x7 boyutlu bir vektör elde edilmiştir. Yöntem rotasyona, ölçeklemeye ve ötelemeye karşı dayanıklıdır.

1.3.1.4. Özellik Eşleştirme

Özellik çıkarımı aşamasından sonraki aşama benzer özellikteki blokların araştırılarak aday kopyala-yapıştır çiftlerinin bulunması yani özellik eşleştirmedir. Öncelikli olarak her bir blok için çıkarılan özellik vektörleri bir matrisin satırlarında tutulur. Benzer ve aynı özellik vektörlerinin bulunmasında en basit yöntem her bir vektörü geri kalan tüm vektörlerle kıyaslamaktır. Ancak bu yöntem hesaplama zamanı açısından büyük yük getirmektedir. Bu zorluğun üstesinden gelmek için birçok yöntem benzer blokları bir araya getirmeyi hedeflemiş ve sonrasında her bir bloğu tüm bloklarla değil kendisinden sonra gelen belli sayıdaki vektörle kıyaslamıştır. Bu yöntemlerden en çok kullanılanı, özellik vektörlerinin tutulduğu matriste benzer özellikli olan vektörleri komşu duruma getiren ve benzerlik tespitini kolaylaştırmak için söz dizimsel olarak satır sıralaması yapan leksikografik (sözlük sıralaması) sıralamadır [27-33]. Leksikografik sıralamanın yanında Radix sıralama [16, 81], sıfır sayısına göre sıralama [37], kd ağacı [23, 68], koşma zamanı ve doğruluğu iyileştirmek için Leksikografik sıralama ve kd ağacının birlikte kullanımı [23], blooms filtrelerinin sayımı [14], tüm özellikler arasında maksimum varyansa sahip vektör bileşenlerine göre sıralama [24], hash değerlerinin kıyaslanması [78], blok sınıflama [82]'de kopyala-yapıştır sahteciliğinin tespitinde kullanılmaktadır.

Benzer özellik vektörleri yukarıda bahsi geçen yöntemlerle bir araya getirildikten sonra benzer özelliklerin kıyaslanmasında çeşitli benzerlik kriterleri kullanılmaktadır: Öklid uzaklığı [52-54]; dis iki vektör arasındaki Öklid uzaklığı olmak üzere $S = 1/(1 +$

dis) kriteri [34,46], Hamming uzaklığı [37], Hausdorff uzaklığı [55], mantıksal uzaklık [32], korelasyon katsayıları [19, 44], faz korelasyonu [48, 67], normalize kross spektrum [38,74], yerel hashleme [72, 8] bu kriterlerden bazılarıdır.

1.3.1.5. Sahtecilik Tespiti

Çoğunlukla tekrarlanmış bölge tespitinde tek bir benzerlik kriteri yeterli olmayacaktır. Çünkü doğal görüntüler bir veya daha çok oldukça benzer bölgeler içermekte bu da yanlış eşleşmelere sebebiyet vermektedir. Bu amaçla tekrarlanmış bölgeleri yanlış eşleşmelerden ayırmak için literatürde Aynı Afine Dönüşümü Seçimi (Same Afine Transformation Selection) [81,83], Shift Vektörü, Uzaklık Kriteri ve Alan kriteri vb. gibi farklı algoritmalar önerilmiştir.

Shift vektörü algoritmasında her eşleşen özellik vektörünün koordinatlarının arasındaki farka bakılır. Aradaki fark shift vektörü olarak adlandırılmaktadır [31-36]. Eğer bölge yapıştırılmadan önce rotasyon, ölçekleme ve öteleme operasyonlarına tabi tutulmamışsa tüm tekrarlanmış bölgeler aynı shift vektörüne sahip olacak düşüncesinden yola çıkılmaktadır.

Uzaklık kriteri algoritmasında aynı görüntü içerisindeki uzaysal olarak komşu bloklar benzer yani çiftlenmiş bölge olarak tespit edilebilir. Bu çiftleri eleyebilmek için aday çiftlerin her biri için üç ölçüt kullanılır: blokların sol üst köşe koordinatları arasındaki Öklid uzaklığı [67-70], maksimum normalize uzaklık [13,18], en yakın komşu uzaklığının en yakın ikinci komşu uzaklığına oranı [62]. Bu üç ölçütün herhangi birinden çıkan değer belirli bir eşik değerinden küçükse bloklar görüntü içerisinde uzaysal olarak yakın yerlerdedir anlamına gelir ve bu blok çifti elenir.

Alan kriteri algoritmasında ise A1 ve A2 eşleşme aşamasından sonra elde edilen iki bölge olsun. [21, 45, 54] çalışmalarında verilen algoritmalar kopyala-yapıştır sahteciliğinin yapıp yapılmadığını $v1 = \min(|A1|, |A2|)$ ve $v2 = ||A1| - |A2|| / \max(|A1|, |A2|)$ değişkenlerine dayalı olarak belirler. Eğer $v1 > \alpha M \times N \times 0.85\%$ ve $v2 < T$ ise A1 ve A2 kopyala-yapıştır yapılmış bölgeler olarak işaretlenir. Burada α ve T eşik değerleridir.

1.3.2. Anahtar Nokta Tabanlı Teknikler

Değişmez anahtar noktalarını vektör temsilinde kullanan bir diğer grup çalışmada ise, görüntü alt bloklara ayrıştırılmamaktadır. Özellikler görüntü alt bloklarından değil, görüntünün tümünden elde edilmektedir. Bu alandaki çalışmalar çoğunlukla Ölçekten Bağımsız Özellik Dönüşümü (Scale Invariant Feature Transform-SIFT) ve Hızlandırılmış Dayanıklı Özellikler (Speeded Up Robust Features-SURF) yöntemlerinden, değişmez anahtar noktalarının elde edilmesinde faydalanmaktadır. Oluşturulan özellik vektörleri ise rotasyon, öteleme ve ölçeklemeye karşı dayanıklıyken parlaklık değişimlerine ve lokal geometrik dönüşümlere ise daha az dayanıklıdır.

SIFT yöntemi ilk olarak Huang vd. tarafından kopyalanan bölgelerin tespitinde kullanılmıştır [84]. Yöntemde öncelikle giriş görüntüsüne ait anahtar noktaları elde edilmekte ve ardından bu anahtar noktaların özellik vektörleri çıkartılarak eşleştirme gerçekleştirilmektedir. Deneysel sonuçlarda kopyala-yapıştır sahteciliği uygulanan görüntünün dönme, ölçekleme, gürültü ekleme, bulanıklaştırma ve JPEG sıkıştırma ataklarına maruz kalması durumunda bile başarılı olduğu ortaya konulmuştur. Önermiş oldukları yöntemde, SIFT anahtar noktalarının eşleştirilmesi için en yakın komşuluk bulma algoritmasını kullanmışlardır. Ardizzone vd. çalışmalarında SIFT’i görüntü içerisindeki çoklu kopyalama işlemini tespit edebilecek şekilde revize etmiştir [85]. Çalışmada noktaların eşleştirilmesi yerine nesneler eşleştirilmektedir. Rapor edilen deneysel sonuçlarda tespit doğruluğu oldukça iyi verilirken yanlış doğru oranları verilmemiştir. Bir başka çalışmada Amerini vd. SIFT algoritmasını hiyerarşik kümeleme ile birlikte kullanarak daha kapsamlı bir çalışma gerçekleştirmiştir [86]. Önermiş oldukları SIFT tabanlı tespit algoritmasında aynı zamanda görüntünün bozulmasında kullanılan görüntü transformasyonunu da belirleyebilmeyi başarmışlardır. Çalışmada test görüntüsünden SIFT algoritması ile elde edilen anahtar noktalara ait tanımlayıcılar hiyerarşik kümeleme ile eşleştirilmiştir. Hatalı eşleştirmeleri yok etmek için RANSAC algoritmasından faydalanılmıştır. Önerilen yöntemin JPEG sıkıştırma, gürültü ekleme, dönme ve ölçekleme gibi ataklara karşı dayanıklı olduğu gösterilmiştir. Bu çalışmaya benzer olarak Pan ve Lyu, görüntüyü bozma esnasında kullanılan geometrik transformasyonu belirleyebilen farklı bir tespit algoritması önermiştir [87, 88]. Şüpheli görüntünün gri seviyeye çevrilmesinin ardından, anahtar noktaları SIFT aracılığıyla belirlenmekte ve bu noktalar etrafındaki görüntü özellikleri elde edilmektedir. Tespit edilen SIFT anahtar noktaları, özellik vektörlerine bağlı olarak eşleştirilir. Ardından uygulanmış olası geometrik transformasyonlar ise RANSAC kullanılarak tespit edilir. Sonuçlar yöntemin, rotasyon,

ölçekleme ve öteleme işlemleri ile gürültü ekleme, bulanıklaştırma, JPEG sıkıştırma işlemlerinin beraber kullanılmasında dahi dayanıklı olduğunu göstermiştir. Zhang vd. önermiş oldukları yöntemde, görüntüde SIFT anahtar noktalarını belirlemişler ve anahtar noktası çiftlerini bulabilmek adına onları eşleştirmişlerdir [89]. Kaynak ve hedefi ayırt edebilmek için ise, uyumlu vektörlerin yönlenimine dayalı bir oylama stratejisinden faydalanmışlardır. Deneysel sonuçlar, yöntemin geometrik manipülasyonlara karşı bile dayanıklı olduğunu göstermiştir.

Shivakumar ve Baboo, SIFT algoritmasını anahtar noktaları tespit etmekte kullanmak yerine, daha hızlı bir algoritma olma özelliği taşıyan Harris algılayıcısını kullanmaktadır [90]. Anahtar noktalar tespit edildikten sonra, SIFT, çıkartılan anahtar noktaların özellik tanımlayıcılarını üretmede kullanılır. Anahtar noktaların eşleştirilmesinde ise kd-ağaç algoritmasından yararlanır. Yöntem rotasyona ve Gauss bulanıklaştırmasına dayanıksızken, ölçeklemeye karşı dayanıklıdır.

Zhu vd. sahte bölgelerin tespiti için düşük hesaplama zamanı avantajı olan dönmeye karşı dayanıklı Yönlü FAST ve Döndürülmüş BRIEF (Oriented FAST and Rotated Brief-ORB) algoritmasını kullanmışlardır [91]. Yazarlar ORB algoritmasına ölçek bağımsız özellik kazandırmak için bu algoritmayı Gauss ölçek uzayında uygulamışlardır. Çalışmada ayrıca bölge gizleme sahteciliği de ele alınarak önerilen yöntemin başarısı vurgulanmıştır. Deneysel sonuçlarda yöntemin JPEG sıkıştırma, bulanıklaştırma, gürültü ekleme, dönme ve ölçek bağımsız olduğu gösterilmiştir. Çalışmanın, SURF ve SIFT tabanlı çalışmalara göre kullanılan performans değerlendirme metriği ile daha etkin olduğu gösterilmiştir.

Bir başka çalışmada Bo vd. SURF tanımlayıcılarını yöntemi rotasyon ve ölçekleme gibi işlemlere dayanıklı hale getirebilmek amacıyla kullanmaktadır [92]. SURF algoritması, SIFT'e göre dönüşümlere daha dayanıklı anahtar noktaların tespit edilebilmesine olanak tanımaktadır. Deneysel sonuç olarak yalnızca eşleşen anahtar noktalar görsel olarak sunulmuştur. Shivakumar ve Baboo bir diğer çalışmalarında ise SURF algoritmasının kullanımına dayanan bir tespit algoritması önermişlerdir [93]. İlk aşamada SURF yöntemi özellik çıkarım aşamasında kullanılırken, ikinci aşamada anahtar noktaları eşleştirilmekte ve ardından doğrulama işlemi gerçekleştirilmektedir. Doğrulama esnasında genel bir örüntünün oluşmasına dikkat edilmektedir. Sonuçlarda yöntemin ölçekleme, rotasyon ve Gauss bulanıklaştırmaya dayanıklı olduğu gösterilmiştir. Diğer kategorilerdeki çalışmalarla kıyaslandığında, bu alandaki çalışmaların daha fazla görüntü

işleme operasyonuna karşı dayanıklı olduğu söylenebilir. Fakat diğer yöntemlere kıyasla zamansal açıdan daha karmaşıktır. Diğer bir dezavantaj ise, kopyalanıp yapıştırılan bölgelerin düz bir yapıya sahip olması durumunda, tespit algoritmasının başarısındaki düşüştür.

Son yıllarda araştırmacılar, blok tabanlı ve anahtar nokta tabanlı teknikleri birleştirerek yeni yöntemler önermiştir. Tespit sonuçlarını iyileştirmek için ise anahtar nokta ve piksel özellikleri birleştirilmiştir. Li vd. 2015'te temelde anahtar noktalarının karşılaştırılmasına dayalı bölütleme tabanlı yeni bir yöntem önermişlerdir [94]. Yapılan bu çalışmada görüntü anahtar noktaları çıkarılmadan önce anlamsal olarak birbirinden bağımsız bölütlere ayrılır. Bu bölgeler arasındaki eşleşmenin kopyalanıp yapıştırılan bölgelere karşılık geleceği düşünülmektedir. Yöntemde eşleşme işlemi iki aşamadan oluşmaktadır. İlk aşamada anahtar noktaları ve bunlara ait özellik vektörleri çıkarılır ve eşleştirilir. Eşleşen bölgeler arasından bir transformasyon matrisi elde edilir. İkinci aşamada transformasyon matrisinin iyileştirilmesi amaçlanmıştır. Yöntem gürültü ekleme, JPEG sıkıştırma, 2-10 derecedeki dönme ve ölçekleme ataklarına karşı dayanıklı iken bulanıklaştırma atağına karşı dayanıksızdır. Pun vd. de [94]'teki bölütleme mantığından yola çıkarak yeni bir yöntem önermişlerdir [95]. İlk aşamada görüntü, karmaşıklık durumuna göre boyutları değişen bölütlere ayrılır. Daha sonra görüntünün bütününden SIFT anahtar noktaları ve özellik tanımlayıcı vektörleri elde edilmiştir. Bu çalışmada her bir bölüt bir blok olarak düşünülmüş ve her bir blokta yer alan anahtar noktalarının özellik tanımlayıcıları ile bloğa ait özellik matrisi oluşturulmuştur. Bu özellik matrislerinin eşleştirilmesi ile sahtecilik yapılan bölgeler ortaya konulmuştur. Bu algoritma dönme, ölçekleme, JPEG sıkıştırma ataklarına karşı dayanıklı iken, gürültü ekleme ve bulanıklaştırma ataklarına karşı dayanıklı değildir. Zandi ve arkadaşları ise 2016 yılında anahtar noktası tabanlı ve blok tabanlı yöntemlerin avantajlarını kullanarak kopyala-yapıştır sahteciliği tespitine uygun yeni bir anahtar noktası çıkarım yönteminin önermişlerdir [96]. Önerilen bu tekrarlamalı yöntem ile düşük çözünürlüğe sahip bölgelerde bile anahtar noktası tespit edebildiği belirtilmiştir. Ayrıca çalışmada hatalı eşleşmelerin hızlı bir şekilde giderilmesi için yeni bir filtreleme algoritması da önerilmiştir. Bunun için test görüntüsü Basit Doğrusal İteratif Bölütleme (Simple Linear İteratife Clustering-SLIC) algoritması ile bölütlenerek anahtar noktası eşleşmesinin eşleşen bölüt çiftinde en az üç tane olup olmadığı kontrol edilmiştir. Daha sonra dönüşüm matrisinin hesaplanması ile sahte bölgelerin sınırları çıkarılmaya çalışılmıştır. Yang ve arkadaşları ise

2017 yılında hibrit bir anahtar noktası tabanlı yöntem önermişlerdir [97]. Yönteme göre şüpheli görüntüden KAZE ve SIFT anahtar noktaları elde edilmektedir. Böylece SIFT algoritması ile anahtar noktası olarak tayin edilemeyen noktaları KAZE algoritması ile belirlenmiş olup daha fazla anahtar nokta çıkarımı sağlanmıştır. Hatalı eşleşmelerin giderilmesi iki aşamada gerçekleştirilmektedir. İlk aşamada, görüntünün SLIC algoritması ile bölütlenmesi gerçekleştirilmiştir. Yönteme göre bir bölütte ikiden daha az eşleşen anahtar noktanın olması durumunda, bu eşleşmeler hatalı eşleşme olarak belirlenir ve eşleşme matrisinden çıkarılır. İkinci aşamada RANSAC algoritması ile diğer hatalı eşleşmeler giderilmektedir. Daha sonra transformasyon matrisinin tahmini gerçekleştirilir.

Anahtar nokta tabanlı tekniklerde yukarıda da bahsedildiği gibi en çok kullanılan iki teknik SIFT ve SURF yöntemleridir. Bu sebeple SIFT ve SURF yöntemlerinin detaylarına aşağıda değinilecektir.

1.3.2.1. Ölçekten Bağımsız Özellik Dönüşümü-SIFT

İki imgedeki aynı bölgelerin bulunması için literatürde birçok yöntem var olmakla birlikte, imgenin boyutundan, alındığı kameranın bakış açısından, alındığı ortamın ışık koşullarından, imgedeki nesnelerin açısından bağımsız olarak eşleme işlemini gerçekleştiren Lowe tarafından önerilen Ölçekten Bağımsız Özellik Dönüşümü-SIFT yöntemidir. Yöntemde eşleme işleminden önce görüntünün özellikleri belirlenmektedir. SIFT algoritmasında temel olarak izlenen 4 adım vardır [98].

- 1-Ölçeksel uzaydaki ekstrem (uç değer) noktaların tespiti
- 2-Anahtar noktaların belirlenmesi
- 3- Anahtar noktalara yönelim atanması
- 4-Tanımlayıcı vektörler oluşturulması

Ölçeksel uzaydaki ekstrem (uç değer) noktaların tespiti: Uç değer, ölçeksel uzayda sabit olan noktalardır. Ölçek değişiminden bağımsız anahtar noktaların belirlenmesi için öncelikli olarak ölçek uzayı oluşturulmaktadır. Bir görüntünün ölçek uzayı, resmin Gauss ile katlanmasıyla oluşur. Görüntüye farklı çekirdekteki Gauss süzgecinin uygulanarak birbirinden çıkartılması ile Gausslar farkı (Difference of Gaussian - DoG) çekirdeği elde edilmektedir. DoG görüntüdeki kenar bölgelerinin ve diğer ayrıntıların ortaya çıkarılmasında kullanılmaktadır. Yüksek frekans ayrıntılarını yok etmesi sebebiyle rasgele gürültüleri yok edebilmekte ve bu özelliği ile benzer algoritmalarından ayrılmaktadır. İki

farklı ölçekteki görüntü için (1.1)'deki Gauss fonksiyonu ile DoG çekirdeği elde edilmesi (1.2), (1.3) ve (1.4)'de verildiği gibidir.

$$G_{\sigma_1} = \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp \left[-\frac{x^2+y^2}{2\sigma_1^2} \right] \quad (1.1)$$

σ_1 için;

$$L_1(x, y) = G_{\sigma_1}(x, y) * I(x, y) \quad (1.2)$$

farklı ölçek uzayı σ_2 için;

$$L_2(x, y) = G_{\sigma_2}(x, y) * I(x, y) \quad (1.3)$$

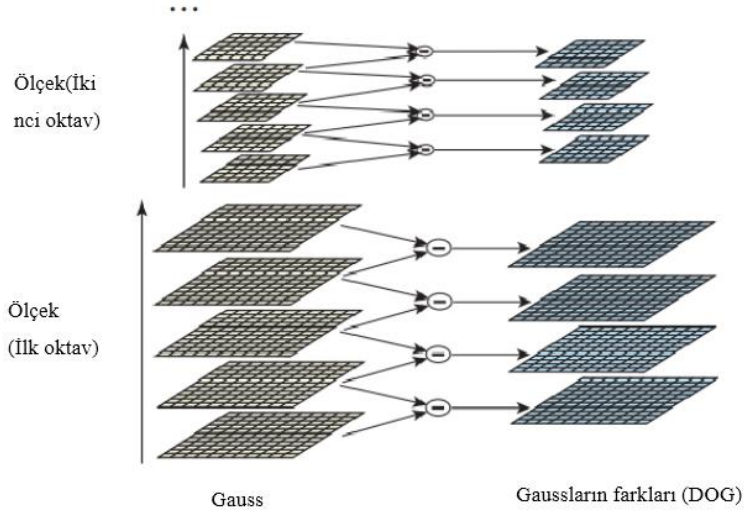
$$L_1(x, y) - L_2(x, y) = G_{\sigma_1}(x, y) * I(x, y) - G_{\sigma_2}(x, y) * I(x, y) = DoG * I(x, y) \quad (1.4)$$

Eşitliklerinde verilen σ_1 ve σ_2 arasındaki çarpım faktörü k kadardır. Lowe bu değerin 1'e yakınsamasının teoride yaklaşım hatasının sıfıra ulaşmasını sağlasa da pratikte $k\sqrt{2}$ gibi bir değer için istikrarlı uç noktaların bulunmasında nerede ise etkisinin olmadığını gözlemlemiştir. Bunun sonucunda $\sigma_1 = \sigma$ ve sonraki her ölçek için $k=\sqrt{2}$ çarpım faktörü kabul edilir ise Gauss çekirdeği ölçek uzayı standart sapma değeri her ölçek için (1.5)'te verildiği gibi;

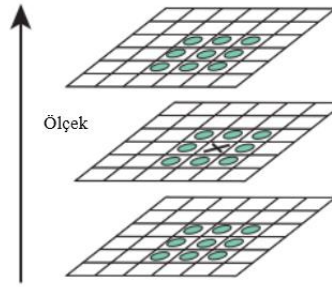
$$\begin{aligned} \sigma_1 &= \sigma \\ \sigma_2 &= k\sigma = \sqrt{2}\sigma \\ \sigma_3 &= k^2\sigma = 2\sigma \\ &\dots \\ \sigma_n &= k^{n-1} = \sqrt{2}^{n-1} \sigma \end{aligned} \quad (1.5)$$

hesaplanmaktadır. Uç değerler önce kabaca konumlandırılır, bunun için de her piksel tüm komşuları ile tek tek karşılaştırılır. Komşuluk, pikselin bulunduğu ortam ile komşu ortamlar arasında da aranır.

Ölçek uzayı belirlenen görüntünün Şekil 1.22.'de görüldüğü gibi her oktavı için DoG hesaplanır. Uç noktalarının bulunması için, her bir DoG görüntüsünün kendi ölçeğinin her pikseli çevresindeki 8 piksel ve komşu ölçeklerdeki 9'ar piksel olmak üzere toplamda 26 piksel ile karşılaştırılır. Bu pikseller Şekil 1.23.'te gösterilmiştir. Şayet piksel o bölgenin yerel maksimumu veya yerel minimumu ise aday özellik olarak belirlenir.



Şekil 1.22. Ölçek uzayında DoG görüntülerinin oluşturulması [98]



Şekil 1.23. DoG görüntülerinde piksel komşuluklarına bakılarak maksimum ve minimum noktaların belirlenmesi [98]

Anahtar noktaların belirlenmesi: Bulunan bu noktaların sayısı toplam piksel sayısından küçüktür. Fakat hala oldukça fazladır ve gerçek uç noktalar olmayabilirler. DoG operatörünün yoğun kenardan etkilenmesi ve gürültüye hassas olmasından dolayı önceki adımda elde edilen uç noktalardan düşük kontrasta sahip olanlar ve kenar bölgelerinde olanlar elenmektedir. Düşük zıtlığa sahip olanları elemek için Taylor serileri kullanılarak uç nokta hataları temizlenmeye çalışılmakta, filtrelemeler ve eşiklemeler uygulanarak ise anahtar noktaların sayısı azaltılabilmektedir. Kötü konumlanmış olanların elenmesi için, kenarların doğrultusunda ve dikine eğimin farkına bakılır. Eğer bu fark, anahtar noktanın konumu ve ölçeğindeki 2×2 Hessian matrisin en büyük özvektörünün en küçük özvektörüne oranından küçükse, anahtar nokta elenmektedir.

Özellik Noktalarına Yön Atanması: Görüntü dönmelerine karşı bağımsızlık elde edebilmek için her anahtar noktaya, yerel özelliklerine bağlı yönelim bilgisi atanmaktadır. Her anahtar noktanın, merkezinde kendisinin olduğu ve komşu pikseller ile gradyan yönelim bilgisi içerdiği bir alanda olduğu düşünülürse, görüntüde oluşacak bir dönmenin sebep olduğu çevre gradyan yönelimleri her bileşen için eşit dönmeye sahip olacaktır. Bir görüntünün gradyanı o görüntüdeki parlaklık değişiminin hangi yönde ve ne büyüklükte olduğunu ifade etmektedir. Gradyan büyüklüğü görüntüdeki parlaklık değişiminin ne kadar çabuk olduğunu verirken yönelim vektörü de hangi yönde daha hızlı değişim olduğunu göstermektedir.

Anahtar noktanın yöneliminin hesaplanabilmesi için seçilen anahtar noktanın ölçek bilgisi kullanılarak tüm hesaplamalar ölçekten bağımsız şekilde gerçekleştirilmiş olur. Her bir özellik noktası etrafında gradyan büyüklükleri $m(x,y)$ ve yönleri $\theta(x,y)$ sırasıyla eşitlik (1.6) ve (1.7) kullanılarak hesaplanmaktadır.

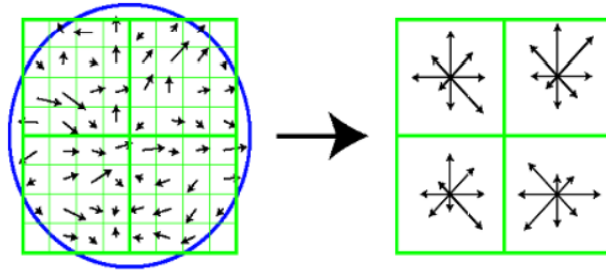
$$m(x,y) = \sqrt{((L(x+1,y) - L(x-1,y))^2 + ((L(x,y+1) - L(x,y-1))^2} \quad (1.6)$$

$$\theta(x,y) = \tan^{-1} \frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)} \quad (1.7)$$

Her biri diğerinden 10 derece açı farkına sahip olmak üzere 360 derecelik alanı kapsayan 36 adet binden oluşan yönelim histogramı, anahtar noktanın bulunduğu ölçek olan σ 'nın 1.5 katı kadar genişlikteki Gauss dairesel penceresi içindeki örnek noktaların gradyan büyüklük değerinin eklenmesi ile oluşturulur. Oluşturulan histogram bir veya birden fazla tepe noktası içermesi durumunda en yüksek tepe noktasına sahip bin yönelim açısı baskın değeri vermekle beraber, bu değerin %80'inden büyük diğer tepe noktalarına sahip yönelimler kullanılarak aynı noktada fakat farklı yönetime sahip anahtar noktalar oluşturulur. Aynı noktada farklı yönetime sahip anahtar noktaların (tüm anahtar noktaların %15'ini geçmemektedir) oluşturulması kararsızlığına sebep oluyormuş izlenimi verse de beraber eşleşme için anahtar noktanın benzersiz bir yapıya kavuşmasını sağlayarak dengeyi sağlamaktadır.

Tanımlayıcı vektörler oluşturulması: Anahtar noktalar için elde edilen tanımlayıcılar anahtar noktaların ışık değişimine karşı bağımsız olmasını sağlamaktadır. Bunun için anahtar noktanın merkezi kendisi olmak üzere bulunduğu ölçek ile orantılı mesafedeki

çevre piksellerin görüntü gradyan büyüklükleri ve yönelim açısı belirlenir. Özellik noktaları etrafında 16x16 boyutlu blok alınarak birbirleri arasında 45 derece açı farkı olan 8 yönelime sahip yönelim histogramı içeren 4x4 örnekleme alanı oluşturulur. Her okun yönelimi histogramın yönelim bilgisini, büyüklüğü ise gradyan büyüklüğünü ifade eder. Bu şekilde her biri 8 yönelimden oluşan 4x4 histogram alanına sahip anahtar nokta 4x4x8=128 boyutlu özellik vektörüne sahip olmaktadır. Şekil 1.24.'te görüntü gradyanları ve anahtar nokta tanımlayıcılar sembolize edilmiştir.



Şekil 1.24. Görüntü gradyanı ve anahtar nokta tanımlayıcılar

[98]

1.3.2.2. Hızlandırılmış Dayanıklı Özellikler-SURF

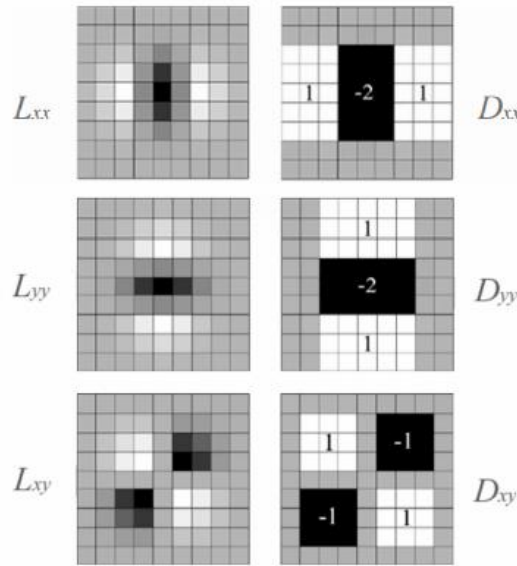
Herbert Bay tarafından 2006 önerilen özellik noktası çıkarım yöntemi olan SURF, görüntüler arasında kolayca eşleştirilebilen özellik noktalarını çıkarmak ve görüntüler arasındaki geometrik dönüşümleri hesaplamak için kullanılan bir yöntemdir [99]. SURF'un SIFT özelliklerinden en büyük farkı, görüntü konvolüsyonlarında kullandığı tümlev görüntü, Hessian matris tabanlı dedektör ve tanımlayıcı için yaklaşık 2D Haar dalgacık yanıtlarının toplamalarını kullanmasının sağladığı düşük hesaplama karmaşıklığı getiren performansdır. SURF'ün standart versiyonu yaklaşık olarak SIFT'e göre dört kat daha hızlıdır.

SURF yöntemi temelde tespit ve tanımlayıcı oluşturma olmak üzere iki aşamadan oluşmaktadır.

Tespit: Hessian matrisinin görüntüdeki blob ortaya çıkarma özelliğinden yararlanılarak görüntüdeki anahtar noktalar bulunur. Hessian matrisi simetriktir. Verilen bir I görüntüsünde, σ ölçeğindeki bir $x=(x,y)$ noktasının Hessian matrisi (1.8)'deki gibi elde edilir:

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{xy}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix} \quad (1.8)$$

Burada L_{xx} (8)'de görüldüğü gibi I görüntüsündeki (x,y) noktasının ikinci derece Gauss türevin konvolüsyon sonucu olmakla beraber L_{xy} ve L_{yy} 'de benzer şekilde (x,y) noktasının sırasıyla xy ve y yönündeki ikinci derecede türevlerinin konvolüsyonlarıdır. Şekil 1.25.'te verilen kutu filtreleri kullanılarak Gaussun ikinci türevi yaklaşık olarak hesaplanmaktadır. Şekilde gri bölgeler sıfıra eşittir. Bu kutu filtreleri en düşük ölçek olan $\sigma=1.2$ ölçeğindeki Gauss ikinci dereceden türevlerin yakınsamasına karşılık gelen 9x9 boyutundaki filtrelerdir.



Şekil 1.25. Kısmi ve yaklaşık türevler [99]

Hessian matrisinin determinanı kullanılarak anahtar noktalar çıkarılmaktadır. Hessian matrisinin determinanı (1.9)'da verildiği gibi hesaplanmaktadır.

$$\text{Det}(H_{\text{yaklaşık}}) = D_{xx}D_{yy} - (wD_{xy})^2 \quad (1.9)$$

Burada w değeri Gauss çekirdekleri arasındaki enerji dönüşümü ile hesaplanır, bu değer Bay ve ark [99] tarafından 0.9 olarak alınmıştır.

SURF yönteminde (x,y) noktasının kutu filtrelerinde karşılık gelen alt dikdörtgensel bölge ile konvüasyonu yerine tümlev görüntüler kullanılarak işlem zamanı oldukça düşürülmektedir. Tümlev görüntüler her nokta için kendisi ile başlangıç noktasının arasında kalan dikdörtgen bloğun alanı içerisindeki bütün piksellerin toplamıdır. Tümlev görüntüler ile verilen bir dikdörtgen alanı hızlı bir şekilde hesaplanır [99]. Tümlev görüntüler hesaplanırken (1.10)'da belirtildiği gibi herhangi bir x ve y noktası için kendisinden önce gelen bütün piksellerin değerlerine toplam işlemi uygulanır. X ve y tamamlayıcı görüntü değeri hesaplanacak pikseli ifade etmek üzere (0,0) noktasından başlayarak bütün (i,j) ikilileri x ve y noktalarına kadar toplanarak (x,y) noktasının değeri olarak atanır. Böylece bütün görüntü için x ve y değiştirilerek tümlev görüntü hesaplanır.

$$I_{\Sigma(x,y)} = \sum_{i=0}^{x-1} \sum_{j=0}^{y-1} I(i,j) \quad (1.10)$$

Şekil 1.26.'da örnek olarak 10x10'luk bir görüntü için tümlev görüntüsü verilmiştir. Bu işlem her noktaya uygulandığı için her nokta kendisinden önceki yoğunluk bilgisini içerir. Şekil 1.27.'de görüldüğü gibi tümlev görüntü kullanılarak resimde S ile belirtilen alanın içindeki piksellerin toplamı (1.11)'de verilen denklem kullanılarak hesaplanabilmektedir. Denklemde görüldüğü gibi A, B ve C, D değerleri kullanılarak aradaki karenin değeri hesaplanması toplamda üç adımda gerçekleştirilerek çok hızlı bir şekilde elde edilir. Bu da boyuttan bağımsız alan yoğunluk hesaplaması yapılmasına imkân sağlar. SURF yönteminde kutu filtrelerin hesaplanmasında konvolüsyon işlemi yerine tümlev görüntü yöntemi kullanılarak daha hızlı hesaplama yapılmaktadır.

$$S = (A + D) - (B + C) \quad (1.11)$$

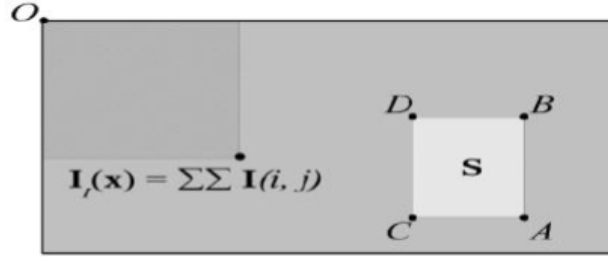
SURF yöntemi 9x9, 15x15, 21x21 ve 27x27 boyutundaki kutu filtrelerini kullanmaktadır. Değişik boyutlardaki kutu süzgeçleri tümlev görüntülere uygulanarak ölçek uzay yapısı oluşturulmaktadır. Farklı boyutlardaki kutu filtreleri yaklaşık olarak farklı ölçek değerlerindeki ikinci derece Gauss türevlerine karşılık gelmektedir. Örneğin 27x27 boyutundaki filtre için yeni ölçek değeri $(27/9)\sigma = 3\sigma = 3 \times 1.2 = 3.6$ olarak hesaplanmaktadır. Bir noktanın anahtar nokta olup olmadığı kutu filtrelerden elde edilen ölçeklerden yararlanılarak bulunmaktadır. Ardışık 3 ölçekten 3x3'lük alanlar seçilir.

Toplamda $3 \times 3 \times 3 = 27$ tane piksel arasında en yüksek gradyan değerine sahip piksel anahtar nokta olarak kabul edilir.

25	125	250	0	111	121	121	121	121	121
121	121	250	0	111	121	123	1	1	1
0	0	11	11	0	121	123	1	1	1
25	125	250	0	111	121	123	1	1	1
25	125	250	90	111	111	111	111	1	1
25	125	250	11	111	21	21	111	111	111
25	11	250	0	111	0	1	111	50	50
25	125	11	90	111	0	1	111	50	50
25	125	255	0	111	0	1	111	50	50
25	125	250	0	111	0	1	111	50	50

25	150	400	400	511	632	753	874	995	1116
146	392	892	892	1114	1356	1600	1722	1844	1966
146	392	903	914	1136	1499	1866	1989	2112	2235
171	542	1303	1314	1647	2131	2621	2745	2869	2993
196	692	1703	1804	2248	2843	3444	3679	3804	3929
221	842	2103	2215	2770	3386	4008	4354	4590	4826
246	878	2389	2501	3167	3783	4406	4863	5149	5435
271	1028	2550	2752	3529	4145	4769	5337	5673	6009
296	1178	2955	3157	4045	4661	5286	5965	6351	6737
321	1328	3355	3557	4556	5172	5798	6588	7024	7460

Şekil 1.26. Orijinal görüntü ve tümlev görüntüsü



Şekil 1.27 Tümlev görüntü

Tanımlayıcı oluşturma: SURF algoritmasında anahtar nokta etrafında çember şeklinde bir alan seçilir ve bu alana Haar dalgacık filtreleri uygulanarak tanımlayıcılar elde edilmektedir [99]. Haar dalgacık dönüşümünün seçilmesinin temel nedeni; parlaklığa karşı duyarsızdır ve tümlev görüntü ile hızlı bir şekilde hesaplanır. Bunun için anahtar nokta etrafında 20σ büyüklüğünde bir alan belirlenir, yani üzerinde bulunduğumuz ölçeği elde ederken kullandığımız σ 'nın 20 katı büyüklüğünde bir alan ve bu alan 5σ olacak şekilde 4×4 'lük karelere bölünür. Bu 4×4 'lük alanlara Haar dalgacık filtresi yatay ve dikey şekilde uygulanıp, x ve y yönündeki türevler dx ve dy hesaplanmaktadır (filtre boyutu $2s$ 'dir). Geometrik bozulmalara ve konum hatalarına karşı dayanıklılığı artırmak için anahtar noktası merkez olacak şekilde dx ve dy yanıtları öncelikle $\sigma = 3.3s$ parametresi ile Gauss ağırlıklandırılması yapılır. Daha sonra her alt alan için elde edilen dx ve dy değerleri toplanır ve bu toplamalar tanımlayıcı vektörün ilk kısmını oluştururlar. Ayrıca tanımlayıcının kutupsal yoğunluk değişimleri hakkında bilgi de tutması için bu yanıtların

mutlak değerlerinin ($|dx|$ ve $|dy|$) toplamaları da elde edilir. Böylece her alt bölge dört boyutlu tanımlayıcı vektöre sahip olmuş olur. $v = (\sum dx, \sum dy, \sum |dx|, \sum |dy|)$. Her 4×4 boyutlu alt vektör için bu dört boyutlu vektör çıkartılır. Dolayısıyla $4 \times (4 \times 4) = 64$ boyutlu tanımlayıcı vektör oluşturulmuş olur.

1.4. Video Üzerinde Yapılan Sahtecilikler

Sayısal video üreticileri Premiere, Photoshop, VirtualDub gibi ticari ve açık kaynak birçok yazılımı, videoları iyileştirebilmek amacıyla kullanmaktadır. Video düzenleme araçlarının kolay kullanılabilir olması, tecrübesiz kullanıcıları bile videolar üzerinde işlem yapabilir hale getirmektedir. Kullanıcılar parlaklığı artırmak, lüminans veya krominans gürültüsünü yok etmek, titreşimi azaltmak gibi çeşitli amaçlarla videoların görsel karakteristiklerini iyileştirmek isteyebilir. Kötü niyetli kullanıcılar ise bir nesneyi saklamak, gerçekleşen bir hareketi olmamış gibi göstermek veya videodaki bazı nesneleri tekrarlamak gibi çeşitli amaçlarla video düzenleme araçlarını kullanabilmektedir. Son yıllarda videoların doğruluğunu ve değiştirilmediğini ispatlamak için çeşitli yöntemler araştırmacılar tarafından geliştirilmektedir.

Video doğrulama alanında kullanılan aktif yöntemlerden video damgalama ve sayısal imza yöntemlerinin ek bir donanım/yazılım gerektirmesi pratik anlamdaki kullanımlarını zorlaştırmaktadır. Bu nedenle video doğrulama alanında son yıllarda pasif yöntemlerin kullanımı popülerlik kazanmıştır. Pasif yöntemler ek bir bilgiye ihtiyaç duymadan, videodan çıkarılan özelliklere göre doğrulamayı gerçekleştirmektedir. Literatürde pasif yöntemleri kullanarak videolarda sahtecilik tespiti yapan çalışmalar; İkili veya çoklu sıkıştırma tespitine göre sahtecilik belirleyen, Video çerçeveleri arasında yapılan sahteciliği tespit eden ve Bölge tabanlı değişiklik tespiti yapan çalışmalar olmak üzere üç ana başlıkta toplanmaktadır.

1.4.1. İkili veya Çoklu Sıkıştırma Tespiti ile Belirlenen Sahtecilikler

Bir video görüntüsünde sahne değişimi oluşuncaya kadar ki çerçeveler neredeyse birbirinin aynısıdır. Video sıkıştırma işlemi ile büyük ölçüde aynı bilgi taşıyan bu

çerçevelerin anlam bütünlüğünü bozmayacak şekilde video verisinden çıkartılmasıyla toplam veri miktarı azaltılmaktadır.

Video sıkıştırma algoritmalarında çerçeveler arası ve çerçeve içi kodlama uygulanmaktadır. Çerçeve içi kodlama, bloğun veya çerçevenin diğer resimlerden bağımsız olarak kodlanmasını sağlarken, çerçeveler arası kodlama ardışık çerçeveler arası benzerlikten yararlanarak daha yüksek sıkıştırma oranı elde etmektedir. Sıkıştırma algoritmaları I, P ve B olmak üzere üç farklı çerçeve biçimi kullanmaktadır. Videodaki I çerçeveleri, çerçevedeki piksellerin tamamının JPEG benzeri bir yöntem ile kodlandığı çerçeve biçimi olup diğer çerçeve tiplerinden daha fazla yer kaplamakta ve çerçeve içi kodlama ile elde edilmektedir. Çerçeveler arası kodlamada P ve B olmak üzere iki tip çerçeve içermektedir. P çerçeveleri önceki çerçeveler, B çerçeveleri ise önceki ve sonraki çerçeveler referans alınarak kodlanır. Çerçeve içi kodlama, çerçeveler arasına göre daha yüksek görüntü kalitesi fakat daha düşük sıkıştırma oranı elde eder. I ve P çerçeveleri B çerçeveleri için referans olarak kullanılır [100].

Farklı çerçevelerin bir araya gelmesi ile bir resimler grubu (Group of Pictures-GOP) oluşur. GOP, bir hareketli görüntüde rastgele olarak erişilebilecek en küçük birimdir. Giriş olarak kodlayıcıya bir video dizisi verilmektedir ve bu video dizisinin her bir çerçevesi, makro blok (MB) adı verilen karelere bölünmektedir. Bir I çerçevesindeki tüm MB'ler ve bir P çerçevesindeki bazı MB'ler, herhangi bir öngörüm olmaksızın doğrudan kodlanan intra-MB'lerdir. Inter-MB'ler üzerinde ise hareket kestirimi yapılmaktadır.

Sıkıştırılmış bir video dosyası üzerinde oynama yapabilmek için önce videonun sıkıştırılmamış formda açılması gerekir. Sıkıştırılmamış video üzerinde oynama yapıldıktan sonra video tekrar sıkıştırılmış formda kaydedilebilmektedir. Eğer bir video iki ya da daha fazla sıkıştırılmışsa bu o videonun sahte bir video olabileceği olasılığını ortaya çıkarır. Çünkü iki veya çoklu sıkıştırma video üzerinde, videoda oynama yapıp yapılmadığının tespitini araştırarak şekilde kalanlar bırakmaktadır. İlk ve sonraki sıkıştırmalarda kullanılan kodlama parametrelerinin çoğu durumda farklı kullanılması bu kalanlara sebep olmaktadır.

Bu alanın öncüleri olan Wang ve Farid, MPEG dizilerindeki I çerçevelerinin iki kere JPEG sıkıştırma sonucunda değişeceği ve çerçeve ekleme ve silme esnasında çerçeveler bir GOP'dan diğerine taşınacağı için oldukça büyük hareket tahmin hatalarının oluşacağı bilgilerinden hareketle MPEG videolardaki sahtecilik tespiti için bir teknik geliştirmişlerdir

[101]. Ancak bu yöntem, silinmiş çerçevelerin sayısı GOP boyutunun tam katı olduğunda başarısız olmaktadır.

Aynı yazarların 2009'da yaptıkları çalışmada ise I çerçevelerindeki her bir makro blok için AKD katsayıları hesaplamışlar ve beklenen dağılımla karşılaştırmışlardır [102]. Karşılaştırma için Normalize Öklid uzaklığından faydalanmışlardır. Çalışmada sadece I çerçeveleri dikkate alınmakla beraber ilk ve ikinci kuantalama ölçeği arasındaki oran 1.7'den küçük olduğunda doğruluk azalmıştır.

2010 yılında Su ve Xu tarafından yapılan çalışmada I çerçevelerinin makro bloklarından elde edilen kuantalanmış AKD katsayılarının dağılımından farklı bir tespit adımıyla yararlanılmıştır [103]. Yöntem basit içerikli ya da ağır hareketler içeren videolarda başarısız olmaktadır.

I, P ve B çerçevelerinden elde edilen AKD katsayılarının dağılımını kullanan başka bir çalışmada Xu ve arkadaşları tarafından 2012 ve 2013'te sunulmuştur [104, 105]. [104]'te 3x3 büyüklüğündeki bir özellik vektörü hesaplanmıştır. Giriş videosunu sahte veya orijinal olarak sınıflayabilmek için ise Destek Vektör Makinesi kullanılmıştır.

Sun ve arkadaşları tarafından 2012'de geliştirilen yöntem [104] çalışması ile benzer olup Parametrik Logaritma yasası kullanılarak sadece I çerçevelerinden 2-D özellik elde edilmiştir [106].

Milani vd. 2012'de Benford yasası doğrultusunda çoklu destek vektör makineleri kullanarak video dizilerine uygulanan sıkıştırma sayısını tespit edebilen bir yöntem önermişlerdir [107]. Destek vektör makinesinin eğitim ve test aşamasında kullanılan özellik vektörünün üretimi için kuantalanmış dönüşüm katsayılarının en önemli rakamının istatistiği kullanılmıştır. Eğer videoda çoklu sıkıştırma yapılmışsa yöntemin performansı düşmektedir.

Videonun iki kere sıkıştırılıp sıkıştırılmadığının tespitinin yanı sıra Vazquez-Padin ve arkadaşları 2012'de eğer bir video iki kere sıkıştırılmış ise ilk sıkıştırmadaki GOP boyutunu tahmin edebilmek için bir yöntem önermişlerdir [108]. Bu yöntemde temel alınan düşünce; bir I çerçevesi P çerçevesi olarak kodlandığı zaman I'daki makro blokların sayısında belirgin bir artış ve S'deki makro blokların sayısında ise önemli bir azalma olacaktır. VPF olarak adlandırılan makro bloklardaki bu değişim kaydedilir ve periyodu hesaplanabilmektedir. VPF ile çalıştığı için H264'ün iyi performans göstermesi beklenir ancak algoritma sabit GOP ile çalışmaktadır.

2013'te Jiang vd. birinci dereceden Markov istatistikleri ile kuantalanmış AKD katsayılarının yatay dikey major köşegen ve minör köşegen boyunca farklarını modellemişlerdir [109]. Bunun sonucunda 9x9 boyutundaki Geçiş Olasılık Matrisi (Transition Probability Matrix) elde etmişlerdir. Her bir GOP Fisher Doğrusal Ayırıştırıcı Analizi (Fisher's Linear Discriminant Analysis-FLD) ile bir ya da iki kere sıkıştırılmış olarak sınıflandırılacak tek bir tespit ünitesi olarak değerlendirilir.

Ravi vd. 2014'te videodan Huber Markov Rasgele Alanına dayalı olarak tahmin edilen sıkıştırma gürültüsünü birinci dereceden Markov süreci ile modellemişlerdir [110]. Bunun sonucunda sahte ya da iki kere sıkıştırılmış çerçeve sayısı, belirlenmiş bir eşik değerinden büyükse videoyu sahte olarak tespit etmişlerdir.

2013'te Subramanyam ve Emmanuel ise verilen çerçevedeki her bir pikseli bulunduğu GOP içerisindeki diğer çerçevelerde o piksele karşılık gelen diğer piksellerden tahmin etmiştir [111]. GOP içerisindeki iki kere sıkıştırılmış çerçeve ya da çerçeveleri belirleyebilmek için pikselin gerçek ve tahmin edilen değeri arasındaki hatayı bir eşik tabi tutmuştur.

Luo vd. 2008'de bloklarda meydana gelen kalanlardan videoda yeniden sıkıştırma yapıp yapılmadığını tespit etmişlerdir [112]. Ancak bu yöntem silinmiş çerçevelerin sayısı GOP boyutunun tam katı olduğunda başarısız olmaktadır.

2012'de Bestagini vd. ilk kodlama aşamasında kullanılan kodek türünü belirleyebilmek için bir yöntem önermişlerdir [113]. İkinci kere sıkıştırma sonucunda oluşturulan video aynı kodek türü ve parametreleri ile yeniden kodlandığında bu videoya oldukça benzer yeni bir video oluşturacak olması düşüncesini temel almışlardır. Bunun için yazarlar test videolarını farklı kodek türleri ve parametreleri ile yeniden sıkıştırarak elde edilen videolar arasındaki benzerlikleri araştırmışlardır.

Literatürde bu alanda önerilmiş algoritmaların çoğu, video ilk sıkıştırmada kullanılan aynı kodlama parametreleri ile yeniden sıkıştırıldığında başarısız olmaktadır. Huang ve arkadaşları bu durumu göz önünde bulundurarak 2014'te yeni bir yöntem önermişlerdir [114]. Yöntem, bir çerçevenin aynı kuantalama matrisiyle tekrar tekrar yeniden sıkıştırıldığında art arda sıkıştırılan iki versiyon arasındaki farklı AKD katsayılarının sayısı monoton bir şekilde azalacak olduğu düşüncesini temel almıştır.

Video doğrulama alanında diğer kodek türleri ile karşılaştırıldığında H.264/AVC ile kodlanan videolarla uğraşmak daha zordur. Tagliasacchi ve Tubaro 2010 yılında blok seviyesindeki hareket resüdülerini kullanarak H.264 ile kodlanan videolarda kuantalama

parametresini bulmak için bir yöntem geliştirmişlerdir [115]. Ancak yöntem sabit GOP yapısı ile çalışmaktadır.

Bir başka çalışmada da Li ve Forchhammer tarafından 2009 yılında videodaki MPEG2 parametre bilgisi tahmin edilmeye çalışılmıştır [116]. Valenzise vd. 2010 yılında H.264 ile kodlanan videolarda kuantalama parametrelerini ve hareket vektörlerini belirleyebilecek bir yöntem önermişlerdir [117].

2010'da Yammine vd. sıkıştırılmış videoyu açarak, bu videonun gürültü varyans davranışına dayalı olarak GOP yapısını bulacak bir yöntem geliştirmişlerdir [118]. Bu yaklaşım GOP yapısı bilinmediği zaman video sahteciliği tespitinde fayda sağlayabilmektedir. Hızlı Gürültü Varyans Tahmin (Fast Noise Variance Estimation algorithm) algoritması gürültüyü ölçmek için kullanılmaktadır.

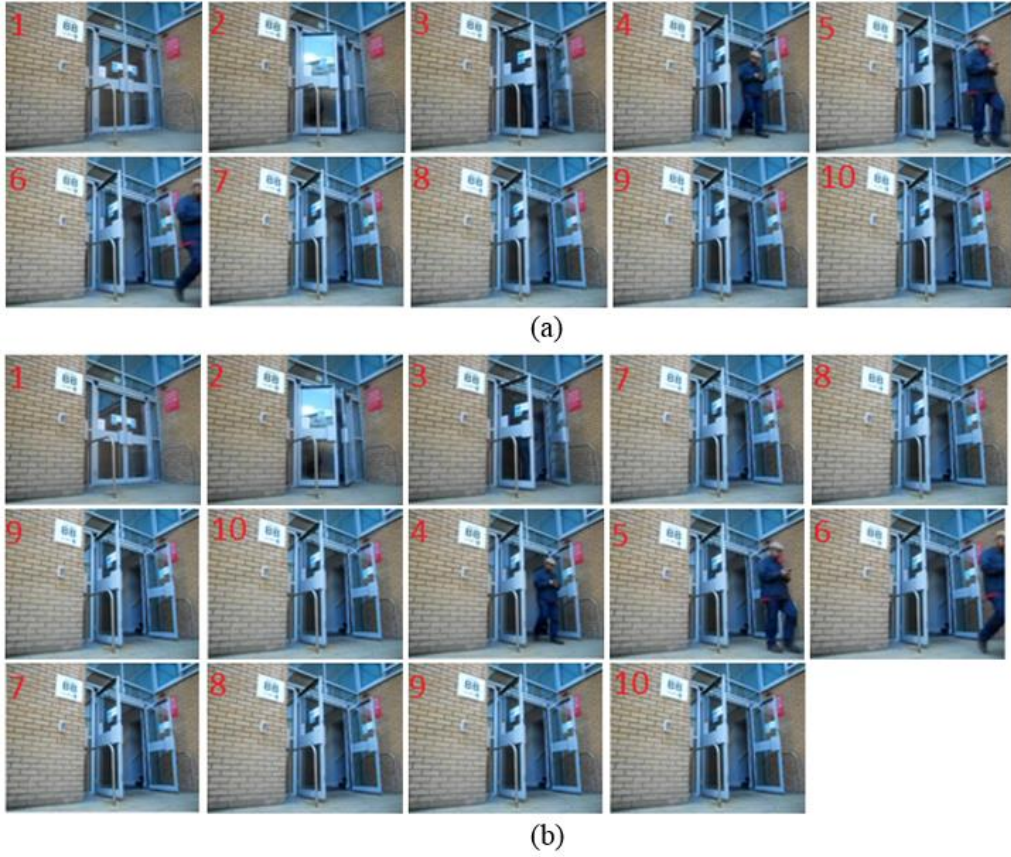
1.4.2. Çerçeveler Arası Yapılan Sahtecilikler

Video çerçeveleri arasında yapılan sahtecilikler çerçeve ekleme, çerçeve silme ve çerçeve tekrarlama sahteciliği olarak üçe ayrılmaktadır. Çerçeve ekleme sahteciliği videonun herhangi bir kısmına farklı ya da aynı videodan alınan çerçeve grubunun eklenmesi ile oluşturulmaktadır. Çerçeve silme sahteciliği videodaki belirli bir çerçeve grubunun silinmesiyle oluşturulmaktadır. Çerçeve tekrarlama sahteciliği ise video içerisindeki belirli bir çerçeve grubunun kopyalanarak yine aynı video içerisinde başka bir yere yapıştırılmasıyla oluşturulmaktadır. Video çerçeveleri arasında yapılan sahteciliklerin en yaygın ve gerçekleştirmesi kolay olanı çerçeve tekrarlama sahteciliğidir. Tez kapsamında da video üzerinde çerçeve tekrarlama sahteciliği tespiti gerçekleştirilmiştir. Bu bölümde Çerçeve Tekrarlama sahteciliği hakkında bilgi verilecek ve ilgili literatürden bahsedilecektir.

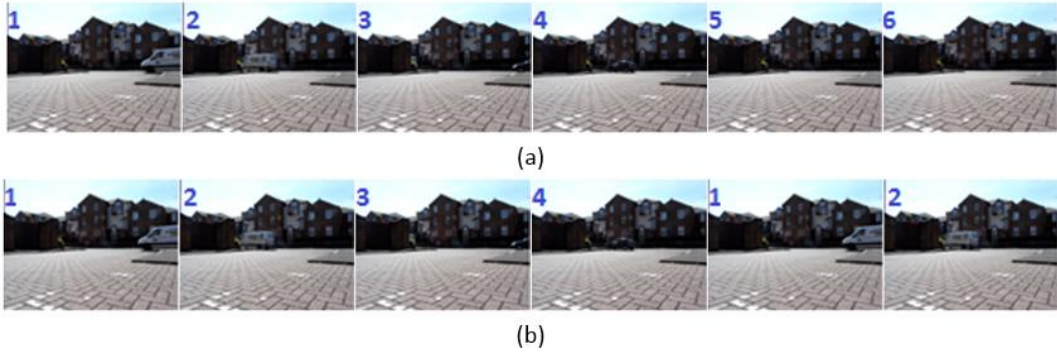
1.4.2.1. Çerçeve Tekrarlama Sahteciliği

Literatürde Çerçeve Tekrarlama Sahteciliği tespit yöntemleri çerçeveler arasındaki uzaysal ve zamansal benzerliği kullanarak videoda yapılan sahtecilikleri tespit etmektedir. Çerçeve Tekrarlama sahteciliği, video içerisindeki bir çerçeve grubunu kopyalayarak yine aynı video içerisindeki başka bir yere yapıştırmakla ya bir hareketi kapamakta ya da bir

hareketi tekrar ederek gerçekleştirilmektedir. Bu tür sahtecilik yönteminin, günümüzdeki modern video düzenleme yazılımları ile gerçekleştirilebilmesi kolay iken, insan gözü tarafından algılanabilmesi bir o kadar zordur. Bu sahtecilik de amaç video içerisindeki bir hareketi kapamak ya da tekrar etmektir. Şekil 1.28.'de hareketi kapama amaçlı yapılan Çerçeve Tekrarlama sahteciliğine bir örnek verilmiştir. Şekil 1.28.(a) ve 1.28.(b) sırasıyla orijinal ve sahte video sıralarını göstermektedir. Orijinal videodaki 7-10 arasındaki çerçeve grubu yine aynı videodaki 3. çerçeveden sonra yerleştirilerek kapıdan giriş yapan adamın giriş ve çıkışı zaman olarak ertelenmiş ve Şekil 1.28.(b)'de verilmiş olan sahte video elde edilmiştir. Şekil 1.29.'da ise yine Şekil 1.29.(a)'da verilen orijinal videodaki ilk iki çerçeve, yine aynı videodaki dördüncü çerçeveden sonra yerleştirilip araba sahneye iki kere giriyormuş gibi gösterilerek Şekil 1.29.(b)'deki sahte video oluşturulmuştur. Sahte videonun oluşturulmasının ardından kaydedilmesi esnasında yeniden kodlama işlemi gerçekleştirilmektedir. Yeniden kodlama işlemi ise video üzerinde fark edilmeyen kalanlara sebep olacağı için, kopyalanıp yapıştırılan çerçeveler birebir aynı özellik göstermeyecektir. Yeniden kodlamadan kaynaklanan bu bozulmalar Çerçeve Tekrarlama sahteciliğinin tespitini oldukça zor hale getirmektedir.



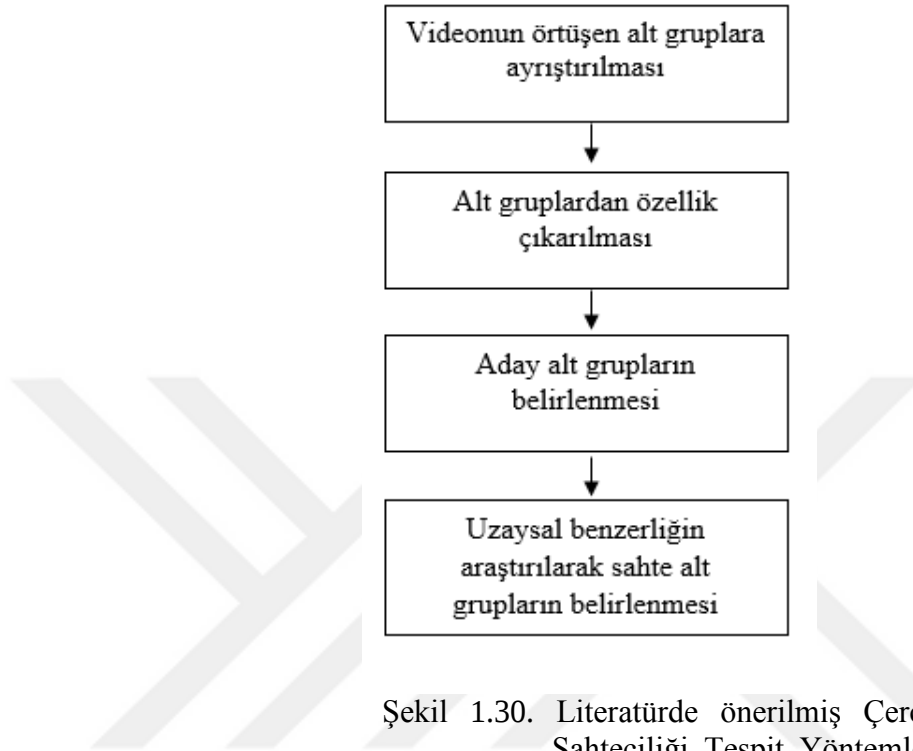
Şekil 1.28. (a) Orijinal (b) Sahte video



Şekil 1.29. (a) Orijinal (b) Sahte video

Literatürde Çerçeve Tekrarlama sahteciliğinin tespiti için önerilen yöntemler çerçeveler arasındaki hem zamansal hem de uzaysal benzerlikten faydalanmaktadır. Şekil 1.30. 'da görüldüğü gibi önerilen yöntemler genel olarak benzer adımlardan oluşmaktadır. Bu adımlarda öncelikle video örtüşen alt gruplara ayrıştırılmakta, bu alt gruplardan dayanıklı özellikler çıkarılmakta sonrasında bu özelliklere göre aday alt grupların

belirlenmesi gerçekleştirilmektedir. Son aşamada ise aday alt grupların benzerliği uzaysal olarak araştırılarak gerçek sahte gruplar tespit edilmektedir.



Şekil 1.30. Literatürde önerilmiş Çerçeve Tekrarlama Sahteciliği Tespit Yöntemlerinin genel akış diyagramı

Literatürde çerçeve tekrarlama sahteciliğini ortaya koyan ve tespit etmeye çalışan ilk çalışma 2006 yılında Wang ve arkadaşları tarafından önerilmiştir [119]. Yazarlar tarafından önerilen yöntem, videoyu n çerçeveden oluşan ve birbirleri ile örtüşen alt çerçeve gruplarına ayırmaktadır. Elde edilen her grup için kendi içinde korelasyon matrisi üretilir. Korelasyon matrisleri arasındaki benzerlik, gruplar arası benzerliğin işaretçisidir. Gruplar arası benzerliğe karar verilmesi durumunda, gruplara karşılık düşen çerçeveler arasında benzerlik kontrolü yapılır. Çerçeveler arasında benzerliğin tespit edilmesi durumunda, yöntem videoda kopyalanıp yapıştırılan çerçevelerin olduğuna karar vermektedir. Önerilen yöntemin algoritması aşağıda detaylandırılarak adımlar halinde verilmiştir.

Adım 1: L uzunluklu video birbiri ile örtüşen n alt gruba ayrıştırılır.

Adım 2: Her bir grup için $n \times n$ boyutunda GK korelasyon matrisi hesaplanır. Bu matrisin (i,j) elemanı hesaplama yapılan alt grup içerisindeki i . ve j . çerçeveleri arasındaki korelasyon katsayısını ifade etmektedir.

Adım 3: Her alt grup için hesaplanan GK matrislerinin kendi aralarında korelasyon değeri hesaplanır. Herhangi iki alt gruptan elde edilen korelasyon matrisleri arasındaki korelasyon değeri belirlenen eşik değerinden büyükse bu gruplar Çerçeve Tekrarlama sahteciliği yapılan aday gruplar olarak işaretlenir.

Adım 4: Belirlenen aday gruplar için blok düzeyinde BK korelasyon matrisleri hesaplanır. Bunun için ilk olarak gruplar içerisindeki çerçeveler birbiri ile örtüşmeyen m tane alt bloğa parçalanır. Aday gruplar içerisindeki karşılıklı çerçeveler birlikte değerlendirilir. Örneğin aday gruplar 10. ve 40. gruplar ise ilk olarak 10. grubun ilk çerçevesi ile 40. grubunun ilk çerçevesi arasında BK hesaplanır. Bu işlem tüm çerçeveler için tekrarlanır. BK matrisinin elemanlarını bu çerçevelerin karşılıklı blokları arasındaki korelasyon katsayı değerleri oluşturur. Tüm çerçeveler arasında hesaplanan BK matrislerinin değerlerinin hepsi belirlenen eşik değerinden büyükse bu gruplar kesin olarak Çerçeve Tekrarlama sahteciliği yapılmış gruplar olarak işaretlenir.

Bu çalışmanın ardından 2012 yılında Lin vd. tarafından videoda tekrarlayan çerçeve grubunun tespit edilebilmesi için video alt gruplara ayrılmış ve her bir grup teker teker sorgu grubu olarak değerlendirilmiştir [120]. Yöntem, iki komşu çerçevenin benzerlik değerlendirmesinde bu çerçeveler arasındaki renk histogram farkından (Histogram Difference-HD) yararlanmıştır. Aday ve sorgu gruplarının histogram farkı hesaplanmış ve bu farklar arasındaki korelasyon değeri elde edilmiştir. Bu değer belirlenen eşik değerinden daha büyükse sorgu ve aday gruplarının karşılıklı çerçeveleri bloklara ayrılmış ve bloklar arasında HD hesaplanmıştır. Sonrasında bir eşikleme yaklaşımı ile tekrarlanan çerçeve grupları işaretlenmiştir. Önerilen yöntem aşağıda adımlar halinde verilmektedir.

Adım 1: Şüpheli video örtüşen gruplara ayrıştırılır ve bu gruplar sahtecilik yapılmış aday gruplar olarak değerlendirilir. İlk grup sorgu grubu olarak işaretlenir. Aday gruplarla sorgu grubunun benzerliği test edildikten sonra bir çerçeve kaydırılarak, sorgu grubu ta ki son video çerçevesine gelene kadar sürekli değiştirilir.

Adım 2: Aday grup için ve o anki sorgu grubu için komşu çerçeveleri arasında HD hesaplanır. HD'ler arasındaki korelasyon değeri belirlenen eşik değerinden daha büyükse bu gruplar benzer kabul edilir.

Adım 3: Benzer olarak belirlenen aday ve sorgu gruplarının çerçevelerine sıkıştırma etkisini azaltmak için yumuşatma işlemi uygulanır ve çerçeveler bloklara ayrıştırılır.

Adım 4: Hem aday hem de sorgu grubundaki her bir bloğun Y bileşeni için histogram hesaplanır. Grupların karşılıklı çerçevelerine düşen karşılıklı blokları için

histogram farkına bakılır. Bu fark belirlenen eşik değerinden daha az ise bloklar benzer olarak kabul edilir.

Adım 5: Aday ve sorgu grubu arasındaki benzer blokların sayısına bakılarak Çerçeve Tekrarlama sahteciliği yapıp yapılmadığına karar verilmektedir.

2014 yılında Yang vd. SVD yönteminde faydalanan Çerçeve Tekrarlama Sahteciliğinin tespiti için benzerlik analizi tabanlı bir yöntem önermişlerdir [121]. Her bir çerçeveden SVD yöntemiyle özellik çıkarılmıştır. Çıkarılan özellik vektörlerinin ilk çerçeve olan referans çerçeveye benzerliği Öklid uzaklığı ile tespit edilmiştir. Çerçeveler arası benzerlik hesaplanan benzerlik değerlerinin gruplanması ile belirlenmektedir. Benzer olarak tespit edilen aday video parçalarındaki çerçeveler üzerinde ise karşılıklı blok düzeyinde karşılaştırma yapılarak tekrarlanan çerçevelere karar verilmiştir. Önerilen yöntemin detayları aşağıda adımlar halinde verilmektedir.

Adım 1: Her bir çerçeveden SVD yöntemiyle özellik vektörü çıkarılır.

Adım 2: İlk çerçeve referans çerçeve kabul edilerek her çerçeveden elde edilen özellik vektörünün referans çerçevenin özellik vektörüne olan Öklid uzaklığı hesaplanır.

Adım 3: Bu uzaklık değeri her bir çerçeve için yeni bir özellik olarak atanır. Video birbiri ile örtüşen gruplara ayrılır ve bu özellikler birleştirilerek her bir grup için özellik vektörü elde edilir.

Adım 4: Çerçeve grupları arasındaki korelasyon değeri belirli bir eşik değerinden büyükse bu gruplar sahtecilik yapılmış aday gruplar olarak belirlenir.

Adım 5: Aday gruplardaki çerçeveler örtüşmeyen 8x8 büyüklüğündeki bloklara ayrıştırılır. Eşleşen çerçeveler için Rasgele Blok Eşleşmesi yöntemi uygulanır. Bunun için eşleşen çerçevelerde rasgele 500 blok seçilir. Seçilen blokların gri seviye ortalama piksel değerleri birleştirilerek GOM matrisi elde edilir. Eşleşen çerçevelerin her ikisinden de elde edilen GOM matrisleri arasındaki korelasyon değeri gruptaki tüm çerçeve çiftleri için 1'e eşitse bu çerçeve grubu tekrar yapılmış çerçeve grubu olarak işaretlenir.

2015'te Singh vd. tarafından önerilen yöntemde video çerçeveleri 4 alt bloğa bölünüp her bir çerçeveden 9 özellik çıkarılarak Çerçeve Tekrarlama sahteciliği tespit edilmeye çalışılmıştır [122]. Yöntemde özellik çıkarımından sonra elde edilen özellikler leksikografik olarak sıralanmıştır. Komşu çerçeveler arasında hataların ortalama karekökü belirlenen eşik değerden daha düşükse bu çerçeveler ihmal edilmekte geri kalanlar şüpheli olarak düşünülmektedir. Tekrarlayan çerçeve gruplarını tespit edebilmek içinse şüpheli

çerçeveler arasında korelasyon hesaplanmaktadır. Ancak önerilen yöntem çok az sayıda video ile test edilmiştir.

2015'te yapılan başka bir çalışmada ise Zhang vd. Yerel İkili Örüntüler (Local Binary Patterns-LBP) ile kodlanmış komşu çerçeveler arasındaki korelasyon katsayılarını kullanarak çerçeve tekrarlama sahteciliğini tespit edecek bir yöntem önermişlerdir [123]. Eğer korelasyon değeri yüksek çıkarsa video orijinal olarak kabul edilmekte aksi takdirde Çerçeve Tekrarlama sahteciliğinin yapılmış olabilirliği göz önünde bulundurulmaktadır. Yöntemde Yerel İkili Örüntülerin Ardışık Korelasyon Katsayılarından yararlanılmıştır. Zheng vd. ise Çerçeve Tekrarlama sahteciliğini tespit etmek için Blok-İçi Parlaklık Varyans Tanımlayıcısı'nı (Block-wise Brightness Variance Descriptor-BBVD) kullanmıştır [124]. Yöntemde orijinal videolarda iki komşu çerçeve arasındaki parlaklığın değişim oranının sabit bir değere yakın olduğunu, ancak sahte videolarda bu değer daha büyük bir varyasyon gösterdiği ortaya konulmuştur. Bunun için öncelikli olarak video küçük örtüşen alt gruplara ayrılıp her bir çerçeve 4x4 bloklara bölünmüştür. Her bir grup içinde BBVD oranı hesaplanmıştır. Ortalama BBVD oranının belirlenmiş bir eşik değeriyle kıyaslanması sonucu sahtecilik tespiti gerçekleştirilmiştir.

Bir diğer çalışmada da video sahteciliğinin tespitinde P çerçevelerindeki kuantalama ve kalan hatalarının oluşturduğu izlerden faydalanılmıştır [125]. Daha sonra, frekans alanında kuantalama hatasının izlerini artırmak için bir dalgacık tabanlı bir algoritma önerilmiştir. Deneysel sonuçlar özellikle yüksek sıkıştırma oranlarına sahip videolar için önerilen algoritmanın etkinliğini göstermektedir.

2017'deki bir çalışmada ise Liu vd. Zernike Karşıt Renklilik (Zernike Opponent Chromaticity) momentlerini kullanarak Çerçeve Tekrarlama sahteciliğini tespit etmişlerdir [126]. Çalışmada öncelikle anormal noktalar çıkarılarak kaba bir tespit gerçekleştirilmektedir. Kaba tespit yeterli etkinliğe sahip olmadığı için ince tespit önerilmiştir. Çalışmada rapor edilen deneysel sonuçlar, bu algoritmanın kıyasladığı algoritmalara göre daha yüksek bir etkinlik ve doğruluğa sahip olduğunu göstermektedir.

1.4.3. Bölge Tabanlı Sahtecilikler

Bu bölümde videodaki bölge tabanlı sahteciliklerin tespiti alanında yapılmış çalışmalar verilecektir. Bu alanda literatürdeki çalışmalar kendi içerisinde iki alt gruba ayrılmaktadır. Bunlardan ilki çerçeve içerisindeki bir bölgenin yine aynı çerçeve

içerisindeki başka bir bölgeye yapıştırılması ya da çerçeve gruplarındaki belirli bölgelerin kopyalanarak yine aynı video içerisindeki başka bir çerçeve grubuna yapıştırılması ile gerçekleştirilmiş video sahteciliklerini tespit eden yöntemlerdir. Videolarda kopyala-yapıştır sahteciliği olarak adlandırılan bu sahteciliklerde, kopyalanan bölge ya da bölgeler yine aynı video içerisinden alınmaktadır. Giriş bölümünde Şekil 1.7.'de kopyala-yapıştır sahteciliğine örnek verilmiştir.

İkinci alt sınıf olan video birleştirme sahteciliğinde ise, videoda yer alan çerçeve ya da çerçeveler içerisindeki bir bölge, başka bir videodaki çerçeve ya da çerçevelerden alınan bölge/bölgeler ile yer değiştirilmektedir. Giriş bölümünde Şekil 1.8.'de videolarda birleştirme sahteciliğine örnek verilmiştir.

Videolarda gerçekleştirilen bölge tabanlı sahteciliklerin görsel izlerini kapatmak adına yapıştırılan bölgelere geometrik dönüşümler ve rötuşlamalar uygulanması bu sahteciliklerin tespitini oldukça zor hale getirmektedir. Literatürde bu alanda önerilmiş çalışmaların çoğunda geometrik dönüşümler göz önünde bulundurulmamıştır.

Wang ve Farid tarafından 2007'de interlaced ve de-interlaced videolarda bölgesel sahteciliğin tespit edilmesi için bir yöntem önerilmiştir [127]. De-interlaced videolarda var olan uzaysal ve zamansal korelasyonların sahtecilik işleminden dolayı bozulabildiğinin gösterildiği çalışmada, komşu çerçevelerin alanları arasındaki hareket ile tek bir çerçevenin alanları arasındaki hareketin bozulabildiği gözlemlenmiştir. Yazarlar çalışmalarında, çerçeve oranı dönüşümünü gerçekleştirerek video sahteciliğini tespit etmektedir. Ancak sıkıştırma kalanları ve gürültü yöntemin performansını düşürmektedir.

Yine aynı yazarlar [119]'da önermiş oldukları çalışmada ise, korelasyon bilgisini kullanarak videolarda Çerçeve Tekrarlama ve bölgesel sahtecilikleri yakalayabildiklerini göstermişlerdir. Sahtecilik işleminin periyot ile senkronize yapılması durumunda, yöntemin başarımının sürdürülebilmesi için, hareketli kamera çekimlerinde kameranin hareketi ve sahnenin periyodu da göz önüne alınmalıdır. Önerilen yöntem statik arka planlı video çerçevelerinde başarısız olmaktadır.

Hsu vd. 2008 yılında önerdikleri çalışmada, çerçevelerdeki gürültü artıklarının blok seviyesindeki korelasyon değerleri ve dağılımları, orijinal ve sahte videolar için Gauss Karışım Modeli (Gaussian Mixture Model-GMM) kullanılarak modellenmiştir [128]. Beklenti maksimizasyonu (Expectation Maximization) algoritması GMM'nin parametrelerini kestirmekte kullanılmıştır. Kestirilen parametreler kullanılarak, Bayes

sınıflayıcı ile optimum eşik değeri belirlenmiştir. Fakat önerilen bu yöntem özellikle hareketli kameralar ve dinamik arka planlı videolar için etkin bir yöntem değildir.

2013 yılında Li vd., komşu çerçeveler arasında hesaplanan hareket vektörlerinin büyüklüğü ve yönlenim bilgisini, video içerisindeki orijinal bölgeler ile sahte bölgelerin ayrımı için kullanmıştır [129]. Hareket vektörlerinin dağılımı normal bölgeler için, sahte bölgeler ile kıyaslanınca, olağanlık göstermektedir. Bu nedenle, sahte bölgelerde hareket vektörlerinin oluşturduğu açının varyansı, diğer bölgelere göre daha büyük olmaktadır.

Chen vd. tarafından 2015'te önerilen yöntemde ise çerçevelerdeki hareket kalıntılarından çıkartılan özellikler kullanılarak, çerçeve üzerindeki sahte bölgelerin tespit edilmeye çalışılması hedeflenmiştir [130]. Önerdikleri sistemin değişken GOP yapıları videolarda da çalışabilmesi için hareket kalıntılarını üretirken çarpışma operatörleri kullanılmışlardır. Yöntemde k ile indekslenen çerçeveye ait hareket kalıntısını elde edebilmek amacı ile k 'nın komşu çerçevelerinden faydalanılmıştır. Görüntü steganalizi alanında kullanılan özellikler: CC-PEV, SPAM, CF, CDF, SRM, CC-JRM ve J+SRM, hareket kalıntılarından özellik çıkarma amacı ile kullanılmıştır. Elde edilen deneysel sonuçlar, CC-PEV ve J+SRM özelliklerinin daha iyi sonuçlar verdiğini göstermiştir.

Video iç boyama teknikleri ile beraber, video içerisindeki hareketli bir objenin ortadan kaldırılmasından sonra görüntüde hayalet gölge kalıntıları (Ghost shadow artifact) oluşmaktadır. Bu kalıntılar hareketli ön planın tutarsızlıklarından algılanabilmektedir. Zhang vd. 2009'da hareketin izini toplamsal çerçeve farklarından elde etmiştir [131]. Yöntemin MPEG sıkıştırılmalara karşı dayanıklı olduğu sonuçlarında gösterilmiştir. Ancak yöntem her bir çerçeve için sahte bölgelerin yerini tam olarak tespit edememekle beraber sadece statik arka planlı videolarda iyi sonuç vermektedir.

2010 yılında Kobayashi vd. görüntü karakteristiklerini kullanarak statik sahneli videolarda şüpheli bölgeleri tespit edebilecek bir yöntem geliştirmişlerdir [132]. Parlamadan kaynaklı gürültünün değişimi, gürültü seviyesi fonksiyonu (Noise Level Function-NLF) kullanılarak tanımlanmıştır. Aynı zamanda her pikseldeki gürültü karakteristiğini kontrol etmek için bir olasılık modeli kullanılmıştır. Sahte bölgelerdeki piksellerde NLF fonksiyonun aldığı değerler diğer bölgelere göre farklılıklar göstermiştir. Yöntem sadece statik sahneli videolarla çalışmakla birlikte yöntemin performansı video sıkıştırmada kullanılan kodeğe göre değişmektedir.

Chetty vd. [133] 2010'da ve Goodwin ve Chetty [134] 2011'de düşük bant genişlikli internet üzerinden gösterilen videolardaki sahteciliğin tespit edilmesi için çerçeveler arası

ve çerçeve içi piksel alt bloklarının kalan özelliklerinden faydalanmışlardır. Yine bu önerilen yöntemler de sadece statik arka planlı videolarda iyi sonuç vermektedir.

Subramanyam vd. 2012 yılında yapmış oldukları çalışmada, Yönlü Meyillerin Histogramı (Histogram of Gradient-HoG) özelliğini her bir çerçevenin örtüşen alt bloklarından elde etmişlerdir [135]. Çerçeveedeki her bloğa ait HoG özelliği, aynı çerçeve içerisindeki diğer bloklardan elde edilen özellikler ile kıyaslanmakta ve çerçeve içi sahtecilik tespiti yapılmaya çalışılmaktadır. Çerçeveler arası sahteciliği tespit edebilmek için ise, bir çerçeveedeki her bloğa ait HOG özellikleri, diğer çerçevede yer alan tüm bloklara ait özellikler ile kıyaslanmaktadır. Çalışmada sabit GOP yapısı kullanılmasıyla birlikte sadece videodaki kopyala-yapıştır sahteciliği tespit edilmektedir.

2014 yılında Lin vd. ise iç boyama teknikleri ile beraber gerçekleştirilen video sahteciliklerini tespit edebilecek bir yöntem önermişlerdir [136]. İç boyama yöntemleri çerçeveedeki kaldırılan nesnelerden arda kalan boşlukları doldurmak için kullanılmaktadır. Çalışmada çerçeve hareket bilgisi, çerçeve gruplama ve merkezileştirme gri seviyeye dönüştürülmüş videodan hesaplanmaktadır. Uzaysal-zamansal tutarlılık analizi ise her çerçeve grubu üzerinden bağımsız olarak gerçekleştirilmektedir. Bu doğal olarak yüksek ya da anormal derecede düşük tutarlılığı olan bölgelerin tanımlanması için kullanılan Grup Tutarlılık Anormallik Örüntüsü (Group Coherence Abnormality Pattern) üretir. Bu örüntü kullanılarak sahtecilik işlemi yapıp yapılmadığı tespit edilmektedir. Önerilen yöntem, statik ve dinamik arka plana sahip videolar ile çalışabilirken sıkıştırmanın artırılması sonucunda yöntemin başarısı düşmektedir.

Pandey vd. 2014 yılında uzaysal ve zamansal kopyala-yapıştır sahteciliğini tespit edebilmek amacı ile iki farklı yöntem önermişlerdir [137]. SIFT ve kNN yaklaşımları zamansal sahteciliği tespit etmekte kullanılırken, gürültü kalanları arasındaki korelasyon kullanılarak uzaysal sahteciliğin tespit edilmesi hedeflenmiştir. Önerilen çalışmada sıkıştırma oranı arttıkça tanılama doğruluğu azalmaktadır.

Su ve arkadaşları 2015'te statik arka planlı videolardaki hareketli ön planın çıkarılması sahteciliği üzerine yoğunlaşmışlardır [138]. Önerilen yöntemde, o anki ve referans çerçeve arasındaki fark hesaplanmış k-Tekil Değer Ayrıştırması (k-Singular Value Decomposition-k-SVD) yöntemi kullanılarak özellik çıkarımı gerçekleştirilmiştir. Her çerçeve için elde edilen tespit sonuçları birleştirilerek, sahtecilik tespiti gerçekleştirilmiştir. Önerilen yöntem statik arka planlı videolarla çalışmaktadır. Aynı zamanda silinen ön plan çok küçükse ya da çok hızlı hareket içeriyorsa yöntemin tanılama başarısı düşmektedir.

D'Amiano vd. 2015 yılında yapmış oldukları çalışmada, Zernike momentlerini ve 3-D parça eşleşmesini kullanarak kopyala-yapıştır sahteciliğini tespit etmişlerdir [139]. Yazarlar Cozzolino ve arkadaşlarının 2014 yılında [140] görüntüler için kullandığı 2-D parça eşleşmesini zaman bilgisini de göz önünde bulundurarak genişlettiler. Yöntemin doğrulama oranı düşük olmasına rağmen, dönme ve ölçeklemeden bağımsızdır.

2015 yılında Bidokhti vd. ise komşu çerçeveler arasındaki Lucas Kanade optik akışını kullanarak kopyala-yapıştır sahteciliğinin tespiti için bir yöntem önermişlerdir [141]. Her video çerçevesi, şüpheli kısım ve geri kalan kısım olarak iki parçaya ayrılmıştır. Her iki kısım üzerinde optik akış ve optik akış değişim faktörü ayrı ayrı hesaplanmıştır. Eğer optik akış değişim vektöründe ikincil tepeler yer alıyorsa, video sahte video olarak etiketlenmiştir.

2. YAPILAN ÇALIŞMALAR

Tez kapsamında önerilen sistem, görüntü ve video dosyalarının doğrulamasını gerçekleştirmektedir. Doğrulama aşamasında görüntü üzerinde Kopyala-yapıştır, video üzerinde ise Çerçeve Tekrarlama ve bölge tabanlı sahteciliklerin tespiti yapılmıştır. Literatürde yer alan çalışmalar, etkin bir görüntü/video sahteciliklerinin tespiti için aşağıda verilen unsurları iyileştirmeyi hedeflemişlerdir. Bu unsurlar:

- Görüntü/video dosyaların adli kurumlarda delil niteliği taşıyacak olması nedeniyle görüntü/video sahteciliklerin tespiti için önerilecek yöntemin tanılama performansının yüksek olması önemli bir unsurdur.

- Doğrulaması yapılacak olan dosyanın özellikle video dosyası olması durumunda önerilen yöntemin hesaplama karmaşıklığının azaltılarak işlem zamanının düşürülmesi hedefler arasında yer almaktadır.

- Önerilecek sistemin tüm görüntü veri tabanlarında test edilebilmesi için kullanılan eşik değerlerinin görüntü/video içerikleri ve boyutundan bağımsız bir şekilde dinamik olarak belirlenmesi önemli bir unsurdur.

- Literatürde görüntü/video sahtecilikleri tespiti için yapılan çalışmalar çoğunlukla kullandıkları özellik çıkarım yöntemine göre farklılar göstermektedir. Görüntü sahteciliklerin tespiti için önerilecek olan özellik çıkarım yönteminin ara ve son işlem operasyonlarına dayanıklı olması, video sahteciliklerin tespiti için önerilecek olan özellik çıkarım yönteminin ise sıkıştırma türünden bağımsız ve sıkıştırmaya dayanıklı olması gereken bir unsurdur.

- Önerilecek olan yöntemin düz geçişsiz dokularda yapılan sahteciliği dahi tespit edebilir şekilde geliştirilmesi hedefler arasında yer almaktadır.

Görüntü ve video sahteciliklerinin tespitinde, literatürdeki araştırmacılar tarafından üzerinde durulan unsurların iyileştirilmesi ve sorunlarının giderilmesi tez kapsamında yapılan çalışmaların hedefini teşkil etmiştir. Genel bilgiler kısmında verilen görüntü/video üzerindeki sahteciliklerin tespitine yönelik geliştirilmiş yöntemler irdelenerek verilen hedefler doğrultusunda iyileşme sağlayacak yeni görüntü/video sahtecilik tespit yöntemleri önerilmiştir.

Tez kapsamında yapılan çalışmalar görüntü ve video için ayrılarak aşağıda maddeler halinde verilmektedir.

a) Görüntü alanında yapılan çalışmalar

1. Kopyala-yapıştır sahteciliğinin tespitinde AKD tabanlı yöntemlerin son işleme operasyonlarına karşı sağlamlığını artırmak için Renk Momentleri ve Renk Düzeni Tanımlayıcılarına (Colour Layout Descriptors-CLD) dayanan bir yöntem önerilmiştir [142]. Çalışmada görüntü öncelikle örtüşen alt bloklara ayrıştırılmaktadır. Sonrasında üç renk kanalı için her bloğun ilk renk momentleri hesaplanır ve bloklar, renk momentlerine göre on üç kümeye gruplanır. Böylece benzer renk dağılımına sahip bloklar aynı grup içerisinde yer almış olacaktır. Ardından, her kümedeki bloklar için özellik vektörü çıkarımı, işlemi hızlandırmak için eş zamanlı olarak ayrı iş parçacıkları (threads) tarafından yapılır. Özellik çıkarımında CLD'den faydalanılmıştır. Deneysel sonuçlardan görüleceği üzere yöntem son işlem operasyonları sonrasında bile diğer AKD tabanlı algoritmalara kıyasla daha yüksek tanılama performansına sahiptir.

2. Kopyala-yapıştır sahteciliği tespiti için önerilen bir başka çalışmada özellik vektörü değerlerini belirli bir aralıkta sınırlamak için AKD-faz'dan ve test görüntüsünün sıkıştırma geçmişini belirleyebilmek için Benford'un Genelleştirilmiş yasasından yararlanılmıştır [144]. Yöntemde özellik vektörlerinin benzerliği testinde Öklid uzaklığı ve kross korelasyon yerine özellik vektörleri değerlerinin karşılıklı elemanları arasındaki eşitliğe bakılmış ve görüntünün sıkıştırma geçmişinden yararlanılarak eşik değer otomatik olarak belirlenmiştir. Elde edilen sonuçlar göz önünde bulundurulduğunda önerilen yöntem kopyalanıp yapıştırılan bölgeleri farklı senaryolar altında dahi tespit edebilmekte ve literatürdeki benzer çalışmalarla kıyaslandığında daha yüksek doğruluk oranı ve daha düşük hata oranı vermektedir.

3. Bir diğer çalışmada Kopyala-yapıştır sahteciliği tespitinde pasif görüntü doğrulama tekniklerine dayalı bir yöntem önerilmiştir [148]. İlk olarak görüntü birbiri ile örtüşen alt bloklara parçalanmaktadır. Her bir bloktan LBP kullanılarak özellik çıkarılmıştır. Sonrasında LBP ile etiketlenen bloklar AKD yöntemiyle frekans uzayına geçilmiştir. AKD'si alınmış her bir bloğun zigzag taraması sonucu elde edilen ilk 15 katsayının işaret bilgisi ile birlikte bloğun Y, Cb, Cr değerlerinin ortalaması o blok için özellik olarak çıkarılmıştır. Deneysel sonuçlara bakıldığında önerilen yöntem görüntü ataklara maruz kalsa dahi AKD tabanlı diğer çalışmalarla kıyaslandığında yüksek doğruluk oranı ve düşük yanlış negatif değerlerine sahiptir.

4. Kopyala-yapıştır sahteciliğini tespit etmek için önerilen bir başka çalışmada örtüşen alt bloklara ayrıştırılan görüntüde her bir bloğu etiketlemek için Yerel Faz

Nicemleyicisi (Local Faz quantization-LPQ) yönteminden faydalanılmıştır [149]. Etiketlenen blokların sütunlarının ortalama değerleri özellik vektörü olarak çıkarılmıştır. LPQ yöntemi literatürde ilk kez Kopyala-yapıştır sahteciliği tespitinde kullanılmıştır. Deneysel sonuçlar, yöntemin diğer yöntemlere kıyasla yüksek düzeyde bulanıklaştırma işlemi altında dahi yüksek doğruluk oranları ve daha düşük yanlış negatif değerlere sahip olduğunu göstermektedir. Önerilen yöntem aynı zamanda çoklu kopyala-yapıştır sahteciliğini de tespit edebilmektedir.

5. Kopyala-yapıştır sahteciliği tespiti için yapılan bu çalışmada tez kapsamında önerilen önceki çalışmalardan farklı olarak blok tabanlı yöntemlerin kullanılması yerine anahtar tabanlı bir yöntem kullanılarak blok yerine görüntünün tamamından özellik çıkarılmıştır [151]. Bu amaçla çalışmada anahtar nokta tabanlı yöntemlerden renkli SIFT kullanılarak sahte görüntünün her bir kanalı için anahtar noktalar çıkarılmıştır. Çalışmada SIFT ve renkli SIFT yöntemleri arasında karşılaştırma yapılmıştır. Karşılaştırma sonucunda renkli SIFT yöntemi ile SIFT yöntemine göre daha fazla eşleşen nokta elde edilerek daha etkin sahtecilik tespiti yapıldığı görülmüştür. Bununla birlikte önerilen yöntemin, kopyalanan bölgenin döndürülme, ölçekleme, görüntünün JPEG sıkıştırmaya tabi tutulmasında dahi tekrarlanan bölgeleri tespit edebildiği sonuçlarda gösterilmiştir.

6. Tez kapsamında anahtar nokta tabanlı önerilen bir başka yöntemde kopyala – yapıştır sahteciliğinin tespitinde LPQ ve SIFT yöntemlerinden faydalanmaktadır [152]. Yöntemde anahtar nokta çıkarımından önce LPQ yöntemiyle doku çıkarılmıştır. SIFT yönteminden LPQ ile elde edilen doku görüntüsünden anahtar nokta çıkarmada faydalanılmıştır. Anahtar noktaların eşleştirilmesi ile sahte bölge tespit edilmektedir. Çalışmada anahtar nokta çıkarımından önce doku çıkarımının yapılması ile anahtar nokta tabanlı pasif görüntü doğrulama yöntemleri iyileştirilmiştir. Elde edilen sonuçlar yöntemin son işlem operasyonlarına da dayanıklı olduğunu göstermektedir.

7. Görüntü için önerilen Kopyala-yapıştır sahtecilikleri tespiti dışında, medikal görüntü doğrulaması çalışma kapsamında gerçekleştirilmiştir [153]. Önerilen yöntem literatürde ilk olarak medikal görüntülerde oynanmış bölge tespitinde pasif görüntü doğrulama yöntemlerinden faydalanmaktadır. Yöntemde anahtar nokta çıkarım yöntemlerini daha başarılı hale getirebilmek için medikal görüntüden yapısal doku bilgisi çıkarılmaktadır. Doku bilgisinin çıkarımında rotasyondan bağımsız Yerel İkili Örüntüler (Rotation Invariant Local Binary Patterns-LBPROT) operatöründen faydalanılırken anahtar nokta çıkarımında SIFT kullanılmıştır. Deneysel sonuçlardan görüldüğü üzere

önerilen yöntem medikal görüntüler çeşitli ataklara uğrasa (Gauss bulanıklaştırma, Beyaz Gauss gürültüsü) ve tekrarlanacak bölge yapıştırılmadan önce ölçekleme ve rotasyon işlemi (ara işlem operasyonları) uygulansa dahi sahte bölgeyi tespit edebilmektedir.

b) Video alanında yapılan çalışmalar

1. Tez kapsamında görüntü sahteciliklerinin yanında video sahteciliklerinin tespiti üzerinde de durulmuştur. Video için Yerel Fark İkiliğini (Local Difference Binary-LDB) kullanan yeni bir Çerçeve Tekrarlama sahteciliği tespit yöntemi önerilmiştir. Yöntemde sahteciliğin uzaklığını tahmin etmek ve tekrarlanmış çerçevelerin tam yerini belirlemek için benzer özellik vektörlerine sahip benzer çerçeveler arasındaki uzaklık kullanılmıştır. Benzer çerçeveler arasındaki Tepe Sinyal Gürültü Oranı (Peak Signal Noise Ratio-PSNR), bu uzaklıkları üç sınıfa ayırtmak için kullanılmakta ve kural tabanlı mekanizma üye olduğu sınıfa göre sahte çerçeveleri belirlemektedir. Elde edilen sonuçlar önerilen yöntemin diğer çalışmalara göre daha düşük çalışma zamanına ve daha yüksek doğruluğa sahip olduğunu göstermektedir.

2. Çerçeve Tekrarlama sahteciliği tespiti için literatürde birçok çalışma önerilmiştir. Ancak bu yöntemlerde, herhangi bir çerçeve grubunun kopyalanarak bu grubun ayna yansıma şeklinin yine aynı video içerisinde herhangi bir yere yapıştırılması ile oluşturulan çerçeve yansıması atağı göz önünde bulundurulmamıştır. Önerilen çalışmada yeni bir Çerçeve Tekrarlama/Yansıma sahteciliği tespit yöntemi geliştirilmiştir [160]. Yöntemde çerçevelerden ikili özellikler çıkarılmakta ve bu özellikler arasındaki benzerlik belirlenmektedir. PSNR kullanılarak çok sayıda aday elimine edilip, sahte çerçevelerin tespiti iyileştirilmektedir. Önerilen yöntem benzer çalışmalarla kıyaslandığında daha iyi çalışma zamanına ve tespit doğruluğuna sahiptir.

3. Çerçeve Tekrarlama sahteciliğinin tespiti için önerilen bir başka çalışmada Sözcük Torbası (Bag of Words-BoW) modeline dayalı yeni bir sahtecilik tespit yöntemi önerilmiştir. Çalışmada BoW görsel sözcükler üretmek ve video çerçevelerinden SIFT ile çıkarılmış anahtar noktalarından bir sözlük inşa edebilmek için kullanılmıştır. Çerçeve özelliklerinden yani çerçevelerdeki anahtar noktalardan elde edilen görsel sözcüklerden video içerisindeki tekrar eden bölümlerin tespiti için faydalanılmıştır. Yöntem yüksek başarı oranı ve düşük çalışma zamanına sahiptir.

4. Tez kapsamında videoda Çerçeve Tekrarlama sahteciliği dışında bölgesel tabanlı sahtecilik de tespit edilmiştir. Video çerçevelerinde bulanık kenarların tespiti ile bölgesel sahtecilik izleri belirlenmektedir. Önerilen yöntem videodaki hem Kopyala-yapıştır hem de

Video Birleştirme sahteciliğini tespit edebilmektedir. Aynı zamanda yöntem sahtecilik tespiti izlerinden faydalandığı için Ayna simetrik Kopyala-yapıştır sahteciliğine dayanıklıdır. Sonuçlar, geliştirilen yöntemin literatürdeki yöntemlere kıyasla doğrulama oranı açısından daha iyi sonuçlar ürettiğini göstermektedir.

5. Video sahteciliklerin tespiti için geliştirilen yöntemler tek bir çatı altında toplanarak, Pasif Video Doğrulama Sistemi isimli bir yazılım da tez kapsamında geliştirilmiştir.

Bu tez çalışması süresince yapılan çalışmalar Matlab simülasyon ortamı kullanılarak denenmiş ve önerilen teknikler ve karşılaştırma yapılan çalışmalar hazır paketler olmaksızın tarafımızdan kodlanarak gerçekleştirilmiştir.

2.1. Görüntü Üzerinde Yapılan Çalışmalar

Yukarıda maddeler halinde verilmiş çalışmalardan görüntü üzerinde yapılan çalışmaların detayları sırasıyla ilerleyen bölümlerde verilmektedir.

2.1.1. CLD ve Renk Momentlerine Dayalı Kopyala-Yapıştır Sahteciliği Tespiti

Kopyala-yapıştır sahteciliğinin tespitinde önerilen AKD tabanlı yöntemler, AKD katsayılarından örtüşen blokları temsil etmekte faydalanmıştır. Kopyala-yapıştır sahteciliği tespiti için yapılan ilk çalışmada Fridrich vd. tarafından blokların özelliklerini çıkarmada AKD yöntemi kullanılmıştır. Huang vd. önerilen bu ilk çalışmanın hızını arttırmayı hedeflemişlerdir [72]. Bu amaçla çalışmada AKD katsayıları önceden belirlenmiş bir q değeri ile kuantalanmaktadır. Ayrıca yöntemde komşu özellik vektörlerinin arasındaki benzerliğin tespiti için yeni bir yöntem önerilmiştir. Önerilen yöntem bazı son işlem operasyonlarına dayanıklıdır. Cao vd. ise özellik vektörlerinin boyutunu azaltabilmek için AKD katsayılarının ortalamasını almıştır [12]. AKD katsayılarının ortalama değeri son işlem operasyonları sonucunda fazla değişmediğinden yöntem son işlem operasyonlarında oldukça iyidir. Çalışmada boyut indirgemesi sonucunda elde edilen özellik vektörü boyutu 1×4 'tür. 2013 yılında AKD ve SVD yöntemlerine dayalı bir başka çalışma da Zhao vd. tarafından yapılmıştır [33]. Çalışmada örtüşen blokların herbirine AKD uygulanmıştır. Blok temsilini kuvvetlendirmek için AKD değerleri önceden belirli bir matrisin değerleri

ile kuantalanmıştır. Sonrasında her quantalanan blok örtüşmeyen alt bloklara ayrılmış ve bu bloklara SVD uygulanmıştır. Yine rapor edilen sonuçlara göre yöntem son işlem operasyonlarına dayanıklıdır. Bu çalışmada bahsedilen AKD tabanlı bu yöntemlerin son işleme operasyonlarına karşı sağlamlığını artırmak için Renk Momentleri ve Renk Düzeni Tanımlayıcılarına dayanan yeni bir yöntem önerilmektedir [142]. Literatürde ilk kez Kopyala-yapıştır sahteliği tespitinde Renk Momentleri ve Renk Düzeni tanımlayıcısından yararlanılmıştır. Renk Momentleri ve Renk Düzeni Tanımlayıcısı yöntemleri aşağıda detaylandırılacaktır.

2.1.1.1. Renk Momentleri

Son yıllarda İçerik Tabanlı Görüntü Erişim (Content Based Image Retrieval) sistemleri araştırmacılar arasında oldukça popüler hale gelmiştir. Bu sistemler sayesinde kullanıcılar görüntü ve video veri tabanlarını araştırıp tarayabilmektedir. Birçok görüntü erişim sistemi sorgu görüntüsünün içeriğine benzer tüm görüntü içeriklerine erişmek için renk, doku ve şekil gibi görsel içerikleri kullanmaktadır. Görüntü erişimi için araştırmacılar tarafından en çok kullanılan özellik renk özelliğidir. Konum ve yöndeki değişimlere karşı dayanıklı olan renk histogramları görüntü veri tabanlarını filtrelemek için birçok yöntem tarafından kullanılmaktadır. Bununla birlikte, renk bölgelerinin konumsal ilişki bilgisini histogram içermeyeceği için görüntüleri ayırt etme amacıyla kullanılamamaktadır. Renk Korelogram ve Renk Uyum vektörleri ise yerel renklerin genel dağılımlarını kullanmaktadır. Bu yöntemler görüntü erişiminde kullanıldığında renk histogramlarına kıyasla daha iyi sonuç vermekte ancak hesapsal karmaşıklıkları bu yöntemlerin en büyük dezavantajını oluşturmaktadır. Stricker ve ark. Renk Momentlerinin görüntülerin renk dağılımlarını karakterize etmede kullanıldığını göstermiştir [143]. Görüntünün üç renk kanalının ilk üç momenti görüntüleri ayırt etmede renk özelliğine dayalı bir ölçüt olarak kullanılmaktadır. Bu momentler görüntüler arasındaki renk benzerliği hakkında bilgi vermektedir.

Stricker ve ark. bir görüntünün üç merkezi momentini (ortalama, standart sapma ve çarpıklık) kullanmışlardır. Çalışmalarında, renk momentlerinin histogram tabanlı yöntemlerden daha dayanıklı ve daha hızlı olduğunu göstermişlerdir. Momentler renkli görüntünün her bir kanalı için hesaplanmaktadır. Görüntüdeki piksel sayısı N , i . renk

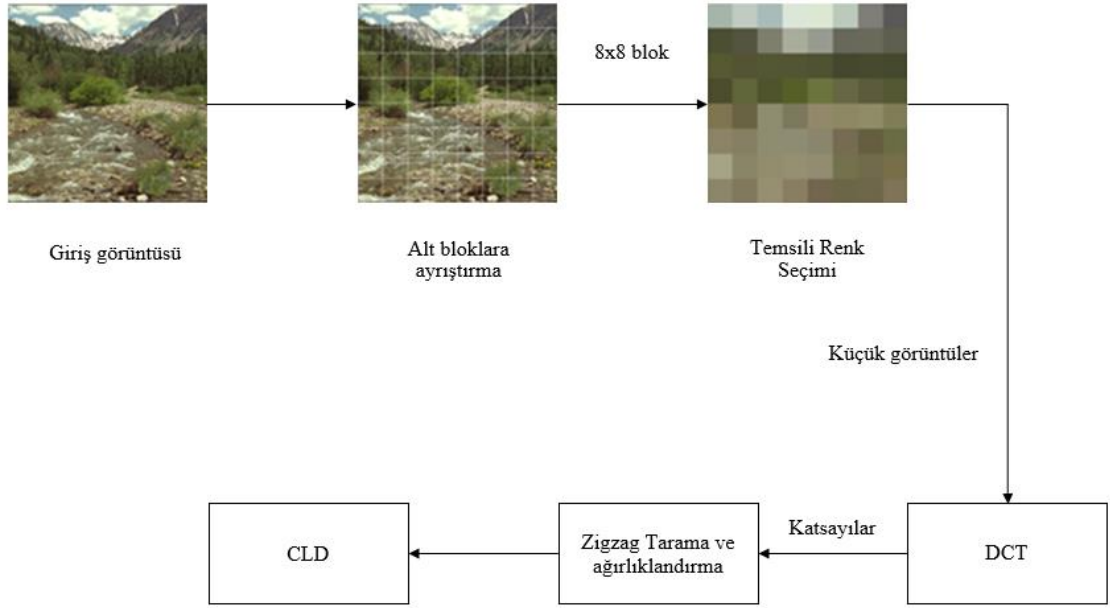
kanalının j . pikseli p_{ij} olmak üzere ilk üç renk momenti (2.1)'de verildiği gibi hesaplanmaktadır.

$$E_i = \sum_{j=1}^N \frac{1}{N} p_{ij} \quad \sigma_i = \sqrt{\left(\frac{1}{N} \sum_{j=1}^N (p_{ij} - E_i)^2 \right)} \quad s_i = \sqrt[3]{\frac{1}{N} \sum_{j=1}^N (p_{ij} - E_i)^3} \quad (2.1)$$

İlk renk momenti E_i görüntüdeki ortalama renk değerini vermektedir. Standart sapmanın karekökü ikinci renk momentini σ_i vermektedir. Üçüncü renk momenti s_i ise dağılımın asimetriklik derecesi hakkında bilgi vermektedir. Önerilen yöntemde Renk momentleri alt blokları renk dağılımına göre sınıflandırmak için kullanılmaktadır.

2.1.1.2. Renk Düzeni Tanımlayıcısı-CLD

Günümüzde, hem kişisel hem de ticari ortamda, multimedya verileri hızlı bir oranda artmaktadır. Bu yüzden bu bilgilere erişim gittikçe zorlaşmaktadır. Bu amaçla içerik tabanlı erişim sistemleri önerilmiş ve bu sistemlerin kullanabileceği bazı görsel tanımlayıcılar belirlenmiştir. Ancak, yine de bu sistemler için herhangi bir standart bulunmamakta ve her sistem kendi seçtiği görsel tanımlayıcıyı kendi seçtiği depolama şekli ile kullanmaktadır. MPEG-7 bu eksikliği gidermek için MPEG grubu tarafından önerilmiş bir standarttır. Görsel ve sessel içeriği tanımlamak amacıyla metotlar ve araçlar kümesi tanımlanmaktadır. Renk Düzeni Tanımlayıcısı-CLD MPEG standartındaki görsel tanımlayıcıların bir tanesidir. CLD görüntüdeki renklerin uzaysal dağılımını temsil etmekte yüksek hızlı görüntü erişimi sağlamaktadır. Benzerliğe dayalı erişimler gibi birçok teknik, özellik çıkarımı için bu tanımlayıcıyı kullanmaktadır. Bir görüntüden CLD vektörünün elde edilmesi dört aşamadan oluşmaktadır. Şekil 2.1.'de görüldüğü üzere öncelikli olarak görüntü alt bloklara ayrıştırılmakta ve temsili renk seçimi yapılmaktadır. Standart bir alt bloğun piksel değerlerinin ortalaması o alt bloğa karşılık gelen temsili renk değeri olarak kullanılmaktadır. Her bir bloğun temsili renkleri giriş görüntüsünün 8x8'lik küçük (tiny) görüntülerini oluşturmaktadır. Ardından küçük görüntüler RGB renk uzayından YCbCr renk uzayına izdüşürülmekte ve küçük görüntünün her bir kanalı (Y, Cb, Cr) AKD kullanılarak frekans uzayına izdüşürülmektedir. Son aşama olarak ise AKD katsayılarından oluşan matrislere ayrı ayrı zigzag tarama uygulanmaktadır. Zigzag tarama sonucunda 1x64 boyutunda elde edilen bu üç vektör görüntünün CLD'sini temsil etmektedir.



Şekil 2.1. Görüntüden CLD vektörünün elde edilişi

Renk Momentleri ve CLD'nin birlikte kullanıldığı çalışmada ilk aşama olarak görüntü örtüşen sabit büyüklükteki alt bloklara ayrılır. Ardından Renk momentleri kullanılarak bloklar renk benzerliklerine göre sınıflandırılır. Oluşan her bir sınıf özelliklerin eşleşme aşamasında işlem zamanını iyileştirmek için iş parçacıklarına dağıtılır ve her iş parçacığı kendine düşen sınıfın bloklarının özellik çıkarımında CLD'yi kullanır. Alt bloklarlardan elde edilen CLD'ler arasındaki benzerlik sahtecilik hakkında bilgi verecektir. En son aşamada ise iş parçacıkları tarafından çiftlenmiş bölgeler işaretlenir. Önerilen yöntemin akış diyagramı Şekil 2.2.'de görülmektedir. Yöntemi adım adım detaylandırarak olursak;

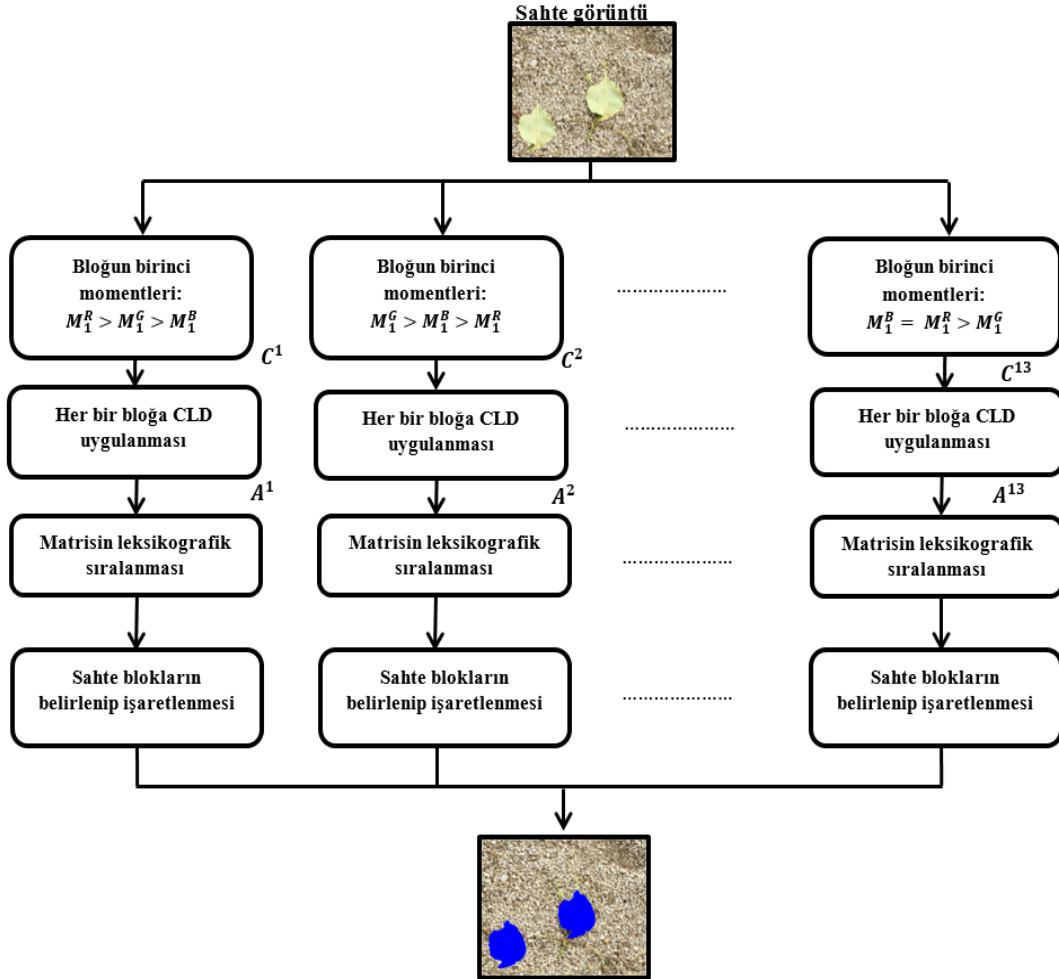
Adım 1: $M \times N \times 3$ boyutunda şüpheli renkli görüntü I olsun. I , $B \times B$ boyutlu örtüşen alt bloklara ayrılır. Her bir blok B_{ij} ile gösterilmek üzere i ve j bloğun başlangıç noktasıdır. Görüntü üzerinde $B \times B$ 'lik pencere en üst sol köşeden en alt sağ köşeye kadar bir piksel ara ile kaydırılarak örtüşen bloklar elde edilir. Bu kaydırma işlemi sonucunda toplam $(M-B+1) \times (N-B+1)$ blok elde edilmektedir.

Adım 2: Orijinal görüntü blokları $B_1, B_2, \dots, B_{N_{\text{block}}}$ olsun. Görüntü renkli görüntü olduğu için her bir blok üç kanala sahiptir. Her bir bloğun üç kanalı içinde birinci moment hesaplanarak bloklar sınıflandırılır. Üç kanalın birinci momentlerinin birbirleri arasındaki

durumları sonucu 13 farklı sınıf oluşacaktır. Tablo 2.1 momentler arasındaki durumlar sonucu oluşan bu 13 sınıfı göstermektedir.

Tablo 2.1. Birinci momentlere göre sınıflandırma

Sınıf No	Üç Kanalı Moment Değerleri	Sınıf No	Üç Kanalı Moment Değerleri
1	$M_1^R > M_1^G > M_1^B$	8	$M_1^R = M_1^G = M_1^B$
2	$M_1^G > M_1^B > M_1^R$	9	$M_1^B = M_1^G > M_1^R$
3	$M_1^B > M_1^G > M_1^R$	10	$M_1^R > M_1^G = M_1^B$
4	$M_1^B > M_1^R > M_1^G$	11	$M_1^B = M_1^R > M_1^G$
5	$M_1^G > M_1^R > M_1^B$	12	$M_1^G > M_1^B = M_1^R$
6	$M_1^R > M_1^B > M_1^G$	13	$M_1^B = M_1^R > M_1^G$
7	$M_1^B > M_1^G = M_1^R$		



Şekil 2.2. Önerilen yöntemin akış diyagramı

Adım 3: Sınıflandırma sonucunda elde edilen 13 sınıf $C^1, \dots, C^{13}$ olsun. Her bir sınıf iş parçacıkları tarafından ayrı olarak işlenmekte ve sonuçta 13 özellik vektör matrisi $(A^1, \dots, A^{13})$ oluşmaktadır. İş parçacıkları kendine ait blokların CLD özelliklerini çıkarmaktadır. Her bir bloğun CLD çıkarımı yukarıda anlatıldığı gibi gerçekleştirilmektedir. CLD çıkarımında bloğun üç kanalı içinde AKD uygulanıp zigzag tarama sonucunda elde edilen tüm katsayılar alınmamakta sadece her üç kanal için de ilk $[p \times B^2]$ lik kısım özellik vektörünü oluşturmaktadır. p , $(0 < p < 1)$ şeklinde tanımlanmış bir faktördür. Bunun sonucunda $A^1, \dots, A^{13}$ matrislerinin satırlarında tuttuğu blokların özellik vektör boyutu $(1 \times 3pB^2)$ olmaktadır. (2.2)'de t . sınıfın n . özellik vektörü verilmektedir.

$$V^{tn} = (V_1^{tn}, V_2^{tn}, \dots, V_{3pB^2}^{tn}) \quad (2.2)$$

Yöntemi JPEG sıkıştırma ataklarına karşı dayanıklı hale getirmek için q kuantalama parametresi kullanılmaktadır. Tüm özellik vektörleri (2.3)'te verildiği gibi q ile kuantalanıp en yakın sayıya yuvarlanmaktadır.

$$V_i^{tn} = \left(\left\lceil \frac{V_i^{tn}}{q} \right\rceil \right), \quad i \in \{1 \dots 3pB^2\}, \quad t \in \{1 \dots 13\} \quad (2.3)$$

Adım 4: Tüm sınıflardan özellik vektörleri elde edildikten sonra özellik vektörlerinin eşleştirilmesi gerekmektedir. Bunun için $A^1, \dots, A^{13}$ özellik vektör matrisleri birbirinden bağımsız olarak leksikografik olarak sıralanır. Her matrisdeki toplam özellik vektör sayısı orijinal görüntü karakteristiğine göre değişmektedir. t . sınıftaki özellik vektör sayısı S olmak üzere A^t matrisi (2.4)'te verildiği gibidir.

$$A^t = \begin{bmatrix} V^{t1} \\ V^{t2} \\ \vdots \\ V^{tS} \end{bmatrix} \quad (2.4)$$

Vektörler arasındaki benzerlik, Öklid uzaklığı kullanılarak araştırılır. Benzerliğe iki vektör arasındaki uzaklığın önceden belirlenen bir eşik değeri ile kıyaslanması ile bakılmaktadır. t_s eşik değer olmak üzere t . sınıfın iki vektörü arasındaki benzerliğe

(2.5)'deki gibi bakılmaktadır. Benzerlik araştırmasında bir vektör tüm vektörlerle değil kendisinden sonraki t_{nb} sayıdaki komşu vektörle kıyaslanmaktadır. Bu işlem eşleştirme işleminin süresini oldukça etkilemektedir. t_{nb} değeri arttığı zaman algoritmanın koşma zamanı artarken bununla birlikte doğruluk oranı da artabilmekte azaldığı zaman ise eşleşme süreci kısa sürmekle birlikte doğruluk oranı azalabilmektedir.

$$\sqrt{\sum_{i=1}^{3pB^2} (V_i^{tn} - V_i^{tm})^2} \leq t_s \quad (2.5)$$

Sadece komşu vektörler arasındaki benzerliğe bakılması sahtecilik tespiti için yeterli olmamaktadır. Bloklar arasındaki uzaklık belirlenen eşik değeri t_d den büyük olmasına bakılması düzgün geçişsiz bölgelerin sahte olarak algılanmasına engel olmaktadır. (x_1, y_1) ve (x_2, y_2) benzer çift olarak bulunmuş blokların sol üst köşe koordinatları olsun. Bu bloklar arasındaki Öklid uzaklığı bu blokların düzgün bir bölgeye ait olma olasılığını gösterecektir. Önerilen yöntem kopyala yapıştır yapılmış bölgelerin birbirinden en az t_d kadar uzakta olduğunu varsaymaktadır. Bloklar arasındaki uzaklığın değerlendirilmesi (2.6)'da verildiği gibidir.

$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} > t_d \quad (2.6)$$

Eğer iki vektör (2.5) ve (2.6)'daki kuralları sağlarsa iki vektöre karşılık gelen bloklar arasındaki shift vektörü hesaplanmaktadır. (x_1, y_1) ve (x_2, y_2) benzer görüntü bloklarının sol üst köşe koordinatlarını temsil etmektedir. Bu bloklar arasındaki shift vektörü (2.7)'deki gibi hesaplanmaktadır.

$$S = (s_1, s_2) = (x_1 - x_2, y_1 - y_2) \quad (2.7)$$

Adım 5: Son aşama olarak aynı shift vektörleri sayısı değerlendirilir. Eğer aynı sift vektör sayısı t_{sc} den büyükse o shift vektörünü oluşturan bloklar sahte olarak işaretlenir.

2.1.2. Eşik Değerinin Otomatik Olarak Belirlendiği Kopyala-Yapıştır Sahteciliği Tespiti

Literatürde kopyala-yapıştır sahteciliğinin tespiti için önerilmiş yöntemler görüntüyü örtüşen alt bloklara bölmekte ve blokların özellik çıkarımında çeşitli yöntemler

kullanılmaktadır. Çıkarılan özellik vektörleri arasındaki benzerlik ile sahtecilik tespit edilmektedir. Ancak bu yöntemler benzerlik araştırmasında önceden belirlenmiş bir eşik değerini kullanmaktadır [12, 14, 33, 35, 70, 72]. Oysaki farklı test görüntüleri farklı eşik değerleri gerektirmektedir. Özellik vektörü elemanlarının aralığı önceden belirlenemediği için en iyi eşik değerin belirlenmesi başka bir problemidir. Bu çalışmada AKD-faz terimleri kullanılarak özellik vektörleri elemanlarının değer aralığı sınırlanmış ve Benford'un Genelleştirilmiş yasası kullanılarak test görüntüsünün sıkıştırma geçmişi hakkında bilgi edinilmiştir [144]. Yöntem ayrıca Öklid uzaklığı veya çapraz korelasyon yerine özellik vektörleri arasındaki eleman bazlı eşitliği ve geçerli test görüntüsünün eşik değerini otomatik olarak belirlemek için sıkıştırma geçmişini kullanır. Yine Kopyala-yapıştır sahteciliği tespitinde AKD-faz ve AKD Uzayında Enerji Olasılığı literatürde ilk kez kullanılmıştır. Bu yöntemler aşağıda detaylandırılacaktır.

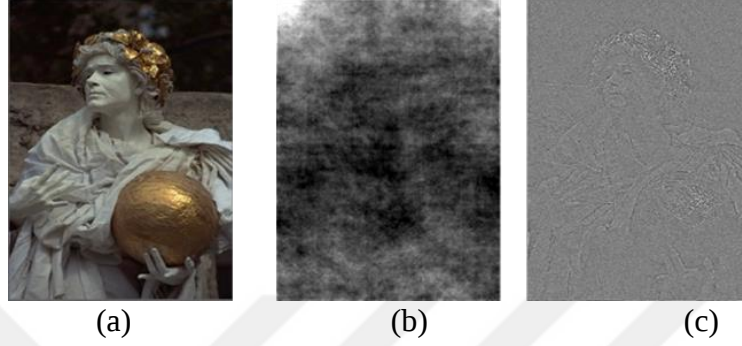
2.1.2.1. AKD Faz

Görüntü sahteciliğinin tespitinde AKD faz yöntemi özellik çıkarımı aşamasında kullanılmaktadır. Bracamonte vd. bir görüntünün AKD fazının $\{-1, 1\}$ olmak üzere iki değerli piksel içermesine karşın, görüntüye ait oldukça büyük bir bilgi temsil ettiğini göstermiştir [145]. $N \times M$ büyüklüğündeki I görüntüsü, bu görüntünün AKD'sinin mutlak değeri $|I_{DCT}|$ ve orijinal I görüntüsüne karşılık gelen faz terimi I'_{DCT} (2.8)'de verildiği gibi hesaplanmaktadır.

$$I = |I_{AKD}| I'_{AKD} \quad (2.8)$$

Görüntünün fazı, görüntünün AKD katsayı değerlerinin işaret bilgisini içermektedir. (2.8)'de verilen mutlak değer AKD katsayılarının genliğine karşılık gelmektedir. Mutlak değeri alınmış AKD görüntüsünün, $|I_{DCT}|$, ters AKD'si AKD genlikli görüntüyü oluştururken, AKD fazı alınmış görüntünün ters dönüşümü ise AKD faz görüntüsünü oluşturmaktadır. AKD faz teriminin etkisini göstermek üzere Şekil 2.3.(a)'da verilen görüntünün Şekil 2.3.(b)'de mutlak AKD görüntüsü ve Şekil 2.3.(c)'de AKD faz görüntüsü verilmiştir. Şekillerden de görüldüğü üzere Şekil 2.3.(c)'de verilen AKD faz görüntüsünün kenar bilgisini içermektedir.

2012 yılında Ayyalasomayajula vd. görüntü erişiminde (image retrieval) üçlü AKD faz yöntemini önermişlerdir [146]. Önerdikleri yöntemde JPEG sıkıştırmadan sonraki herhangi bir görüntü birçok sıfır değerli AKD katsayısı içereceği için, AKD faz terimleri $\{-1, 0, 1\}$ kümesinden değer almaktadır. Sonuç olarak, çalışmada sıfır değerli katsayıların değerlendirilmesinin önemi rapor edilmektedir.



Şekil 2.3. (a) Orijinal görüntü (b) Genlik görüntüsü (c) Faz görüntüsü

Çalışmada örtüşen alt blokların her biri için AKD faz terimleri hesaplanarak özellik vektörleri çıkarılmıştır. Bu sayede her bir bloğu temsil eden özellik vektörleri değerleri önceden belirlenebilmiş ve sadece $\{-1, 0, 1\}$ değerlerini almıştır. Özellik vektörü değerlerinin önceden bilinerek sınırlı bir aralık içinde olması benzer özellik vektörlerinin araştırılması aşamasını oldukça kolay hale getirmiştir.

2.1.2.2. AKD Uzayında Enerji Olasılığı

Görüntü sahteciliğinin tespitinde AKD uzayında enerji olasılığından özellik çıkarımında faydalanılmaktadır. $M \times N$ boyutundaki görüntünün gri seviye matrisi A_{mn} olmak üzere, görüntünün AKD'si $F(p, q)$, (2.9)'da verilmiş olan bağıntılara göre hesaplanmaktadır.

$$F(p, q) = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos \frac{\pi(2m+1)p}{2M} \cos \frac{\pi(2n+1)q}{2N}$$

$$0 \leq p \leq M - 1, \quad 0 \leq q \leq N - 1$$

$$\alpha_p = \begin{cases} \frac{1}{\sqrt{M}}, & p = 0 \\ \frac{\sqrt{2}}{M}, & 1 \leq p \leq M - 1 \end{cases} \quad \alpha_q = \begin{cases} \frac{1}{\sqrt{N}}, & q = 0 \\ \frac{\sqrt{2}}{N}, & 1 \leq q \leq N - 1 \end{cases} \quad (2.9)$$

AKD katsayıları kullanılarak E toplamsal enerji değeri (2.10)'daki ifade ile hesaplanır. Toplamsal enerji değeri, bloğa ilişkin enerji olasılığı değerlerini elde etmede kullanılmaktadır. EP(p,q) enerji olasılık değeri ise (2.11)'deki ifade kullanılarak elde edilir.

$$E = \sum_{p=1}^M \sum_{q=1}^N (F_{pq})^2 \quad (2.10)$$

$$EP(p, q) = \frac{(F_{pq})^2}{E} \quad (2.11)$$

AKD faz değerlerini kullanarak otomatik eşik belirleyen çalışmada önerilen yöntem dört aşamadan oluşmaktadır: Özellik çıkarımı, Benzerlik araştırması, Eşik değerinin belirlenmesi ve Sahte bölgelerin işaretlenmesi.

Özellik Çıkarımı: Şüpheli görüntü YCbCr uzayına dönüştürülmekte ve birbiri ile örtüşen alt bloklara ayrıştırılmaktadır. Her bir alt blok için AKD faz alınarak oluşturulmuş işaret blokları, zigzag taranmakta ve belirli sayıdaki elementin alınmasıyla her bir blok için özellik vektörü oluşturulmaktadır. Sonrasında oluşturulan özellik vektörleri leksikografik sıralanmakta ve bir matriste tutulmaktadır.

Benzerlik Araştırması: Yöntemde benzer özelliklerin bulunması için özellik vektörleri arasında karşılıklı elemanlara bakılmaktadır. Özellik vektörlerinin temsil ettiği blokların aynı olup olmadığına bir eşik değeri kullanılarak karar verilmektedir. Bu eşik değeri görüntünün karakteristik özelliğine göre otomatik olarak belirlenmektedir. Bunun sonucunda eğer iki vektör benzer çıkarsa bu vektörlere karşılık gelen shift vektör hesaplanıp, kaydedilmektedir.

Eşik Değerinin Belirlenmesi: Eşik değerinin belirlenmesinde öncelikli olarak görüntünün sıkıştırılıp sıkıştırılmadığına bakılmaktadır. Eğer sıkıştırılmışsa, sıkıştırmada kullanılan kalite faktörü hesaplanmaktadır. Eşik değeri de kalite faktörüne göre seçilmektedir.

Sahte bölgelerin işaretlenmesi: Eğer kaydedilen shift vektörleri sayısı belirli bir eşik değerini geçerse, bu vektörlerin karşılık geldiği bölgeler yöntem tarafından sahte olarak işaretlenmektedir.

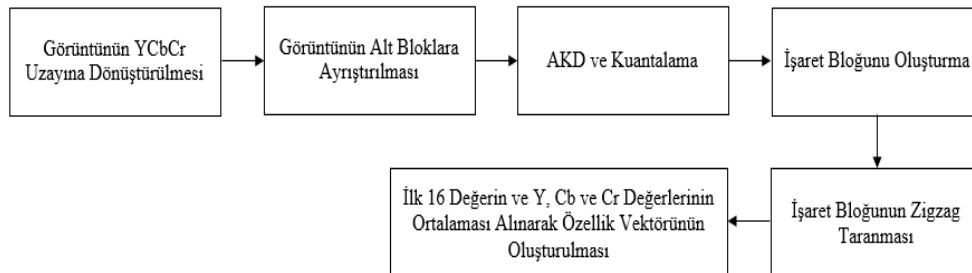
Bu dört aşamanın detayları aşağıda verilmektedir.

Özellik Çıkarımı: NxM boyutundaki I görüntüsü YCbCr uzayına dönüştürülmekte ve 8x8 büyüklüğündeki birbiri ile örtüşen alt bloklara ayrıştırılmaktadır. Her bir blok $B^i, i = 1 \dots (N - 7)(M - 7)$ AKD ile frekans uzayına dönüştürülür ve elde edilen katsayılar qt değeri ile kuantalanır. Böylece her bir blok değeri, $B^i = \lfloor \text{DCT}(B^i) / qt \rfloor$ kullanılarak yeniden hesaplanır. Frekans uzayındaki katsayıların işaretleri (2.12)'de verildiği gibi hesaplanarak B^i 'den işaret bloğu S^i oluşturulur. i. blok j. satır ve k. sutundaki katsayı B_{jk}^i olmak üzere S^i 'nin değerleri $i = 1 \dots (N - 7)(M - 7)$ B^i 'ye karşılık gelen faz terimlerini göstermektedir.

$$\begin{aligned} \forall B_{jk}^i > 0 &\Rightarrow S_{jk}^i = 1 \\ \forall B_{jk}^i < 0 &\Rightarrow S_{jk}^i = -1 \\ \forall B_{jk}^i = 0 &\Rightarrow S_{jk}^i = 0 \end{aligned} \quad (2.12)$$

Düşük frekanslı AKD katsayılarının JPEG ataklara karşı daha dayanıklı olması sebebiyle, düşük frekanslı katsayıların işaret bilgisini çıkarabilmek için işaret bloğu zigzag taranır. Zigzag taranmış vektörün ilk 16 değeri alınarak i. bloğa karşılık gelen V^i vektörü oluşturulur.

Ayrıca her bir blok için Y, Cb ve Cr değerlerinin ortalaması alınarak özellik çıkarılmaktadır. Bu üç ortalama değerler yukarıda oluşturulan 16 elemanlı özellik vektörü sonuna eklenerek her bir blok için 19 elemanlı bir özellik vektörü elde edilmiş olmaktadır. Eğer iki vektör farklı renk karakteristiklerine sahipse diğer özelliklere bakılmasına ihtiyaç kalmayacaktır. Bloklar için çıkarılan renk bilgisi benzerlik araştırması sırasında karşılaştırma sayısını oldukça düşürmektedir. Özellik çıkarma yönteminin adımları Şekil 2.4.'te şematize edilmiştir.



Şekil 2.4. Önerilen Özellik Çıkarma Yöntemi Aşamaları

Özellik çıkarımı sonucunda $[(N - 7) \times (M - 7), 19]$ boyutundaki alt bloklara karşılık gelen faz terimleri hakkında bilgi içeren A matrisi oluşturulur. A matrisindeki benzer vektörleri bir araya getirmek adına matris leksikografik olarak sıralanmaktadır. Sıralama işleminden sonra vektörler arasındaki benzerlik sahteciliğin tespitinde değerlendirilecektir.

Benzerlik Araştırması: A matrisindeki her bir vektör t_f kadar özellik vektörü ile kıyaslanacaktır. İki vektör arasındaki benzerlik oranı bu vektörlerin içeriklerine göre hesaplanmaktadır. Her bir özellik vektörü 19 eleman içermektedir. Bu elemanlardan ilk 16'sı $\{-1, 0, +1\}$ değerlerinden birini almaktadır. V^i ve V^j 'nin kıyaslanacak iki vektör olduğu varsayılırsa ilk kriter olarak bu vektörlerin temsil ettiği blokların renk karakteristiklerinin uyumlu olması gerekmektedir. Bunun için vektörlerin karşılıklı son üç elemanları arasındaki fark belirli bir eşik değerinden küçükse $(|V_k^i - V_k^j| \leq t_c)$, $k = 17 \dots 19$, vektörlerin ilk 16 elemanı (2.13)'te verildiği gibi birebir çarpılır aksi takdirde diğer vektörlerin değerlendirilmesine geçilir.

$$Z_k^{ij} = V_k^i \cdot V_k^j, \quad k = 1 \dots, 16 \quad Z_k^{ij} \in \{-1, 0, 1\} \quad (2.13)$$

Z^{ij} vektörü karşılıklı elemanların benzerliği hakkında bilgi içermektedir. Benzerlik oranı r^{ij} , (2.14)'te verildiği gibi Z vektörüne ve vektör değerlerine göre hesaplanmaktadır.

$$\forall (Z_k^{ij} == 1) \vee (V_k^i == 0 \wedge V_k^j == 0) \Rightarrow d = d + 1, \quad k = 1 \dots 16$$

$$r^{ij} = d/16 \quad (2.14)$$

İki vektör arasındaki benzerlik oranı t_s den büyükse ve bu vektörlerin temsil ettiği bloklar arası uzaklık belirlenen t_d eşik değerinden büyük ise (2.15)'te verildiği gibi bloklara karşılık gelen shift vektörleri sh hesaplanır ve bir listeye eklenir.

$$\forall \left(\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \geq t_d \right) \wedge (r^{ij} \geq t_s) \Rightarrow sh_m = (x_i, y_i, x_j, y_j, |x_i - x_j|, |y_i - y_j|) \quad (2.15)$$

Listenin ilk dört elemanı (x_i, y_i, x_j, y_j) , karşılık gelen blokların sol üst köşe koordinatlarını, son iki eleman ise ilgili blokların shift vektörünü temsil etmektedir. Benzerlik oranı r^{ij} iki vektörün karşılıklı eşit değerli elemanlarının yüzdesini göstermektedir. Örneğin iki vektörün benzerlik oranı %87 ise iki vektör arasında karşılıklı 16 elemanın 14 tanesi eşit demektir. Literatürdeki çalışmaların çoğunda benzerlik oranı belirli bir eşik değeri ile kıyaslanmaktadır (Benzerlik ölçütü olarak genellikle Öklid uzaklığı kullanılmaktadır). Ve bu eşik değerinin optimum olarak belirlenmesi için de birçok test gerçekleştirilmektedir. Önerilen yöntemin avantajı benzer çalışmalarla kıyaslandığında burada görülmektedir. Aşağıda bu avantaj daha da detaylandırılacaktır.

Önerilen yöntem üç değerli özellik vektörleri üretmektedir. Literatürdeki yöntemler bloklardan özellik çıkarımında uzaysal veya frekans tabanlı yaklaşımları kullanmakla birlikte ürettikleri özellik vektörlerinin değer aralığı kestirilememektedir. Sahte bölgelerin tespitinde, tespit yöntemlerinin özellik vektörlerini karşılaştırması gerekmektedir. Özellik vektörü değerlerinin herhangi bir değeri aldığı düşünüldüğünde bu vektörlerin karşılaştırılması ve uygun eşik değerinin bulunması birçok test gerektirip zaman alacaktır.

Özellik çıkarımında AKD faz yönteminin kullanım nedenini göstermek için, bir görüntüden ve görüntünün JPEG ile sıkıştırılmış halinden 8x8 boyutunda farklı karakteristikte (düz ve karmaşık yapıya sahip) iki blok alınmıştır. Bu şekilde yöntemin eşik değerini nasıl kolay bir şekilde belirlediği de gösterilecektir.

Test bloklarının kalite faktörü 50 olarak belirlenmiştir. Yöntemde bloğun AKD'si alınarak elde edilen katsayılar belirli bir değer ile kuantalanmıştır. Bloğun zigzag taraması sonucunda elde edilen vektörün ilk 16'sı alınarak bloğun özellik vektörü oluşturulmuştur. Tablo 2.2 blokların özellik vektörünü göstermektedir (Orijinal ve kalite faktörü QF=50 ile sıkıştırılmış blokların). Sahte görüntüye, sahteciliğin izlerini gidermek amacıyla JPEG sıkıştırma uygulandığında Tablo 2.2'den de görüldüğü üzere orijinal blok ile sıkıştırılmış halinden farklı özellik vektörleri üretilecektir.

Alınan blok düz bir bölge içeriyorsa, iki özellik vektörü arasındaki Öklid uzaklığı $\sqrt{7} \cong 2.64$ olduğundan benzerlik eşik değeri en azından 2.64 olmalıdır. Eğer alınan blok yüksek karmaşıklığa sahipse eşik değeri Öklid uzağı sonucunda en az $\sqrt{4} = 2$ olmalıdır.

Sonuç olarak tüm görüntü için seçilecek eşik değeri 2.64 olursa yüksek karmaşıklıkta sahte bölgeler yakalanamayacak veya aslında sahte olmayan bloklar sahte olarak işaretlenip yöntemin yanlış pozitif değeri artırılabilecektir. Eşik değeri 2.64 yerine 2 olarak seçilirse düz sahte bölgeler yakalanamayacaktır. Görüldüğü üzere Öklid uzaklığı karşılaştırma yöntemi olarak seçildiğinde uygun eşik değerin belirlenmesi çeşitli karakteristikteki görüntüler ile birlikte birçok test gerektirmektedir. Farklı karakteristikteki test görüntüleri de farklı eşik değeri gerektirecektir. Bunun sonucunda birçok çalışmada benzerlik eşik değerinin belirlenebilmesi için ortalama eşik değeri kullanılmaktadır.

Aynı blokların ve sıkıştırılmış hallerinin önerilen yöntem sonucunda çıkarılan özellik vektörleri Tablo 2.3'te gösterilmektedir. Tablo 2.3'ten de görüldüğü üzere özellik vektörleri, sadece faz terimleri kullanılarak karşılık gelen bloklar hakkında oldukça büyük bir bilgi içermektedir. AKD katsayılarına karşılık gelen faz terimleri özellik vektörlerinin benzerlik değerlendirilmesi için yeterli olmakla birlikte üretilen özellik vektörleri elemanlarının sadece $\{-1, 0, 1\}$ kümesinden değer alabilmesi vektör karşılaştırmasını oldukça kolay hale getirmektedir.

Tablo 2.2. Düz ve karmaşık blokların ve JPEG sıkıştırılmış hallerinin özellik vektörleri

Karmaşık Blok (Sıkıştırılmamış)	98	2	5	9	3	0	9	-9	-1	0	4	-5	8	-1	0	0
Karmaşık Blok (QF=50)	97	2	5	8	3	0	9	-9	-1	0	4	-5	7	-1	-1	0
Düz Blok (Sıkıştırılmamış)	111	0	-1	0	-1	-1	0	-1	0	0	0	-1	-1	-1	0	0
Düz Blok(QF=50)	111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Yöntemde iki vektör arasındaki benzerliğe vektörlerinin karşılıklı elemanları arasındaki benzerliğe bakılarak karar verilmektedir. İki vektörün karşılıklı eşit elemanlarının yüzdesi, belirli bir yüzdeden büyükse bu bloklar aynı kabul edilmektedir. Özellik çıkarımında AKD faz terimlerinin kullanılmasının sonucunda, AKD katsayılarının işareti değişmedikçe sıkıştırmadan ötürü AKD katsayılarındaki sayısal değişiklikler yöntemin performansını etkilememektedir. Tablo 2.3'ten de görüldüğü üzere karmaşık bloğun sıkıştırılmış ve orijinal vektör elemanları değerlerinin 16 'sının 15'i, düz bloğun ise tüm elemanları aynıdır. Bu değerler sonucunda iki vektörün karşılıklı eşit eleman sayısının tüm eleman sayısına oranı $14/16$ (en azından 16 elemanın 14'ü eşitse)'dan yüksekse bu iki

vektörün aynı bloğu temsil ettiği söylenebilmektedir. İki vektörün karşılaştırılmasında kullanılan ve (2.15)'te verilen t_s eşik değeri bu şekilde belirlenmekte, bu değerin bulunabilmesi için birçok test yapılmasına gerek kalmamaktadır. Yöntemde t_s eşik değerini şüpheli görüntünün karakteristiği belirleyecektir. Algoritmanın detayları bir sonraki bölümde verilecektir.

Tablo 2.3. Düz ve karmaşık bloklar ve sıkıştırılmış hallerinin önerilen yöntem sonucunda çıkarılan özellik vektörleri

Karmaşık blok (Sıkıştırılmamış)	1	1	1	1	1	0	1	-1	-1	0	1	-1	1	-1	0	0
Karmaşık Blok (QF=50)	1	1	1	1	1	0	1	-1	-1	0	1	-1	1	-1	-1	0
Düz Blok (Sıkıştırılmamış)	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Düz Blok(QF=50)	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Eşik değerinin belirlenmesi: Eşik değeri t_s genelleştirilmiş Benford'un yasası kullanılarak otomatik olarak belirlenecektir. Benford'un yasası olarak bilinen İlk Rakam yasasına göre: Bir doğal sayılar kümesinin ilk rakamlarının olasılık dağılımı logaritmiktir. Bir veri seti Benford'un yasasına göre (2.16)'daki kriteri sağlayacaktır.

$$p(x) = \log_{10}(1 + 1/x) \quad , x = 1 \dots 9 \quad (2.16)$$

$p(x)$, x 'in olasılığıdır. Fu vd. AKD katsayılarının ilk rakamlarının dağılımının (2.16) kriterini sağladığını göstermiştir [147]. Ancak JPEG ile sıkıştırılmış görüntünün AKD katsayılarının ilk rakam dağılımı yaklaşık olarak logaritmik yasaya uygunluk göstermektedir. Bunun için JPEG katsayılarının ilk rakamlarının dağılımını modelleyebilmek için Fu vd. (2.17)'de verilen parametrik logaritmik fonksiyon olarak adlandırılan Genelleştirilmiş Benford'un yasasını önermişlerdir [147].

$$p(x) = N \log_{10} \left(1 + \frac{1}{s+x^q} \right), x = 1, \dots, 9 \quad (2.17)$$

Burada N normalizasyon faktörü, s ve q çeşitli görüntü ve kalite faktörlerinin dağılımlarını tanımlayabilmek için model parametreleridir. Yöntemde t_s eşik değerinin belirlenebilmesi için Benford'un genelleştirilmiş yasasından faydalanılmıştır.

Öncelikli olarak yöntem test görüntüsünün sıkıştırılıp sıkıştırılmadığını test etmektedir. Eğer görüntü sıkıştırılmış ise ikinci olarak kalite faktörü belirlenmektedir. Test

görüntüsü sıkıştırılmamış görüntü ise de yöntemde bu görüntü bulanıklaştırma ya da görüntü ekleme atağı yapılmış görüntü olarak düşünülmektedir.

[147]'de önerilen yöntemde göre ilk aşama olarak JPEG sıkıştırmada en yüksek kalite faktörü 100 olduğu için, test görüntüsü $QF=100$ ile sıkıştırılmaktadır. Eğer test görüntüsü önceden sıkıştırılmamış ise yeni görüntü ilk kez sıkıştırılmış olacaktır. JPEG katsayılarının (kuantalanmış AKD katsayıları) ilk rakam dağılımları elde edilerek görüntünün önceden sıkıştırılıp sıkıştırılmadığına karar verilecektir. Eğer JPEG katsayılarının dağılımı Benford'un yasasına uyarsa görüntü ilk kez sıkıştırılmış demektir. Aksi takdirde ise görüntü öncesinde sıkıştırılmıştır. Yöntemin etkinliği gösterilmek için oluşturduğumuz veri tabanından sıkıştırılmamış bir görüntü ve bu görüntünün $QF=50$ ile sıkıştırılmış hali alınmış ve $QF=100$ ile her iki görüntü de sıkıştırılmıştır. Sonuç görüntülerin JPEG katsayılarının ilk rakam dağılımları Şekil 2.5.'te gösterilmektedir. Eğer test görüntüsü ilk kez sıkıştırılmış ise (mavi eğri), ilk rakam dağılımı Benford'un yasasına (yeşil eğri) uymaktadır. Eğer çift sıkıştırılmış ise (kırmızı eğri), dağılımı Benford'un yasasına uymayacaktır.

Görüntünün sıkıştırılmış görüntü olduğu tespit edildiğinde ikinci aşama olarak F_u 'nun yöntemine göre kalite faktörü belirlenecektir. Bu yöntem bir görüntünün bir kalite faktörü ile tekrar sıkıştırıldığında (ilk sıkıştırmadaki kalite faktöründen farklı olarak), tekrar sıkıştırılmış görüntünün JPEG katsayılarının ilk rakam dağılımı, ikinci kalite faktörünün ilk kalite faktörüne eşit oluncaya kadar Benford'a uymadığını göstermektedir. Kalite faktörü belirlenmesi yalancı kodu aşağıda verilmektedir.

Algoritma 1. Kalite Faktörünün Belirlenmesi

```

Giriş: Test görüntüsü I
Çıkış: Kalite faktörü q
% Compress (a, b) → a görüntüsü b kalite faktörü ile sıkıştırılır
% Sffit_Curve (x, a) → a'nın JPEG katsayılarının ilk rakam dağılımı elde edilerek
Benford'un genelleştirilmiş yasasına uydurulur ve geri dönüş değeri SSE'yi verir.
j=1; x = [1..9]; min_sse = 999;
for qf=30:5:100
    I_compressed[j] = Compress(I, qf);
    j = j + 1;
end
j_end = j; qf = 30;
for j=1: j_end
    temp = Sffit_Curve(x, I_compressed[j]);
    if temp < min_sse
        min_sse = temp;
    end
end

```

```

    q = qf;
end if
qf=qf + 5;
end
return q;

```

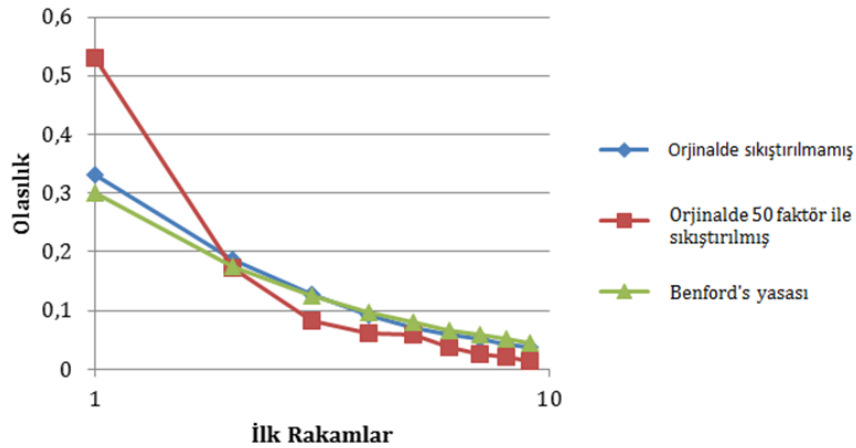
Belirlenen kalite faktörü q , eşik değeri t_s 'nin tahmininde kullanılacaktır. İki vektör arasındaki en iyi eşleşme 16 eleman değerinin hepsinin eşit olduğu zaman elde edilmektedir. Bu durumda r^{ij} değeri 1 olacaktır. Önerilen yöntem tespit edilen kalite faktörü q 'yu kullanarak (2.18)'de verildiği gibi benzerlik oranını belirlemektedir.

$$\forall(q \leq 30) \Rightarrow t_s = 12/16 = 0.75$$

$$\forall(q > 30) \wedge (q < 70) \Rightarrow t_s = 13/16 \cong 0.81$$

$$\forall(q \geq 70) \Rightarrow t_s = 14/16 \cong 0.87$$

(2.18)



Şekil 2.5. Sıkıştırılmamış ve QF=50 faktör ile sıkıştırılmış versiyonunun JPEG katsayılarının ilk rakam dağılımları. (Log-log ölçeği kullanılmıştır)

Eğer ilk aşamada görüntünün sıkıştırılmamış olduğu tespit edilirse t_s değeri algoritma tarafından 0.87 olarak seçilmektedir.

Bölgelerin İşaretlenmesi: Algoritmanın bu kısmında sahte bölgelerin belirlenebilmesi için sh dizisi değerlendirilecektir. Eğer shift vektörlerin sayısı t_{shift} eşik değerini aşarsa bu vektörlere karşılık gelen bloklar sahte olarak işaretlenir. Aksi takdirde bu bloklar ihmal edilir. t_{shift} değeri benzer ancak tekrar edilmemiş blokları elemek için kullanılmaktadır.

Algoritma 2. Bölgelerin İşaretlenmesi

```

Giriş: Vektör sh, Test görüntüsü I
Çıkış: İşaretlenmiş görüntü MI
shift = sort(sh); MI = I;
[N, M] = size(shift);
j = 1;
while (j<=N)
    [temp_x temp_y] = shift(j,5:6);
    i=j+1; z = j;
    while (i<=N)
        [x y] = shift(i,5:6);
        if (x==temp_x) and (y==temp_y)
            count = count + 1;
            i = i + 1;
        else
            break;
        end
    end
    if count > t_shift
        for k = j: i
            MI(shift(k, 1): shift(k, 1)+7, shift(k,2): shift(k,2)+7) = 0;
            MI(shift(k, 3): shift(k, 3)+7, shift(k,4): shift(k,4)+7) = 0;
        end
    end
end
end

```

2.1.3. LBP-AKD ‘ye Dayalı Kopyala-Yapıştır Sahteciliği Tespiti

Kopyala-yapıştır sahteciliğinin tespiti için önerilen AKD tabanlı yöntemlerde amaç yöntemi JPEG sıkıştırma ataklarına karşı dayanıklı hale getirmektir. Önerilen çalışmada LBP ve AKD’ye dayalı yeni bir kopyala-yapıştır sahteciliği tespit yöntemi gerçekleştirilmiştir [148]. Literatürde sahtecilik tespitinde ilk defa LBP ve AKD yöntemleri birlikte kullanılarak diğer AKD tabanlı yöntemlere göre daha etkin sonuçlar elde edilmiştir.

Çalışma özellik çıkarımı, eşleştirme ve işaretleme olarak iki kısımdan oluşmaktadır.

Özellik Çıkarımı: Yöntemi JPEG ataklara karşı dayanıklı hale getirmek için görüntü özellik çıkarımından önce YCbCr uzayına dönüştürülmektedir. NxM boyutunda YCbCr uzayındaki test görüntüsü I ile temsil edilsin. Test görüntüsü birbiri ile örtüşen 8x8 büyüklüğündeki alt bloklara ayrıştırılmaktadır. Görüntüdeki her bir blok $B_i, i = 1 \dots (N - 7)(M - 7)$ ile temsil edilmektedir. Her bir blok için aşağıda verilen algoritma adımları uygulanmaktadır.

Adım 1: Mevcut blok için her bir kanalın (Y, Cb, Cr) ortalama parlaklık değerleri hesaplanır ve bu değerler Y_{avg} , Cb_{avg} , Cr_{avg} ile temsil edilmektedir.

Adım 2: Mevcut bloktaki her bir piksel değeri için (8,2) komşulukta LBP değeri hesaplanır. LBP etiketleri ile oluşturulan yeni blok L_i ile temsil edilir. Ve yeni bloğun sadece Y kanalı alınır.

Adım 3: LBP etiketlerinden oluşan L_i bloğu AKD kullanılarak frekans uzayına dönüştürülür. Dönüşümden sonraki yeni blok $D_i = DCT(L_i)$ ile temsil edilir.

Adım 4: Frekans uzayındaki D_i zigzag taranır ve tarama sonucundaki ilk 15 eleman blok özellik vektörü V_i 'yi oluşturur.

Adım 5: Vektör $S_i = \{s_{ij} | j = 1 \dots 15\}$, $V_i = \{v_{ij} | j = 1 \dots 15\}$ vektöründeki AKD katsayılarının işaretlerini tutmakta ve (2.19)'da verildiği gibi hesaplanmaktadır.

$$\forall v_{ij} < 0 \Rightarrow s_{ij} = -1 \quad \forall v_{ij} > 0 \Rightarrow s_{ij} = 1 \quad \forall v_{ij} = 0 \Rightarrow s_{ij} = 0 \quad (2.19)$$

Adım 6: İşaret vektörü S_i ve kanalların ortalama parlaklık değerleri Y_{avg} , Cb_{avg} , Cr_{avg} (2.20)'de verildiği gibi 1x18 boyutundaki blok özellik vektörü $F_i = \{f_{ij} | j = 1 \dots 18\}$ 'yi oluşturmaktadır.

$$\forall j \in [1 - 15] \Rightarrow f_{ij} = s_{ij} \quad f_{i16} = Y_{avg}, f_{i17} = Cb_{avg}, f_{i18} = Cr_{avg} \quad (2.20)$$

Özellik çıkarım algoritması her bir blok için LBP bloğu oluşturup bu bloğu AKD yardımıyla frekans uzayına dönüştürerek özellik vektörü oluşturmaktadır. AKD dönüşümü kullanılarak yöntemin JPEG sıkıştırma ataklarına karşı dayanıklı hale getirilmesi sağlanmıştır. AKD katsayılarının $\{-1, 0, 1\}$ değeri alabilen işaret bilgisi kullanılmıştır. Eşleştirme algoritmasının farklı renk karakteristiğindeki blok kıyaslamalarını elimine etmesi için bloklardan çıkarılan özellik vektörünün son üç elemanı kullanılmıştır. Tüm bloklardan çıkarılan özellik vektör matrisi $[(N - 7)(M - 7), 18]$ boyutunda bir A matrisi ile temsil edilmektedir. Benzer vektörleri bir araya getirmek için bu matris sözlüksel olarak sıralanmaktadır. Eşleştirme öncesi sözlüksel sıralama, bir vektör kendisinden sonraki belirli sayıda vektör ile kıyaslanacağından işlem zamanını oldukça düşürmektedir.

Eşleştirme ve İşaretleme: Önerilen yöntemde vektörler arasında benzerliğin belirlenebilmesi için benzer vektörlerden shift vektörü hesaplanmaktadır. Benzer vektörleri

bulabilmek için A matrisindeki her bir vektör kendisinden sonraki t_n tane özellik vektörü ile kıyaslanmaktadır. Eğer herhangi iki vektör benzer ise, algoritma bu vektörlere karşılık gelen kayma vektörünü hesaplayacak ve bunları sh ile temsil edilen listede kaydedecektir. Aşağıda verilen eşleştirme algoritması adımları A matrisindeki her bir özellik vektörü için tekrarlanacaktır.

Adım 1: Renk karakteristiğini belirleyebilmek için F_i vektörünün son üç elemanı kendisinden sonraki F_j vektörünün son üç elemanı ile kıyaslanacaktır. Eğer renk şartı sağlanır ise (2.21)'de görüldüğü gibi algoritmada bir sonraki adıma geçilecek aksi takdirde Adım1 tekrarlanarak bir sonraki vektör ile değerlendirilecektir. Algoritmada iki vektörün benzer dağılıma sahip olup olmadığını belirleyebilmek için t_c eşik değerinden faydalanılmaktadır.

$$\forall (|f_{j16} - f_{i16}| < t_c) \wedge (|f_{j17} - f_{i17}| < t_c) \wedge (|f_{j18} - f_{i18}| < t_c) \Rightarrow \text{Adım 2'ye geç} \\ \text{değilse} \Rightarrow k = k + 1 ; \text{Adım 1'e geç} \quad (2.21)$$

Adım 2. Eğer iki vektör benzer ise bir sonraki adım blokların ne kadar uzaklıkta olduğuna bakılmasıdır. Bununla benzer vektörlerin düz bölgelerden gelmemesi sağlanmaktadır. F_i ve F_j vektörlerinin sol üst köşe koordinatları sırasıyla (x_i, y_i) ve (x_j, y_j) olsun. Bloklar arası uzaklığın uygun olup olmadığına bakılacaktır. Önerilen yöntemde iki blok arası uzaklık en azından eşik değeri t_d 'den büyük olması gerekmektedir. Bu adım (2.22)'de verilmektedir.

$$\forall \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \geq t_d \Rightarrow \text{Adım 3'e geç}, \quad \text{değilse} \Rightarrow \text{Adım 1'e geç} \quad (2.22)$$

Adım 3. Bu kriterle de $k = 1 \dots 15$ için vektörlerin elemanları arası benzerlik (2.23)'teki gibi değerlendirilecektir.

$$\forall (f_{ik} - f_{jk}) = 0 \Rightarrow cnt = cnt + 1 \quad (2.23)$$

Eğer iki vektör aynı ise cnt değeri 15 olacaktır. Yöntemde benzerliğin test edilmesinde $cnt/15$ değeri önceden belirlenmiş t_s değeri ile kıyaslanacaktır. Eğer $cnt/15 > t_s$ ise bir sonraki adıma geçilecek değilse başa dönülecektir.

Adım 4. İki vektör arasında shift vektörü hesaplanacak ve bu değer sh listesine eklenecektir. Bu listede aynı zamanda kayma vektörünün karşılık geldiği blokların üst köşe koordinatları da tutulmaktadır. Vektör $\begin{bmatrix} |x_i - x_j| & |y_i - y_j| & x_i & y_i & x_j & y_j \end{bmatrix}$ bu şekilde sh listesine eklenecektir. Eklemekten sonra algoritma tekrar Adım 1'e dönerek bir sonraki vektörü değerlendirecektir. Eğer aynı shift vektörlerinin sayısı eşik değeri t_{shift} 'i aşarsa algoritma bu bölgeleri işaretleyecektir.

2.1.4. LPQ 'ya Dayalı Kopyala-Yapıştır Sahteciliği Tespiti

Kopyala-yapıştır sahteciliğinde görüntü üzerindeki bir bölge alınarak yine aynı görüntü üzerine yapıştırılmaktadır. Yapıştırılma sonrası sahtecilik izlerini kapatmak için görüntüye çeşitli ara ve son işlem operasyonları uygulanmaktadır. Bu operasyonlardan en yaygın olanı da bulanıklaştırma işlemidir. Bu nedenle çalışmada görüntü bloklarından özellik çıkarımında literatürde ilk kez LPQ kullanılmıştır [149]. LPQ ile özellik çıkarım sonucunda görüntü yüksek seviyelerde bulanıklaştırılrsa dahi yöntemden yüksek tanılama performansı elde edilmiştir. Aşağıda önce LPQ yönteminin detayları verilecek ardından önerilen yönteminin adımlarından bahsedilecektir.

2.1.4.1. Yerel Faz Nicemleyicisi-LPQ

Yerel Faz Nicemleme operatörü, 2008 yılında Ojansivu ve Heikkila tarafından önerilen bulanıklaştırma işleminden bağımsız bir doku sınıflandırma yöntemidir [150]. LPQ, görüntünün yerel faz bilgisini kullanmaktadır. Bu bilgi, her görüntü konumu için bir pencerede yerel olarak hesaplanan kısa vadeli Fourier dönüşümü (Short-Term Fourier Transform) kullanılarak çıkarılır. Düşük frekans çözünürlüğü daha yüksek uzaysal çözünürlüğü ortaya çıkarmaktadır. Düşük frekanslı faz açıları merkezi olarak simetrik olup bulanıklığın değişmez olduğunu gösterir.

LPQ, Ayırık Fourier Dönüşümünün (Discrete Fourier Transform-DFT) kuantalanmış faz bilgisine dayalı uzaysal bulanıklaştırma için önerilmiştir.

Zamansal uzayda bulanıklaştırma (2.24)'te verildiği gibi görüntü parlaklıkları ve nokta yayılım fonksiyonu (Point Spread Function-PSF) arasındaki konvolüsyon işlemi ile temsil edilmektedir [150].

$$g(x) = (f * h)(x) \quad (2.24)$$

$g(x)$, $f(x)$ ve $h(x)$ sırasıyla orijinal görüntünün uzaysal bulanık bağımsız ayırık modeli, orijinal görüntü ve PSF'yi temsil etmektedir. $*$ işlemi 2-D konvolüsyonu ve x ise koordinatları $[x, y]^T$ olan bir vektörü temsil etmektedir. Fourier alanında (2.24) eşitliği (2.25)'e karşılık gelmektedir.

$$G(u) = F(u) \cdot H(u) \quad (2.25)$$

Burada $G(u)$, $F(u)$ ve $H(u)$ sırasıyla bulanıklaştırılmış görüntü $g(x)$ 'nin, orijinal görüntü $f(x)$ 'nin ve PSF fonksiyonunu $h(x)$ 'nin ayırık Fourier dönüşümlerini ve u koordinatları $[u, v]^T$ olan bir vektörü temsil etmektedir. (2.25) denklemini (2.26)'daki gibi genlik ve (2.27)'deki gibi faz olarak iki kısma ayırırsak;

$$G(u) = |F(u)| \cdot |H(u)| \quad (2.26)$$

$$\angle G(u) = \angle F(u) + \angle H(u) \quad (2.27)$$

Eğer bulanık PSF, $h(x)$ merkezi simetrik, $h(x)=h(-x)$, bir fonksiyon olursa bu fonksiyonun Fourier dönüşümü (2.28)'de görüldüğü gibi gerçek değerli ve faz değeri iki değerli $\angle H(u) \in (0, \pi)$ bir fonksiyon olacaktır.

$$\angle H(u) = \begin{cases} 0, & H(u) \geq 0 \\ \pi, & H(u) < 0 \end{cases} \quad (2.28)$$

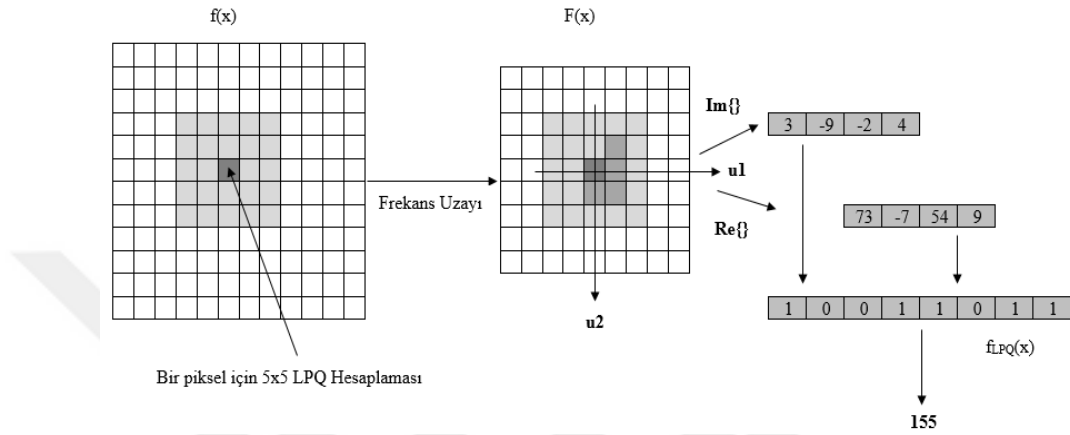
Bunun sonucunda; tüm $H(u) \geq 0$ için $\angle G(u) = \angle F(u)$ olacaktır.

$H(u)$ 'nun pozitif olduğu yerlerde elde edilen görüntünün fazı merkezi simetrik bulanıklaştırmadan bağımsız olacaktır.

LPQ yöntemi, üstte verilen Fourier faz spektrumunun bulanıklaştırma bağımsız özelliğine dayanmaktadır. Yöntemde Şekil 2.6.'da görüldüğü gibi $f(x)$ görüntüsünde x pozisyonundaki her bir pikselin $M \times M$ 'lik komşuluğundaki N_x piksel için (2.29)'da tanımlandığı gibi kısa zamanlı Fourier dönüşümü uygulanmaktadır.

$$F(u, x) = \sum_{y \in N_x} f(x - y) e^{-j2\pi u^T y} = w_u^T f_x \quad (2.29)$$

Burada w_u frekans u 'da 2-D DFT'nin temel vektörü, f_x ise N_x 'den M^2 örnek içeren bir vektördür.



Şekil 2.6. Bir piksel için 5x5'lik bölgede LPQ etiketinin elde edilmesi

STFT'yi gerçekleştirmenin en etkili yöntemi tüm u değerleri için 2-D konvolüsyon $f(x) * e^{-2\pi u^T x}$ uygulamaktır. Hesaplama, temel vektörler ayrılabilir olduğundan satırlar ve sütunlara 1-D konvolüsyon uygulanarak gerçekleştirilmektedir.

LPQ yönteminde yerel Fourier katsayıları sadece $[0, \pi/2, \pi, 3\pi/2]$ dört açı için hesaplanmakta bu iki boyutlu frekansta $u_1 = [a, 0]^T$, $u_2 = [0, a]^T$, $u_3 = [a, a]^T$, ve $u_4 = [a, -a]^T$ 'e karşılık gelmektedir. Bunun sonucunda (2.30)'da verilen F_x vektörü elde edilmektedir.

$$F_x^c = [F(u_1, x), F(u_2, x), F(u_3, x), F(u_4, x)] \quad \text{ve} \quad F_x = [\text{Re}\{F_x^c\}, \text{Im}\{F_x^c\}]^T \quad (2.30)$$

Burada $\text{Re}\{\cdot\}$ ve $\text{Im}\{\cdot\}$ sırasıyla kompleks sayının reel ve sanal kısımlarını döndürmektedir.

Fourier katsayılarının faz bilgisi F_x^c deki her bir bileşenin reel ve sanal kısımlarının işaret bilgisinin alınmasıyla elde edilmektedir. Bu işlemde (2.31)'de verildiği gibi basit bir sayısal kuantalama işlemiyle yapılabilir.

$$q_j = \begin{cases} 1, & \text{eğer } g_j \geq 0 \\ 0, & \text{aksi takdirde} \end{cases} \quad (2.31)$$

Burada g_j F_x 'deki her bir bileşeni temsil etmektedir. Elde edilen q_j ler kullanılarak $f_{LPQ(x)} \sum_{j=1}^8 q_j 2^{j-1}$ fonksiyonu ile 0-255 değer aralığında değişen tamsayı değerler elde edilmektedir. Bu değerlerin histogramı alınarak da tüm görüntüden elde edilen LPQ özellik vektörü çıkarılmış olmaktadır.

Önerilen yöntem iki aşamadan oluşmaktadır: Özellik çıkarımı ve Benzerlik araştırması ve işaretleme.

Özellik Çıkarımı: Görüntü öncelikle örtüşen alt bloklara bölünmekte ve her blokta LPQ ile özellik çıkarılmaktadır. Özellik vektörleri bir matrise yerleştirilmektedir. Sonrasında benzer vektörleri bir araya getirebilmek için bu matris sözlüksel olarak sıralanmaktadır. Benzer vektörler ise sahte bölgelere işaret etmektedir. Özellik çıkarımın detayları aşağıda adımlar halinde verilmektedir.

Adım 1: $M \times N$ boyutundaki gri seviye şüpheli I görüntüsü örtüşen $b \times b$ boyutundaki bloklara ayrıştırılmaktadır. Çalışmada b değeri 18 olarak alınmıştır.

Adım 2: Her bir blok B_i $i = 1 \dots (N - 17)(M - 17)$ pencere büyüklüğü 9 alınarak elde edilmiş LPQ değerlerinden oluşmaktadır. LPQ değerleri önceden belirlenmiş bir q_t değeri ile kuantalanmaktadır. Dolayısıyla $B_i = B_i/q_t$ olmaktadır. B_i 'nin boyutu 10×10 olmakla beraber bu matrisin sütun değerleri ortalaması alınarak her bir blok için 1×10 boyutundaki F^i özellik vektörü elde edilmektedir. Tüm bloklar için elde edilen özellik vektörleri F^i 'ler bir araya getirilerek $[(N - 17)(M - 17), 10]$ boyutunda A matrisi elde edilmektedir. Sonrasında A matrisi sözlüksel olarak sıralanarak benzer vektörler bir araya getirilmektedir. Vektörlerin eşleştirilmesi işleminden önceki yapılan bu sözlüksek sıralama, bir vektör tüm özellik vektörleri yerine komşu özellik vektörlerinin sadece önceden tanımlanmış miktarı ile karşılaştırılacağı için benzerlik araştırmasını hızlandırır.

Benzerlik araştırması ve işaretleme: Özellik vektör matrisi içerisinde benzer blokların tespit edilmesi için araştırma yapılmaktadır. Eğer iki vektör benzer olarak bulunursa, algoritma bu vektörlere karşılık gelen shift vektörünü hesaplamakta ve vektörleri bir listeye kaydetmektedir. Benzer vektörlerin araştırılması algoritması aşağıda adımlar halinde verilmektedir. Bu algoritma A matrisindeki tüm F^i 'ler için uygulanmaktadır.

Adım 1: Vektörler arasındaki benzerlik Öklid uzaklığı ile belirlenmektedir. Elde edilen uzaklıkların belirli bir eşik değeri ile kıyaslanması sonucunda vektörlerin

benzerliğine karar verilmektedir. Eşik değeri t_s için iki vektör arasındaki benzerlik (2.32)'deki gibi hesaplanmaktadır.

$$F^i = (F_1^i \ F_2^i \ ... \ F_{10}^i) \quad \sqrt{\sum_{k=1}^{10} (F_k^i - F_k^j)^2} \leq t_s \quad (2.32)$$

Vektör F^i kendisinden sonra t_n özellik vektörü ile kıyaslanmaktadır.

Adım 2: Komşu vektörler arasındaki benzerlik olası sahte bölgelerin belirlenmesi için gerekli ancak yeterli değildir. Aynı zamanda benzer bloklar arasındaki uzaklık önceden belirlenmiş bir eşik değeri olan td 'den de büyük olması gerekmektedir. Örneğin sahtecilik yapılan bölgenin gökyüzü olduğu düşünülürse bu uzaklığın sağlanması elzem olmaktadır. F^i ve F^j vektörlerinin temsil ettiği blokların sol üst köşe koordinatları sırasıyla (x_i, y_i) ve (x_j, y_j) olsun. (2.33)'ten de görüldüğü gibi iki bloğun benzer olabilmesi için aralarındaki uzaklığın en az d kadar olması gerekmektedir.

$$\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \geq t_d \quad (2.33)$$

Adım 3: Son aşama olarak eğer iki vektör (2.32) ve (2.33)'teki kuralları sağlarsa bu vektörler arasında shift vektörü hesaplanmaktadır. Shift vektörünün hesaplanması diğer çalışmalarda verilmiştir. Aynı shift vektörüne sahip şüpheli blok çiftlerinin sayısı belirli bir eşik değerini geçerse bu blok çiftlerinin oluşturduğu bölgeler kopyala-yapıştır yapılmış bölgeler olarak işaretlenmektedir.

2.1.5. Renkli SIFT Yöntemine Dayalı Kopyala-Yapıştır Sahteciliği Tespiti

Literatürde Kopyala-yapıştır sahteciliği tespiti için önerilen anahtar nokta tabanlı yöntemlerden olan SIFT tabanlı çalışmalar, gri seviyeye dönüştürülmüş görüntüler üzerinde çalışmaktadır. Önerilen yöntemde renkli SIFT yöntemiyle renkli görüntü üzerindeki kopyala-yapıştır sahteciliği tespit edilmektedir. Çalışmada görüntüyü gri seviyeye çevirmek yerine renkli görüntüye ait kanalların her birine SIFT yöntemi uygulanarak görüntünün anahtar noktaları çıkarılmaktadır [151]. Anahtar noktaların eşleştirilmesi ile kopyala yapıştır yapılmış olan bölgeler tespit edilmektedir. Çalışmada,

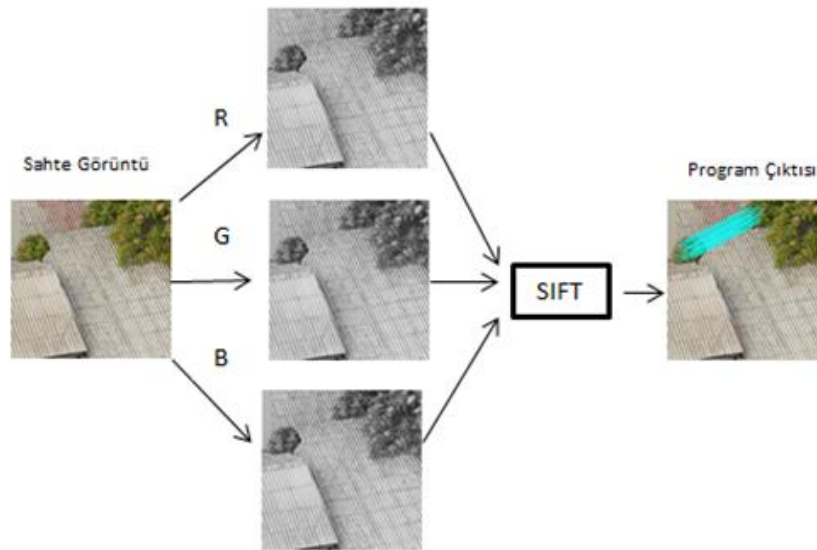
kopyala-yapıştır sahteciliğinin tespitinde, görüntü özelliklerinin çıkarımında literatürde ilk olarak renkli SIFT algoritmasının kullanılması gerçekleştirilmiştir. Aynı zamanda renkli SIFT'in SIFT'e göre daha etkin sahtecilik tespiti yaptığı gösterilmiştir.

Önerilen yöntemin akış diyagramı Şekil 2.7.'de verilmektedir. Şekilden de görüldüğü üzere $N \times M \times 3$ büyüklüğündeki renkli sahte görüntünün R,G ve B kanalı için ayrı ayrı SIFT algoritması uygulanmaktadır. Üç ayrı kanal için elde edilen anahtar noktalar k, y ve m 'nin özellik tanımlayıcıları birleştirilmekte ve anahtar nokta sayısı kadar 128 boyutlu özellik tanımlayıcıları elde edilmektedir. Anahtar noktalar ve tanımlayıcıları $D=(k + l + m) \times 128$ boyutlu bir matriste tutulur.

Bir sonraki adımda her bir özellik tanımlayıcısı vektör ile diğer tüm tanımlayıcı vektörler arasındaki iç çarpım p hesaplanır ve iç çarpımların (2.34)'de verildiği gibi ters kosinüs bileşenleri sıralanır.

$$[Değer, İndex] = sort(cos^{-1}(p)) \quad (2.34)$$

Sıralama sonucunda iki en yakın komşu Değer(1) ve Değer(2) arasındaki oran belirli bir eşik değeri t ile karşılaştırılmaktadır. Eğer $Değer(1)/Değer(2) < t$ ise bu iki özellik tanımlayıcısının ait olduğu anahtar noktalar eşleştirilir. Son olarak ise elde edilen eşleşen noktalar kümesindeki her bir nokta çifti arasında doğru çizdirilerek kopyala-yapıştır yapılmış bölgeler tespit edilir.



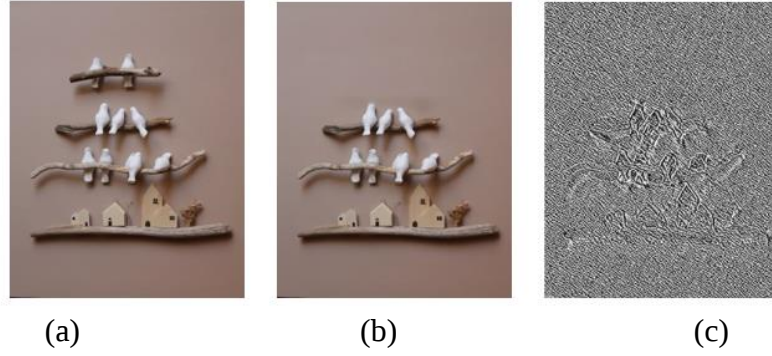
Şekil 2.7. Önerilen yöntemin akış diyagramı

2.1.6. LPQ ve SIFT Yöntemlerine Dayalı Anahtar Nokta Tabanlı Kopyala-Yapıştır Sahteciliğinin Tespiti

Literatürde pasif görüntü doğrulama yöntemleri blok ve anahtar nokta tabanlı yöntemler olmak üzere ikiye ayrılmaktadır. Anahtar nokta tabanlı yöntemler, blok tabanlı yöntemlerin meydana getirdiği iş yükünü azaltmak için önerilmiştir. Ancak gerek blok tabanlı gerekse de anahtar nokta tabanlı yöntemler eğer görüntü üzerinde düz bir bölge (gökyüzü gibi) kopyalanıp başka bir bölgeyi gizlemek amacıyla bu bölgenin üzerine yapıştırılırsa etkin bir sonuç üretemeyecektir. Blok tabanlı yöntemler görüntüyü örtüşen alt bloklara bölmekte ve bloklar arasında bir eşik değeri kullanarak benzerlik araştırması yapmaktadır. Yanlış negatiflerin artmaması için çok sayıda benzer grup bloğu (örn. resimde gökyüzüne karşılık gelen iki blok grubu) ihmal edilmektedir. Anahtar nokta tabanlı yöntemler de karmaşık bölgelerden doku çıkarabildiğinden yine aynı şekilde düz sahte bölgeleri tespit edememektedir. Bu nedenle çalışmada LPQ ve SIFT yöntemlerine dayalı bir kopyala-yapıştır sahteciliği tespit yöntemi önerilmiştir [152]. Çalışmada öncelikle sahte görüntüden doku bilgisi çıkarılmaktadır. Görüntülerin görünüşte pürüzsüz bölgelerinde de bir doku (sensör ve/veya niceleme gürültüsünden dolayı) olduğu için önerilen yöntem, LPQ operatörünü kullanarak bu bölgelerin dokusunu ortaya çıkarmaktadır. Böylece, anahtar nokta çıkarım algoritmaları görüntünün dokusal bilgilerinden önemli noktalar elde edebilmektedir. Kopyala-yapıştır yapılmış bölgeler anahtar noktaların eşleştirilmesi ile tespit edilmektedir. Önerilen yöntem özellikle düz, pürüzsüz bölgelerde SIFT tabanlı yöntemlere kıyasla yüksek tanılama performansına sahiptir.

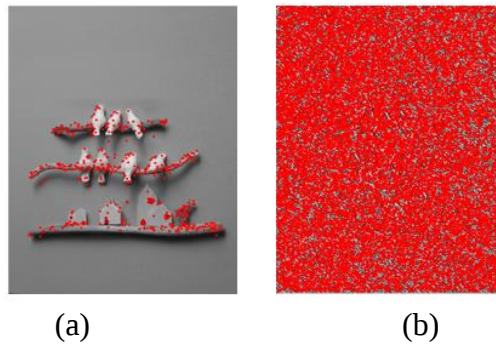
Önerilen yöntem üç aşamadan oluşmaktadır. Önce görüntüden LPQ ile doku görüntüsü oluşturulmakta sonrasında doku görüntüsünden SIFT ile anahtar nokta çıkarılmakta ve en son aşama olarak anahtar noktaların eşleştirilmesi ile kopyala-yapıştır yapılmış bölgeler tespit edilmektedir. Bu aşamalar aşağıda detaylandırılacaktır.

LPQ ile görüntüden doku çıkarımı: Önerilen yöntem sahte görüntünün yapısal doku bilgisini LPQ operatörü kullanarak çıkarmaktadır. Bu şekilde doku görüntüsünden SIFT ile anahtar nokta çıkarılabilmektedir. Şekil 2.8.(a), 2.8.(b) ve 2.8.(c)'de sırasıyla orijinal görüntü, Kopyala-yapıştır sahteciliği yapılmış sahte görüntü ve bu görüntüden LPQ ile oluşturulmuş doku görüntüsü verilmektedir.



Şekil 2.8. (a) Orijinal görüntü (b) Sahte görüntü (c) Doku Görüntüsü

LPQ görüntüsünden SIFT ile anahtar noktaların çıkarımı: Önerilen yöntemde LPQ ile çıkarılan doku görüntüsünden anahtar nokta çıkarımında SIFT yönteminden faydalanılmaktadır. Şekil 2.9.(a) ve 2.9.(b)'de sırasıyla Şekil 2.8.(b) ve 2.8.(c)'deki sahte görüntü ve bu görüntünün doku görüntüsünden SIFT ile çıkarılan anahtar noktalar gösterilmektedir. Şekilden de görüldüğü üzere SIFT duvar bölgesi düz bölge olduğu için bu bölgeden anahtar nokta çıkaramamaktadır. Aynı görüntünün doku bilgisini çıkarılmış haline SIFT uygulandığında ise düz ve karmaşık bölgelerin hepsinden birçok anahtar nokta elde edilmektedir. Sonuçtan da görüldüğü üzere doku bilgisi anahtar nokta sayısını artırdığından yöntemde anahtar nokta çıkarımından önce doku bilgisi çıkarılmaktadır.



Şekil 2.9 (a) SIFT sonucu (b) LPQ+SIFT sonucu

Kopyala-yapıştır sahteciliği tespitinde anahtar noktaların eşleştirilmesi: Anahtar noktaların eşleştirilmesi Renkli SIFT çalışmasında verildiği için bu bölümde

verilmeyecektir. Eşleşen anahtar noktalar kopyala-yapıştır yapılmış bölgeleri işaret etmektedir.

2.1.7. Pasif Görüntü Doğrulama Yöntemlerine Dayalı Medikal Görüntülerde Sahtecilik Tespiti

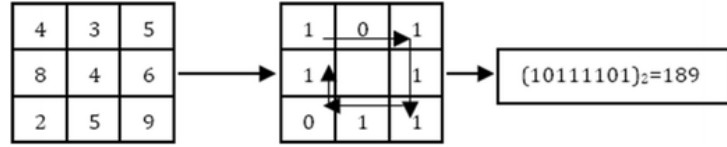
Teletıp son zamanlarda oldukça büyük bir popülerlik kazanmıştır. Tele teşhisi tıbbi personel arasında yaygınlaştırmak ve tıbbi personel tarafından hasta geçmişine herhangi bir ortamdan erişebilir hale getirmek için tıbbi görüntüler internet üzerinde transfer edilebilmektedir. Transferin internet üzerinden gerçekleştirildiği düşünülürse tıbbi görüntünün içerik doğrulaması yapılması gerekmektedir. Tıbbi görüntünün içerik kontrolünde birçok damgalama tekniği önerilmesi ile birlikte bu tekniklerde görüntü iletiminden önce extra bilgi (damga) gerekmektedir. Görüntüye extra bilgi gömülmesi ise görüntünün görsel kalitesini düşürmekte bu da tıbbi görüntünün yorumlanması aşamasında yanlış teşhislere yol açabilmektedir. Önerilen yöntem literatürde ilk olarak medikal görüntülerde oynanmış bölge tespitinde pasif görüntü doğrulama yöntemlerinden faydalanmaktadır [153]. Yöntemde anahtar nokta çıkarım yöntemlerini daha başarılı hale getirebilmek için medikal görüntüden yapısal doku bilgisi çıkarılmaktadır. Doku bilgisinin çıkarımında LBPROT operatöründen faydalanılırken anahtar nokta çıkarımında SIFT kullanılmıştır. Önerilen yöntem medikal görüntüler çeşitli ataklara uğrasa (Gauss bulanıklaştırma, Beyaz Gauss gürültüsü) ve tekrarlanacak bölge yapıştırılmadan önce ölçekleme ve rotasyon işlemi uygulansa dahi sahte bölgeyi tespit edebilmektedir.

LBPROT operatörü verildikten sonra önerilen yöntemin detaylarından bahsedilecektir.

2.1.7.1. LBPROT Operatörü

LBPROT, temel LBP operatörünün [154] rotasyondan bağımsız şekli olup önerilen çalışmada görüntüden doku bilgisi çıkarım amacıyla kullanılmıştır. LBP operatörü Şekil 2.10'da görüldüğü üzere her bir bloktaki merkez piksel ile komşu pikselleri arasındaki ikili farkı hesaplamaktadır. Genel LBP operatöründe örneklenen nokta sayısı ve komşuluk boyutunda herhangi bir sınırlama yoktur. Ojala ve ark. LBP'nin genelleştirilmiş şeklini yıllar sonra önerdiler [154].

I gri seviye görüntü olmak üzere x . satır ve y . sütundaki bir piksel $i_c = I(x,y)$ olsun. (x,y) noktasının çembersel komşuluğunu R yarıçapında P nokta oluşturacaktır. Şekil 2.11.'de farklı R yarıçapındaki 8 komşu noktalı çembersel komşuluklar görülmektedir.



Şekil 2.10 LBP operatörü

i_p , P noktalarının gri seviye değerini göstermek üzere örneklenen (x,y) noktası (2.35)'teki gibi hesaplanmaktadır. Önerilen yöntemde (x,y) noktasının 8 çembersel komşuluğu ve R yarıçapı kullanılmıştır.

$$\begin{aligned} i_p &= I(x_p, y_p), \quad p = 0, \dots, P-1 \\ x_p &= x + R \cos(2\pi p/P) \\ y_p &= y - R \sin(2\pi p/P) \end{aligned} \quad (2.35)$$

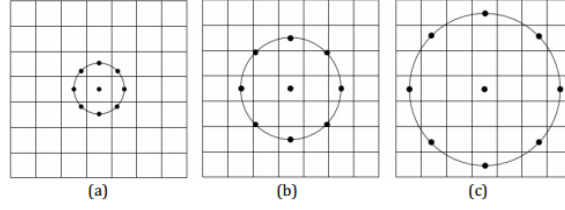
$s(t)$ birim basamak fonksiyonu göstermek üzere $t \geq 0$ olma durumunda fonksiyon bir aksi durumda sıfır sonucunu geri döndürecektir. Birim basamak fonksiyonu kullanılarak 8 komşuluklu R yarıçaplı noktanın LBP değeri (2.36)'da verildiği gibi hesaplanmaktadır.

$$LBP(x_p, y_p) = \sum_{p=0}^7 s(i_p - i_c) 2^p \quad (2.36)$$

Eğer görüntüdeki bir bölge herhangi bir bölgeye yapıştırılmadan önce döndürülürse, bölgenin LBP örüntüsü de merkezi etrafında dönecektir. Önerilen yöntemde LBPROT operatörü kullanılarak yöntemin rotasyon ataklarına dayanıklı olması sağlanmıştır. LBPROT operatörü (2.37)'de tanımlanmıştır. $ROR(x, i)$, x bit dizisinin i adımlarla sağa döndürülmüş dairesel bit dizisini ifade etmektedir.

$$LBP_{x_p, y_p}^i = \min_i ROR(LBP(x_p, y_p), i) \quad (2.37)$$

LBPROT operatörü, dairesel bit dizisi işlemleri sonuçları arasında en küçük LBP koda sahip olanını seçmektedir. En küçük LBP kodu, LBP_{x_p, y_p}^{ri} , 3×3 'lük bloğun merkez pikselini etiketlemek için kullanılır.



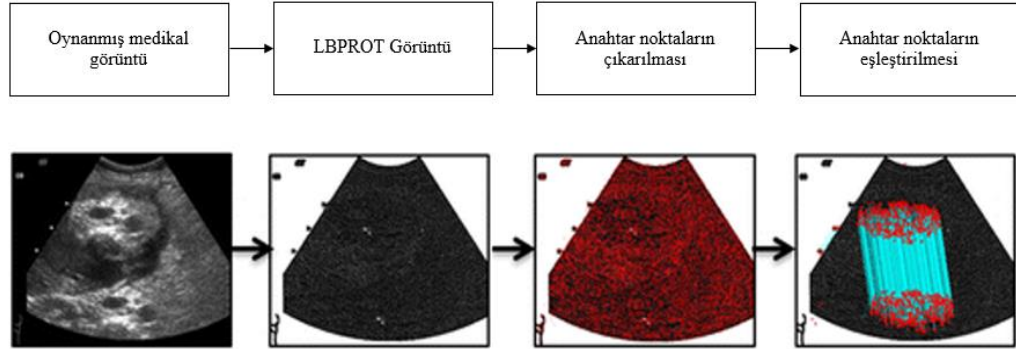
Şekil 2.11. 8 komşuluklu R yarıçaplı çembersel komşuluklar (a) LBP(8,1) (b) LBP(8,2) (c) LBP(8,3)

Çalışma;

- LBPROT operatörü ile görüntüden doku çıkarımı
- LBPROT görüntüsünden anahtar noktaların çıkarımı
- Oynanmış bölge tespitinde anahtar noktaların eşleştirilmesi

olmak üzere üç aşamadan meydana gelmektedir. Önerilen yöntemin akış diyagramı

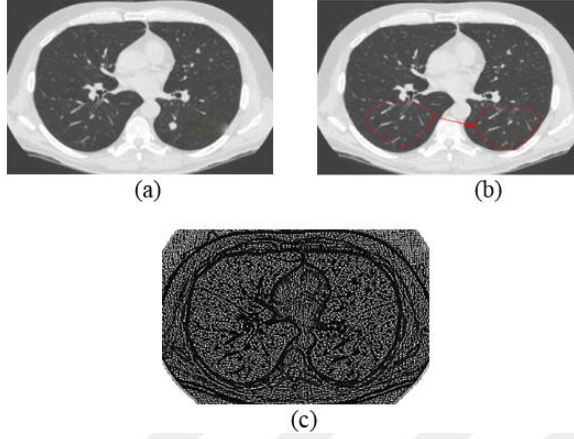
Şekil 2.12.'de verilmektedir. Aşamalar aşağıda detaylandırılacaktır.



Şekil 2.12. Önerilen yöntemin akış diyagramı

LBPROT operatörü ile görüntüden doku çıkarımı: Bu çalışmada ilk aşama olarak medikal görüntüden yapısal doku bilgisi çıkarılmaktadır. Görünüşte düz bölgeler (sensör ya da kuantalama sonucunda oluşan gürültü) bir dokuya sahiptir. Önerilen yöntem LBPROT operatörünü kullanarak bu bölgelerdeki dokuyu açığa çıkarmaktadır. Bunun sonucunda anahtar nokta çıkarım algoritmaları doku görüntüsünden anahtar nokta

ıkarabilmektedir. Őekil 2.13.(a)'da medikal grnt, 2.13.(b)'de oynanmıő grnt ve 2.13.(c)'de ise LBPROT grnts verilmektedir.



Őekil 2.13.(a) Medikal grnt (b) Oynanmıő medikal grnt (c) LBPROT grnt

LBPROT grntsnden anahtar noktaların ıkarımı: Oluőturulan doku grntsnden SIFT yntemi ile anahtar nokta ıkarılmaktadır.

Oynanmıő blge tespitinde anahtar noktaların eőleőtirilmesi: Doku grntsnden elde edilen anahtar noktaların eőleőtirilmesi ile oynanmıő blge tespiti gerekleőtirilmektedir. Anahtar nokta eőleőtirilmesi Renkli SIFT alıőmasında verildiėi gibidir.

2.2. Video zerinde Yapılan alıőmalar

Yukarıda maddeler halinde verilmiő alıőmalardan video zerinde yapılan alıőmaların detayları sırasıyla ilerleyen blmlerde verilmektedir.

2.2.1. LDB'ye Dayalı ereve Tekrarlama Sahteciliėi Tespiti

Tez kapsamında gerekleőtirilen bu alıőmada kodekten ve hareketten baėımsız bir ereve tekrarlama sahteciliėi tespit yntemi nerilmiőtir. nerilen yntemin literatre katkısı ise var olan alıőmalara gre sahtecilik tespitinde daha dők alıőma zamanı ve daha iyi tespit doėruluėu saėlamasıdır. alıőmada erevelerden zellik ıkarılmasında

2014 yılında Yang vd. tarafından [155] önerilen Yerel Fark İkili-LDB adıyla bilinen yeni bir ikili tanımlayıcı kullanılmıştır. Çerçevelerden çıkarılan LDB özellik vektörleri benzer çerçeveleri belirleyebilmek için karşılaştırılmakta ve sonrasında Sahtecilik uzaklığı olarak tanımlanan yeni bir yöntem benzer çerçevelere uygulanarak tekrar eden alt gruplar arasındaki gerçek uzaklık belirlenmektedir. Özellik çıkarımı ve gerçek uzaklığın belirlenmesi aşamalarından sonra elde edilen çerçeve çiftleri aday çerçeve çiftleri olarak belirlenmektedir. Son aşamada ise yöntem bu çiftleri aralarındaki PSNR değerine göre çok benzer, benzer ve az benzer olarak üç sınıfa ayırmaktadır. Sınıflardaki eleman sayısına göre de yöntem test videosunun sahte kısmını belirlemektedir. LDB yöntemi verildikten sonra aşağıda önerilen yöntemin detaylarından bahsedilecektir.

2.2.1.1. Yerel Fark İkili-LDB

2014 yılında Yang vd. tarafından LDB olarak adlandırılan yeni bir tanımlayıcı önerilmiştir [155]. Bu yeni özellik tanımlayıcı yöntemi, benzer ikili özellik çıkarım yöntemlerine kıyasla daha yüksek ayırt edicilik sağlamaktadır [156-159].

LDB, görüntüyü ızgaralara böler ve tanımlayıcı oluşturmak için ızgaraların ortalama piksel değerlerini ve birinci dereceden türevlerini kullanır. Varsayalım ki I görüntüsü $n \times n$ boyutundaki ızgaralara bölünsün. Özellik çıkarım yönteminde her bir hücreden bilgi çıkarılır ve tipik bir özellik elde etmek için bu hücrelere ikili test uygulanır. F hücrelerden bilgi çıkarmak için kullanılan bir fonksiyonu göstermektedir. (2.38) bağıntısında ise τ ikili testi ve i, j mevcut görüntüdeki hücreleri temsil etmektedir. τ fonksiyonu iki parametre alıp bu değerleri karşılaştırarak ikili bilgi üretmektedir.

$$\tau(F(i), F(j)) := \begin{cases} 1, & \text{if } ((F(i) - F(j)) > 0) \wedge i \neq j \\ 0, & \text{diğer} \end{cases} \quad (2.38)$$

F fonksiyonu [155] tarafından hesaplama hızından dolayı ortalama fonksiyonu olarak belirlenmiştir. Ortalama fonksiyonu bilgi çıkarmak için her bir hücreye uygulanmaktadır. Ayrıca özellik çıkarım yönteminin esnekliğini arttırmak için I görüntüsünün birinci dereceden türevi de değerlendirilmektedir. F fonksiyonu her bir hücre için üç değer geri

döndürmektedir. (2.39)'da verilen $I(i, k)$ i. hücrenin k. pikselini ve m ise hücredeki piksel sayısını göstermektedir.

$$F = \left\{ \frac{1}{m} \sum_{k=1}^m I(i, k), \text{Gradient}_x(i), \text{Gradient}_y(i) \right\} \quad (2.39)$$

Burada F fonksiyonun ilk döndürdüğü değer i. hücrenin ortalama piksel değeridir. $\text{Gradient}_x(i)$ ve $\text{Gradient}_y(i)$ ile gösterilen son iki değer ise sırasıyla x ve y yönündeki i'hücresinin bölgesel türevinin ortalama değerleridir.

Özellik çıkarım algoritması, her bir hücre için üç sonuç alır ve (2.38) 'de verilen ikili karşılaştırma ile ızgaralama sonucunda oluşan hücrelere karşılık gelen bu sonuçları karşılaştırır. LDB tanımlayıcısı bu şekilde $3n^2(n^2 - 1)/2$ (toplam karşılaştırma sayısı) adet ikili değerden elde edilir.

Algoritmada özellik çıkarımı için uygun ızgaralama boyutunun seçilmesi bir başka problemdir. Eğer hücrelerin boyutu küçükse, tanımlayıcının dayanıklılığı daha düşük olur, ancak daha fazla ayrıntı yakalayabilir. Hücre boyutunun büyük olması durumunda ise tam tersi durum söz konusu olur. Bu sebeple yazarlar çoklu-ızgaralamayı tercih etmişlerdir. Örneğin bir görüntü 2x2 ve 3x3'lük hücrelere bölünmekte ve bu bölümler sonucunda elde edilen ikili sonuçlar birleştirilerek bir tanımlayıcı oluşturulmaktadır.

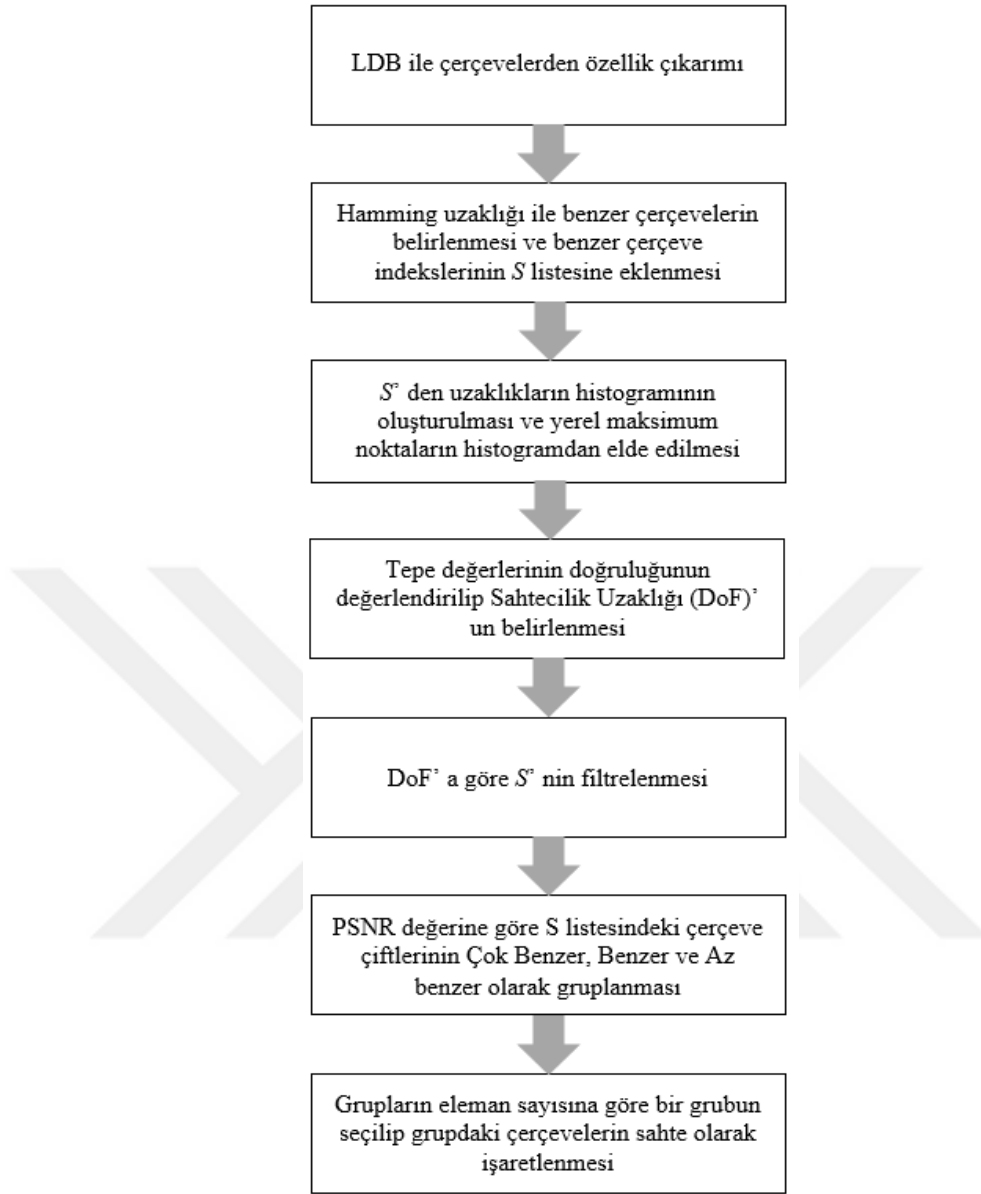
Bu çalışmada her bir çerçeve 2x2, 3x3 ve 4x4'lük hücrelere bölünmüş ve elde edilen ikili sonuçlar birleştirilerek 1x486 bit boyutunda bir özellik vektörü elde edilmiştir. 18 bit 2x2'lik hücrelerin kıyaslanması ile 108 ve 360 bit sırasıyla 3x3 ve 4x4'lük hücrelerin kıyaslanması sonucu elde edilmiştir.

Şekil 2.14.'te önerilen yöntemin genel yapısı verilmektedir. Algoritma Çerçevelerden Özellik Çıkarılması, Sahtecilik Uzaklığının Belirlenmesi ve Benzer Çerçevelerin Gruplanması olmak üzere üç aşamadan oluşmaktadır. Öncelikli olarak algoritmada video çerçevelere ayrıştırılmakta ve çerçevelerden LDB ile özellik çıkarılmaktadır. Özellik vektörleri benzer çerçeveleri belirlemek için kullanılmaktadır. Kopyalanan ve çoğaltılan video dizileri arasındaki uzaklık daha sonra benzer çerçeveler listesini kullanan Sahtecilik Uzaklık metodu kullanılarak tahmin edilir. Bu metot benzer çerçeveleri giriş olarak alır ve kopyalanan ve tekrarlanan bölümler arasındaki uzaklığı belirler. Bunun sonucunda belirlenen uzaklığı ihlal eden benzer çerçeve çiftleri benzer çerçeveler listesinden çıkarılır. Son aşama olarak ise algoritma benzer çerçeveleri PSNR değerlerine göre çok benzer,

benzer ve az benzer olarak üç sınıfa ayırır. Sınıflardaki eleman sayısına göre de sahte diziyi belirlemektedir. Önerilen yöntemin her bir aşaması aşağıda sırasıyla verilmektedir.

Çerçevelerden Özellik Çıkarımı: Algoritmanın bu aşamasında yöntem videoyu çerçevelere ayırıştırıp çerçevelerden özellik çıkarmaktadır. Çerçevelerden ikili bilgi çıkarırken LDB kullanılmakta ve özellik vektörleri arasındaki Hamming uzaklığına bakılarak benzer çerçeveler belirlenmektedir. Algoritmanın ilk kısmı sonucunda en çok benzeyen çerçeve çiftlerinin indisleri ikinci kısma girdi olarak verilmektedir. Bu aşamanın detayları aşağıda verilmektedir.

V test videosu $V = \{V^i | i = 1 \dots N\}$ olmak üzere N çerçeveden oluşsun. Algoritmada her bir V^i 'nin x ve y yönündeki birinci türevleri G_x^i ve G_y^i hesaplanmaktadır. Çerçeve ve türevleri sırasıyla 2x2, 3x3 ve 4x4'lük ızgaralarla bölünür ve her bir hücreden ortalama değerler hesaplanır. Örneğin mevcut çerçeve 2x2'lik ızgara ile eşit boyutta dört hücreye bölünsün ve (A, B, C, D) her bir hücre için ortalama piksel değerini göstere. Birinci türevi alınmış çerçeveler G_x^i ve G_y^i 'nin ortalama piksel değerleri ise sırasıyla (A_x, B_x, C_x, D_x) ve (A_y, B_y, C_y, D_y) olsun. Yöntemde A, (B, C, D) ile B, (C, D) ile ve C'de D ile kıyaslanarak toplamda 6 bitlik bir ikili bilgi elde edilmektedir. Aynı kıyaslama (A_x, B_x, C_x, D_x) ve (A_y, B_y, C_y, D_y) için yapıldığında çerçevenin ve türevlerinin 2x2'lik ızgara ile bölünmesi sonucunda toplamda 18 bitlik bir bilgi elde edilmektedir. 3x3'lük hücreler düşünüldüğünde her bir hücre kendisinde dışında kalan tüm hücrelerle karşılaştırılmaktadır. Dolayısıyla ilk hücre sekiz hücreyle, ikinci hücre yedi hücreyle bu şekilde karşılaştırma yapılmaktadır. Toplamda $(8 \times 9)/2 = 36$ kıyaslama yapılmaktadır.

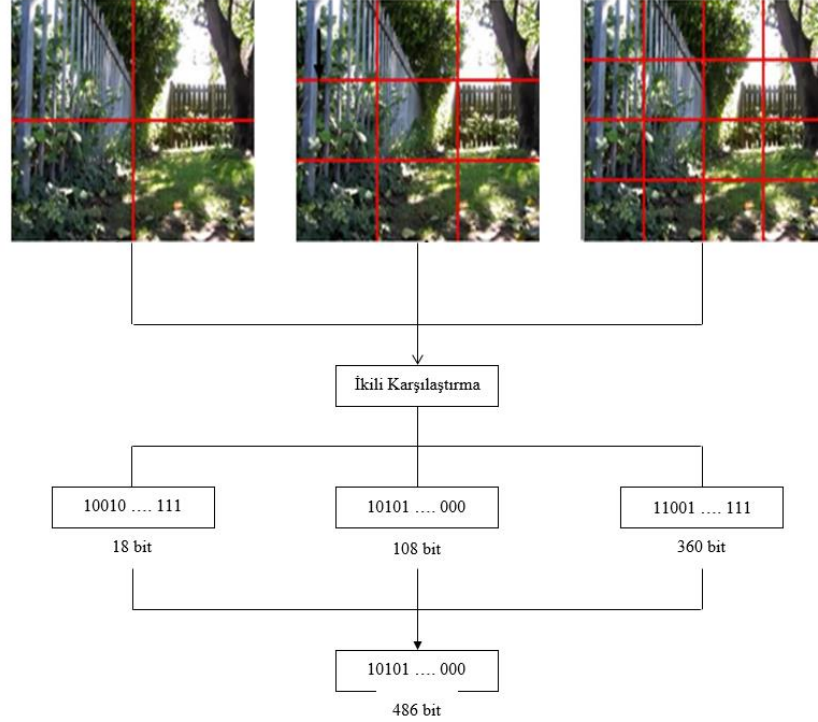


Şekil 2.14. Önerilen yöntemin akış diyagramı

Bu karşılaştırmanın V^i , G_x^i ve G_y^i hücreleri için yapıldığı düşünülürse 3×3 'lük hücreler için toplamda $36 \times 3 = 108$ bit elde edilmektedir. 4×4 'lük hücrelerde aynı yöntem izlendiğinde 360 bitlik bir ikili bilgi elde edilmektedir. Bunun sonucunda ise her bir çerçeveden $1 \times 486(18 + 108 + 360)$ boyutunda bir özellik vektörü elde edilmektedir. Şekil 2.15.'te özellik vektörünün elde edilişi gösterilmektedir.

Çerçevelere karşılık gelen özellik vektörleri $F = \{F^i | i = 1 \dots N\}$ ile temsil edilsin. Her bir özellik vektörü 486 ikili değer içermekte ve algoritmada çoğaltılan çerçeve dizilerinin benzer özellik vektörüne sahip olacağı düşüncesinden yola çıkılmaktadır. Çünkü

aynı çerçevelere karşılık gelen iki özellik vektörü sıkıştırma kalanlarından dolayı eşit olmayacaktır.



Şekil 2.15. Özellik vektörünün hesaplanması

Yöntemde özellik vektörleri kullanılarak benzer çerçeveler belirlenmektedir. Her bir özellik vektörü kendisi dışında kalan özellik vektörleriyle kıyaslanmaktadır. Vektörler ikili değerler içerdiğinden ötürü kıyaslamada Hamming uzaklığından faydalanılmaktadır. Özellik vektörünün F^i olduğunda bu vektörün benzerlik araştırması yapılırken vektör F^{i+w} 'dan F^N 'ye kadar olan tüm vektörlerle karşılaştırılacaktır. Algoritmada komşu çerçeveler benzer vektörlere sahip olacağından bir vektör kendisinden w adet sonraki vektörle kıyaslanacaktır. (2.40)'da verilen şarta göre F^i ile F^j kıyaslanmakta ve F_k^i i. vektörün k . elemanını göstermektedir. Eğer vektörlerin karşılıklı elemanlarından farklı olanlarının sayısı önceden tanımlanmış bir eşik değeri t 'yi geçmezse, algoritma iki vektörün benzer olduğunu varsayar ve indis değerlerini benzer çerçeveler listesi S' 'ye eklenir.

$$\text{If } \sum_{k=1}^{486} (F_k^i \oplus F_k^j) \leq t, \text{ ekle } (i, j) \text{ yi } S' \text{ ye} \quad (2.40)$$

Bu aşamanın son kısmında sahtecilik uzaklığının belirlenebilmesi için benzer çerçeve çiftlerini içeren S diğer aşamaya girdi olarak verilir.

Sahtecilik Uzaklığının Belirlenmesi: Algoritmanın bu aşaması dört adımdan oluşmaktadır. Şekil 2.16.'da Sahtecilik Uzaklığı Belirlenmesi algoritmasının genel yapısı verilmektedir.

Adım 1: Yöntem benzer çerçeve listesi S'yi kullanarak kopyalanan ve tekrarlanan çerçeve dizileri arasındaki uzaklığı belirler. Liste birinci sütunda kopyalanan çerçevelerin, ikinci sütunda tekrarlanan çerçevelerin indislerini tuttuğu iki sütundan oluşmaktadır. Algoritmada bu çerçeve indisleri arası uzaklık belirlenip S listesine üçüncü sütun olarak eklenmektedir. S listesinin yeni yapısı (2.41)'de görülmektedir.

$$S = [S^1 \ S^2 \ |S^1 - S^2|] \quad (2.41)$$

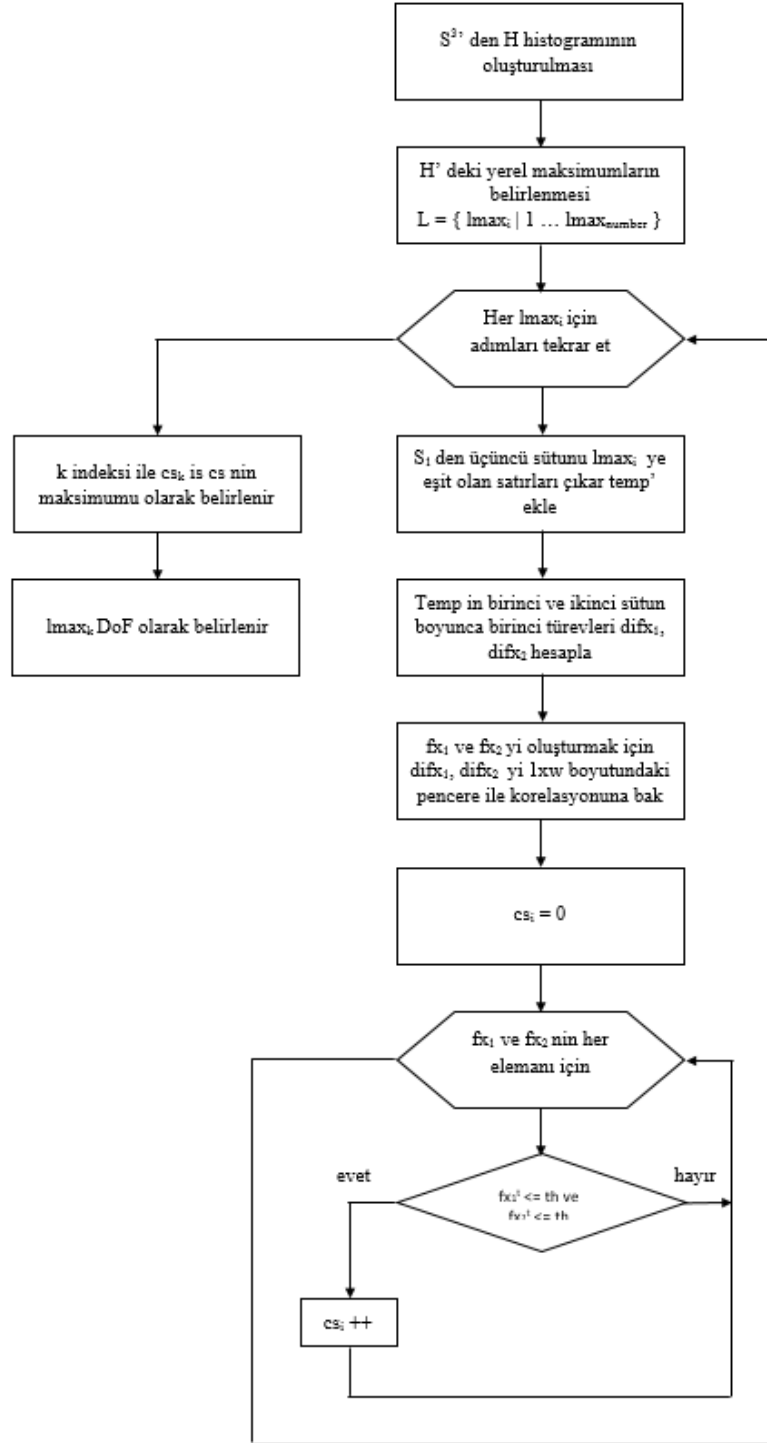
Burada S^1 ve S^2 sırasıyla birinci ve ikinci sütunu gösterirken üçüncü sütunda benzer çerçeve indisleri arasındaki mutlak uzaklıklar yer almaktadır.

Adım 2. Bu adımda uzaklıkların frekansını belirleyebilmek için üçüncü sütun S^3 'teki uzaklıkların histogramı H hesaplanır. Histogramdaki yerel maksimumlar L listesine eklenir. $L = \{lmax_i | i \in [1 \dots lmax_{number}]\}$ 'de $lmax_i$ 'ler yerel maksimumları göstermektedir. Örneğin mevcut histogramın dört maksimum içermesi, S listesindeki uzaklık değerlerinden en sık dört tanesi ile karşılaşılıyor anlamına gelmektedir.

Adım 3. Algoritma her bir lokal maksimumun doğruluğunu belirlemek için değerlendirir. Örneğin değerlendirilecek olan nokta $lmax_i$ olsun. S listesinden indis değerleri arasındaki farkı $lmax_i$ 'ye eşit olan benzer çerçeve çiftlerinin indisleri çıkartılır. S'deki bu filtreleme işlemi için (2.42) kullanılmaktadır.

$$temp = S(S(:,3) \equiv lmax_i), 1:2) \quad (2.42)$$

Burada $S(:,3) \equiv lmax_i$ işlemi, üçüncü sütunda $lmax_i$ içeren satırın 1:2 işlemiyle bir ve ikinci sütunundaki değerleri yani indisleri döndürür. Dolayısıyla temp listesi indisleri arasındaki uzaklığı $lmax_i$ 'ye eşit olan çerçeve indislerini içerir. Yöntemde temp'in sırasıyla birinci ve ikinci sütun boyunca türevleri hesaplanarak $difx_1$ ve $difx_2$ vektörleri elde edilir.



Şekil 2.16. Sahtecilik Uzaklığı Belirleme algoritması akış diyagramı

Eğer sahtecilik işlemi bir test videosunda yapılmışsa benzer çerçevelerin indis değerleri ardışık olmalıdır. Örneğin, eğer $lmax_i = 40$ ise ve 10-30 arasındaki çerçeveler 50 ile 70 arasına kopyalanmış ise temp listesi $temp = \{10\ 50, 11\ 51, 12\ 52, 13\ 53, 14\ 54, \dots, 30\ 70\}$ şeklinde ardışık çerçeve çiftlerini

içermelidir. Bunun sonucunda $difx_1$ ve $difx_2$ sırasıyla $\{1\ 1\ 1\ \dots\ 1\}$ ve $\{1\ 1\ 1\ \dots\ 1\}$ olarak elde edilir. Bununla birlikte temp listesindeki tüm değerler sıkıştırma artefektlerinden dolayı her zaman doğru olmayacağı gibi bazı çerçeve çiftleri de algoritma tarafından $temp = \{10\ 50, 12\ 52, 13\ 53, 15\ 55, \dots, 30\ 70\}$ 'de olduğu gibi tespit edilmemiş olabilir. Bu durumda birinci ve ikinci sütunların türevleri $\{2\ 1\ 2\ \dots\ 1\}$ ve $\{2\ 1\ 2\ \dots\ 1\}$ olmaktadır.

Adım 4. Bu adımda algoritmada birinci türevler değerlendirilmekte ve $lmax_i$ 'nin doğruluğuna karar verilmektedir. $1 \times w$ boyutundaki bir pencere oluşturulur ve tepe değerinin doğruluğunu belirlemek için $difx_1$ ve $difx_2$ üzerinde kaydırılır. Her adımda pencere ve $difx_1$ ve $difx_2$ arasındaki öklid uzaklığı ayrı ayrı hesaplanır ve fx_1 ve fx_2 'ye sırasıyla eklenir. Eğer tepe değeri sahtecilik uzaklığına karşılık geliyorsa pencere türevlere uyacak ve fx_1 ve fx_2 'nin elemanları eşik değeri th 'den küçük olacaktır.

(2.43) bağıntısı fx_1 ve fx_2 'ye uygulanarak yerel maksimum değerinin doğruluğu belirlenecek ve $lmax_i$ için doğruluk skoru cs_i hesaplanacak ve bu değer corr listesine eklenecektir.

$$\begin{aligned} cs_i &= \sum_{t=1}^{size(fx)} (fx_1^t \leq th) \wedge (fx_2^t \leq th) \\ corr_i &= [lmax_i\ cs_i] \end{aligned} \quad (2.43)$$

Yukarıda verilen bu aşamalar diğer tepe değerleri için de tekrarlanacak ve corr listesi güncellenecektir. Yerel maksimum değeri yani en yüksek cs_i 'ye sahip değer Sahtecilik uzaklığı, DoF, olarak belirlenecektir. Önerilen yöntemde S listesindeki benzer çerçeveleri filtrelerken (2.44)'te görüldüğü gibi DoF'u kullanmaktadır.

$$S = S(S(:,3) \equiv DoF, 1:2) \quad (2.44)$$

Benzer Çerçevelerin Gruplanması: Algoritmanın bu aşamasında S listesindeki benzer çiftler Çok benzer, benzer ve az benzer çerçeveler olmak üzere üç gruba ayrılmaktadır. Çerçeve çiftlerini gruplamak için ise PSNR'den faydalanılmaktadır. PSNR değeri iki görüntünün benzerliği hakkında bilgi vermektedir. Yöntemde gruplama amacıyla (2.45)'te verilen değer aralıkları kullanılmaktadır. Çok Benzer, Benzer ve Az Benzer listeleri benzerliklerine göre çerçeve çiftlerini içerir.

$$\begin{aligned}
(t_1 \leq PSNR < t_2) &\rightarrow \text{Az Benzer} \\
(t_2 \leq PSNR < t_3) &\rightarrow \text{Benzer} \\
(PSNR \geq t_3) &\rightarrow \text{Çok Benzer}
\end{aligned} \tag{2.45}$$

S’de ki her benzer çerçeve çifti için PSNR hesaplanır ve çıkan PSNR değerlerine göre çerçeve çiftleri gruplanır. Örneğin gruplama işleminden sonra Çok Benzer listesi benzer çerçeveleri içeriyorsa yöntem bu çerçeve çiftlerini sahte çerçeve olarak işaretler. Aksi takdirde benzer ya da az benzer gruptaki çerçeve çiftlerinden hangisinde daha fazla eleman varsa yöntem o gruptaki çerçeve çiftlerini sahte olarak işaretler.

2.2.2. İkili Özelliklere Dayalı Çerçeve Tekrarlama\Yansıma Sahteciliği Tespiti

Literatürde Çerçeve Tekrarlama sahteciliği tespiti için önerilmiş yöntemler benzer algoritmik yapıya sahip olduğu için aynı problemten etkilenmektedir. Bu yöntemler tekrarı yapılacak olan çerçeve grubu, eklenmeden önce ayna yansıması alınmış ise bu şekilde yapılmış sahtecilikleri tespit edememektedir. Wang vd. tespit aşamasında korelasyon bilgisinden faydalandığı için orijinal çerçeve ile ayna yansıması şeklini aynı içeriğe sahip olsalar bile benzetememektedir [119]. Yine renk histogramına dayalı yöntemde de aday çerçeveler ikinci aşamada karşılıklı blok benzerliğine bakılarak elenmekte, yansıtılmış bloklar da örtüşmeyeceği için bu yöntem de tespitte başarılı olamamaktadır [120]. Diğer bir çalışma olan çerçevelerden SVD ile özellik çıkarıp bu özelliklerin mesafesine bakan yöntemde çerçeve ile yansıtılmış şeklinin mesafeleri benzer çıkmayacağı için yine çalışma tespit de yetersiz kalacaktır [121]. Şekil 2.17.’de çerçeve yansımasına bir örnek verilmiştir. Şekil 2.17.(a) ve 2.17.(b) sırasıyla orijinal ve sahte videoları göstermektedir. Şekil 2.17.(b)’deki sahte video orijinal videodaki ikinci, üçüncü ve dördüncü çerçevelerin sırasıyla ayna yansıması alınarak oluşturulan M2, M3 ve M4 çerçevelerinin altıncı çerçeveden sonra orijinal videoya eklenmesi ile oluşturulmuştur.

Bu çalışmada benzer çalışmalarda elde edilen çalışma zamanı ve tanılama performansı değerlerinin iyileştirilmesi hedeflenmiştir [160]. Aynı zamanda çalışma literatürde video sahteciliği alanında yeni bir atak olan çerçeve yansıması atağını duyurmaktadır. [119-121] çalışmaları çerçeve yansıması atağını tespit edemezken önerilen yöntem çerçeve yansıma atağını etkin bir şekilde tespit edebilmektedir.

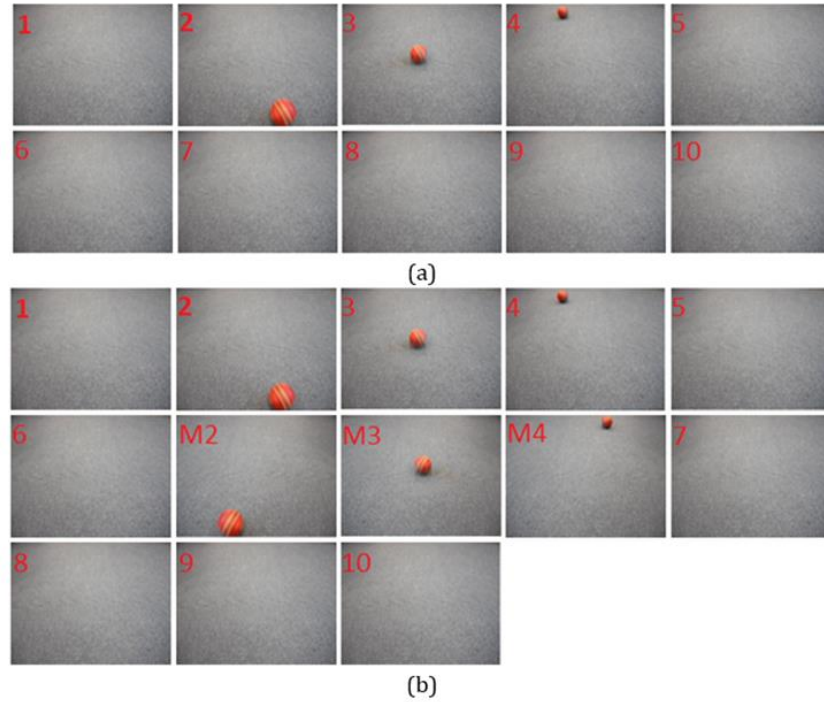
Yöntem yansıma yapılarak tekrarlanmış ya da sadece tekrarlama yapılmış çerçevelerinin tespitini dört aşamada gerçekleştirmektedir: Çerçevelerden özellik çıkarımı, Özelliklerin benzerliği kullanılarak aday çerçeve seçimi, PSNR ile aday çerçevelerin elimine edilmesi ve tespit sonuçlarının iyileştirilmesi için yapılan son işlem operasyonları. İlk aşamada yöntemi video sıkıştırma kalanlarına daha dayanıklı hale getirebilmek için ikili çerçevelerden özellik çıkarılmaktadır. Kötü niyetli bir kullanıcı videodaki bir bölümü gizlemek ya da çoğaltmak için değiştirir ve sonrasında videoyu çeşitli kodek özellikleri ile yeniden kaydeder. Bu nedenle sahte videodaki iki çerçeve tamamen aynı olsa bile bu çerçevelerin piksel değerleri yeniden sıkıştırmadan dolayı farklılık gösterecektir. Bunun içinde çerçevelerden bilgi çıkaran özellik çıkarım yönteminin sıkıştırma kalanlarına karşı dayanıklı olması gerekmektedir. Yeniden sıkıştırmadan da sonra sahte videodaki aynı çerçevelerden benzer özelliklerin elde edilmesi gerekmektedir. Bu çalışmada yeniden sıkıştırmaya karşı dayanıklı özellikler elde edebilmek için bir ikili özellik çıkarım yöntemi önerilmiştir.

İkinci aşamada çerçeveler arası benzerliği belirleyebilmek için çerçevelerden çıkarılan özellikler arasında Öklid uzaklığına bakılmaktadır. Bu adımdaki temel hedef, kaba bir yaklaşımla videodaki benzer çerçeve gruplarını belirleyebilmektir. Eğer videodaki bir çerçeve grubu kopyalanıp yine aynı video içerisindeki başka bir yere yapıştırılmışsa, kopyalanıp yapıştırılan gruptaki karşılıklı çerçeveler aynı olacağı için bu çerçevelerin ikili özellikleri de benzer olacaktır. Öklid uzaklığı çıkarılan her ikili özellik arasında hesaplanacaktır. Eğer ikili özellik bir başka ikili özelliğe benziyorsa bu özelliklere karşılık gelen çerçeveler şüpheli çerçeve olur ve bu çerçevelerin çerçeve numaraları aday listesine eklenir. Bu listenin ilk sütununda kaynak çerçeve numaraları, ikinci sütununda ise aday çerçevelerin çerçeve numaraları bulunmaktadır. Aday listesi sahtecilik işlemindeki şüpheli çiftleri içermektedir.

Üçüncü adımda sahtecilik yapılan aday çerçeve çiftleri olarak belirlenmiş çiftleri daha da elimine edilerek nihai sonuca varılmaktadır. Önceki aşamada elde edilen çerçeve çiftleri benzer ikili özellik göstermektedir. Bu aşamada ise aday listesindeki benzer çerçeve çiftleri arasında PSNR hesaplanmaktadır. Eğer iki çerçeve yüksek PSNR değeri verirse algoritma bu çiftlerin aynı olduğuna karar vermekte aksi takdirde bu çift aday listesinden kaldırılarak yanlış adaylar elimine edilmektedir.

Son adım olarak tespit algoritmasının performansı, kalan adayların son işlem operasyonlarına tabi tutulmasıyla daha da geliştirilir. Bu aşamada algoritmada son işlem

operasyonunun gerçekleştirilebilmesi için aday listesi ilk sütun değerine göre m gruba bölünmektedir. Her bir grup bir kaynak çerçeveyi ve bu kaynak çerçeveye benzer aday çerçeveleri temsil eder, böylece farklı kaynak çerçevelerinin sayısı m değerini belirler. Örneğin önceki aşamadan elde edilen aday listesi (2, 23) (2, 24) (2, 25) (3, 24) (3,25) (3, 26) olması durumunda m değeri iki olacaktır. Kaynak çerçeve 2 iken aday çerçeveler 23, 24 ve 25 olmakta, kaynak çerçeve 3 iken ise 25 ve 26 olmaktadır. Eğer bir kaynak çerçeve örnekte olduğu gibi aday çerçeve listesindeki birden çok çerçeveye benzer çıkarsa algoritma kaynak ve aday çerçeveler arasında PSNR hesaplamaktadır. Aday çerçevelerden kaynak çerçeve ile en iyi PSNR değeri veren çerçeve kaynak çerçevenin tekrarı yani sahte çerçeve olarak seçilmektedir. Şekil 2.18.'de genel akışı verilen algoritmanın detayları aşağıda adımlar halinde verilecektir.

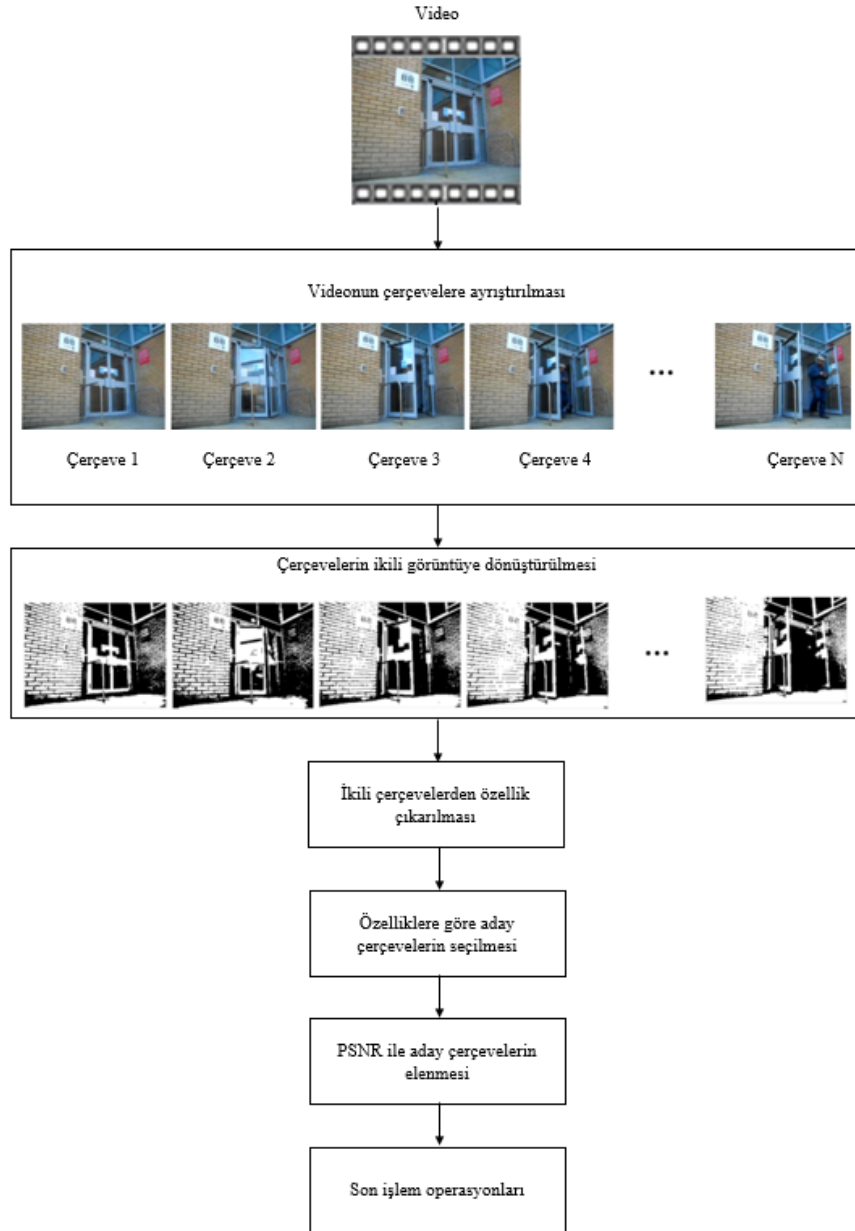


Şekil 2.17. (a) Orijinal (b) Ayna yansıması ile oluşturulmuş sahte video

Adım 1. (Çerçevelerden özellik çıkarımı) Yöntem video V 'yi çerçevelere ayırtırmakta ve her bir çerçeveyi ikili hale dönüştürmektedir. Otsu yöntemi siyah beyaz piksellerin varyansını en aza indirgeyecek optimum eşik seçmek için kullanılmaktadır. V videosu $H \times W$ boyutunda N tane çerçeveden $V = F^1 \dots F^N$ oluşsun. Optimum eşik

değerinin hesaplanması için Otsu algoritması her bir çerçeveye uygulanmakta ve çerçeveler $B^1 \dots B^N$ olmak üzere ikili görüntülere dönüştürülmektedir.

Aşağıdaki özellik çıkarma algoritması, video dizisini özellik vektörleri ile temsil etmek için her ikili görüntüye uygulanmaktadır. Özellik çıkarma algoritmasının genel yapısı Şekil 2.19.(a)'da verilmektedir.



Şekil 2.18. Önerilen yöntemin akış diyagramı

Mevcut çerçeve B^i olsun ve bu çerçevenin satırları $B_1^i \cdots B_H^i$ ile gösterilsin. Her satır kendisinden bir sonraki satırla olan Hamming uzaklığıyla temsil edilir. Diğer bir deyişle, ikili görüntünün satırları kendisinden bir sonraki satırları ile XOR işlemine tabi tutulur ve sonuçtaki 1 sayısı, geçerli satırı temsil etmek için kullanılır. Bunun sonucunda özellik çıkarım algoritması her bir satırı $[0 \cdots W]$ arasında bir değer ile temsil eder.

Özellik vektörü $X^i = x_1^i \cdots x_H^i$ ile temsil edilsin. (2.46) bağıntısında i . çerçeve olan B^i 'ye özellik vektörünün hesaplanması verilmektedir. Özellik vektörünün ilk elemanı $x_1^i = 0$ her zaman sıfır olacaktır.

$$x_j^i = \sum_{k=1}^W (B_{jk}^i \oplus B_{j-1,k}^i), j \in \{2 \cdots H\}, x_j^i \in \{0 \cdots W\} \quad (2.46)$$

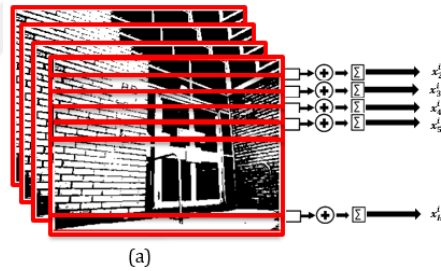
En kötü durumda iki ardışık satırın tüm bit değerleri farklı olacağı için sonuç 1 olacak ve özellik vektörünün değeri maksimum W olacaktır. Minimum ise ardışık satır elemanlarının hepsinin birebir aynı olması durumunda sıfır olacaktır. (2.46)'da verilen XOR sonucunda yapılan toplama işleminin yansıma işlemi sonucunu değiştirmeyeceği için önerilen yöntem yansıma ataklarına karşı dayanıklıdır. Önerilen özellik çıkarma yöntemi sayesinde bir çerçeveden elde edilen özellik vektörü ile bu çerçevenin ayna yansıması alınmış şekinden elde edilen özellik vektörü aynıdır. Şekil 2.19.(b)'de bu durum örneklenmektedir. Şekilden de görüldüğü üzere iki vektör B_1^1, B_2^1 arasındaki toplam fark ile ayna yansıma şekilleri mB_1^1, mB_2^1 arasındaki fark eşit olarak 4 bulunmuştur.

Bu adım sonucunda tüm ikili çerçeveler $B^1 \cdots B^N$ için özellik vektörleri $X^1 \cdots X^N$ elde edilmektedir.

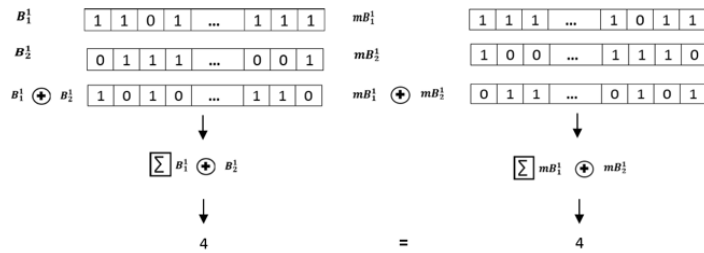
Adım 2. (Özelliklerine göre aday çerçeve seçimi) Bu adımda benzer çerçeveler özellik vektörleri kullanılarak belirlenmektedir. Aday listesini belirleyebilmek için her bir özellik vektörü kendisi dışındaki tüm özellik vektörleriyle kıyaslanmaktadır. Algoritma özellik vektörleri arasındaki benzerliği test edebilmek için X^i nin $X^{i+1} \cdots X^N$ vektörleriyle olan Öklid uzaklıklarını karşılaştırır. (2.47) bağıntısında özellik vektörleri arasındaki Öklid uzaklığının hesaplanması verilmektedir.

$$\sqrt{\sum_{j=1}^H (x_j^i - x_j^{i+1})^2} < t_d \quad (2.47)$$

Burada i kaynak $i+1$ ise aday çerçeveyi göstermektedir. Algoritma eğer iki vektör arasındaki uzaklığı önceden belirlenen eşik değeri t_d 'den az bulursa $(i, i + 1)$ çiftini aday listesi C 'ye ekler. Önerilen yöntem en ideal eşik değerini belirleyebilmek için vektörler arasındaki en kötü (maksimum) ve en iyi (minumum) Öklid uzaklıklarını değerlendirir. İki vektör arasındaki minimum uzaklık, en iyi senaryo için 0 olurken, buna karşılık en kötü durumda iki vektörün tüm elemanlarının farklı değerlere sahip olması gerekmektedir. Eğer çerçevelerin birinden çıkarılmış özellik vektörünün elemanlarının hepsi bir (W 'ya eşit) diğer çerçeveden elde edilen özellik vektör elemanlarının hepsi sıfır (minumum) ise bu iki vektör arasındaki Öklid uzaklığı $W\sqrt{(H-1)}$ olup maksimum olmaktadır. Eğer iki vektör arasındaki uzaklık yaklaşık olarak $0.01W\sqrt{(H-1)}$ ise özellik vektörleri elemanları arasında %1 oranında bir farklılık gösterdiği söylenebilir. Önerilen yöntem optimum eşik değeri t_d 'yi belirleyebilmek için $[0.01W\sqrt{(H-1)} - 0.012W\sqrt{(H-1)}]$ değer aralığını test etmektedir. Yani vektörler arasında en fazla %12'lik bir değişim olacağı öngörülmektedir.



(a)



(b)

Şekil 2.19. (a) Özellik çıkarma yönteminin genel yapısı (b) Önerilen ikili özellik çıkarma yönteminin yansımadan bağımsızlığın gösterimi

Aynı zamanda hem bu çerçeveler arasında hem de kaynak çerçeveye benzerliği araştırılan aday çerçevenin ayna yansıması alınmış şekli arasında PSNR hesaplanmaktadır.

Hesaplanan iki PSNR değerinden maksimum olanı üçüncü elaman olarak C listesine eklenmektedir. (2.48) bağıntısında C listesinin k. satırı gösterilmekte ve listede (i, i + 1) çifti aday olarak görülmektedir. Bu bağıntıda G renkli çerçeveyi gri seviyeye çeviren mirr ise parametre olarak verilen görüntünün ayna yansıması oluşturan fonksiyondur.

$$C^k = \left(i, i + 1, \max \left(PSNR \left(G(F^i), G(F^{i+1}) \right), PSNR \left(G(F^i), \text{mirr} \left(G(F^{i+1}) \right) \right) \right) \right) \quad (2.48)$$

Bu aşamada (kaynak, aday) çiftlerini belirleyebilmek için her özellik vektörü kendisi dışındaki tüm özellik vektörleriyle kıyaslanır ve bu adım sonucunda aday listesi C'nin her bir satırında benzer çerçeve indisleri ve aralarındaki PSNR değeri bulunur.

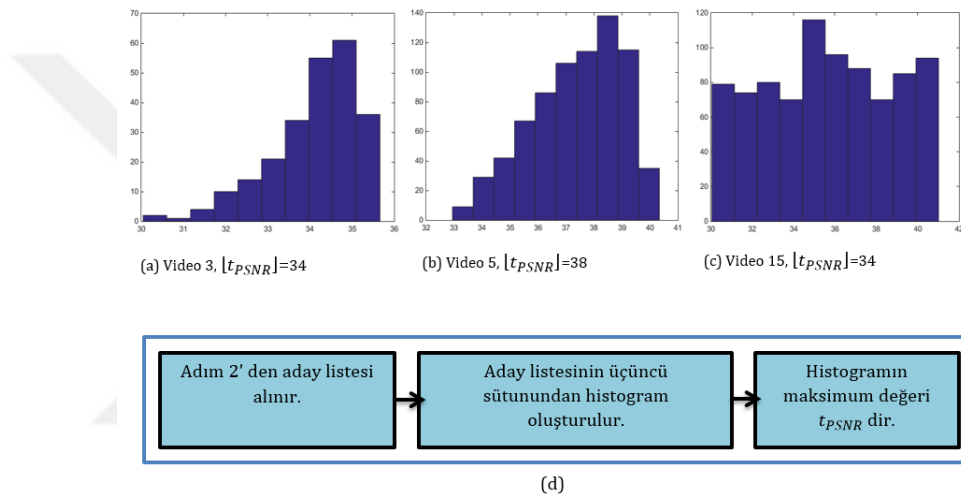
Adım 3. (Aday çerçevelerin PSNR değeriyle elenmesi) Yukarıda da bahsedildiği gibi aday ve kaynak çerçeve arasındaki PSNR değeri aday listesinin üçüncü sütununda bulunmaktadır. Bu adımda C'nin üçüncü sütunun histogramı oluşturulmakta ve yanlış adayları elimine etmek için histogramın tepe değerine karşılık gelen PSNR değeri seçilmektedir. Histogram tepe değeri t_{PSNR} olsun. Benzer çerçeveler yaklaşık olarak eşit PSNR verdiği ve aday listesi de benzer çerçeveler içerdiğinden histogramın tepe değeri algoritma tarafından kullanılacak olan PSNR eşliğini belirleyecektir.

Listedeki çiftlerin PSNR değerleri t_{PSNR} ile kıyaslanmaktadır. Eğer bu çiftlerden birinin PSNR değeri $[t_{PSNR}]$ 'den az ise bu çift aday listesinden kaldırılmaktadır. (2.49) ile yanlış adaylar elimine edilmektedir.

$$\forall i: \text{Eğer } (C_3^i \leq [t_{PSNR}]) \quad C^i \text{ satırını } C \text{ listesinden çıkar} \quad (2.49)$$

PSNR histogramının maksimum değeri sıkıştırma kalanları hakkında bilgi vermektedir. Aday listesindeki çerçeve çiftlerinin bir kısmı gerçek çerçeve tekrarı yapılmış çiftler olacağı için videoda kayıplı sıkıştırma yapılmış olmasaydı bu çiftler arasındaki PSNR değeri sonsuz olacaktı. Bununla birlikte tüm sahte videolar sahtecilik yapıldıktan sonra sahtecilik izlerini yok etmek için yeniden sıkıştırılmaktadır. Bu yüzden çerçeveler aynı olsa dahi PSNR değerleri hiçbir zaman sonsuz olmayacaktır. Histogramın tepe değerine karşılık gelen PSNR değeri, kayıplı sıkıştırma nedeniyle çoğaltılan çerçeveler arasındaki PSNR'nin en az t_{PSNR} olması gerektiğini gösterir.

Şekil 2.20.(a), 2.20.(b) ve 2.20.(c) bazı test videoları için $[t_{PSNR}]$ değerinin belirlenmesini göstermektedir. Yöntemde adım 1 ve adım 2 bu test videolarına uygulamakta ve aday listesi elde edilmektedir. Bu test videoları için elde edilen aday listelerinin üçüncü sütunlarından elde edilen histogramlar Şekil 2.20.(a), 2.20.(b) ve 2.20.(c)'de gösterilmektedir. Histogramların tepe değerleri test videoları için PSNR eşiği $[t_{PSNR}]$ değerine karşılık gelmektedir. Dolayısıyla yöntem tarafından test videoları için elde edilen t_{PSNR} sırasıyla 34, 38 ve 34'tür. Şekil 2.20.(d)'de de t_{PSNR} değerinin dinamik olarak belirlenmesinde uygulanan yöntemin akışı verilmektedir.



Şekil 2.20. (a)(b)(c) Video 3, Video5 ve Video 15 için $[t_{PSNR}]$ değerinin belirlenmesi (d) $[t_{PSNR}]$ değerinin belirlenmesinde kullanılan algoritma adımları

Adım 4.(Son İşlem Operasyonları) Aday listesi C, bundan önceki adımların sonucunda elde edilen adayların büyük bir listesinden en olası adayları içermektedir. Bu aşamada uygulanacak olan son işlem operasyonları sonucunda sahte videodaki sahte video dizisinin yeri nihai olarak bulunacaktır.

Aday listesi C satırlarının ilk elemanları aynı olacak şekilde $G^1 \dots G^m$ olmak üzere m gruba bölünmektedir. Sonrasında her bir grupta en yüksek PSNR değerine sahip grup seçilmektedir. Her bir grup için seçilen en iyi satır yani en yüksek PSNR'ye sahip olan satır $m \times 3$ boyutundaki sonuç listesi R'ye eklenmektedir. Daha sonra R listesinden PSNR değerleri kaldırılarak listede sadece aday çerçevelerin indisleri ve aralarındaki uzaklık bilgisi tutulmaktadır. (2.50)'de verilen bağıntı her bir gruba uygulanır. $best_PSNR(G^i)$ fonksiyonu grup indisini parametre olarak alır ve geriye en yüksek satırı döndürür. Seçilen

satır g olmak üzere g_1 , g_2 ve $|g_1 - g_2|$ 'den oluşan üç değere sahiptir. İlk iki değer sırasıyla kopyalanıp ve yapıştırılan çerçevelerin indislerine, son değer ise bu çerçeve indisleri arasındaki uzaklığa karşılık düşmektedir.

$$\forall i: i \in 1 \dots m, \quad g = \text{best_PSNR}(G^i), \quad R^i = [g_1 \ g_2 \ |g_1 - g_2|] \quad (2.50)$$

Eğer video çerçeve tekrarlama sahteciliği ile sahte hale getirilmişse tekrarlanan çerçeve çiftlerinin indisleri arasındaki uzaklık eşit olacaktır. Dolayısıyla sonuç listesi R 'nin üçüncü sütunundaki kaynak ve hedef çerçeve indisleri arasındaki fark eşit olacaktır. Eğer aynı video üzerinde iki farklı sahtecilik yapılmışsa bu sefer R listesi üçüncü sütununda iki farklı uzaklık değeri içerecektir. Örneğin bir videoda (10,20) ve (30,40) arasındaki çerçeveler sırasıyla yine aynı videoda (110,120) ve (140,150) çerçevelerine kopyalanmış olsun. Bu durumda R listesi üçüncü sütununda 100 ve 110 uzaklık değerlerini içerecektir. Listede bu iki uzaklık değeri dışında uzaklıklar, ya kodek tabanlı kalanlar ya da yanlış aday çiftleri sonucu oluşmuştur.

Algoritma R 'nin üçüncü sütunundaki th 'dan daha fazla tekrar eden uzaklık değerlerini bulur. Varsayalım ki bu kuralla $d_1 \dots d_k$ olan k tane uzaklık değeri elde edilmiş olsun. Sahte çerçeve grubunu belirleyebilmek için aşağıda verilen kaynak kod R listesine uygulanmaktadır. R listesinin i . satırındaki üç eleman sırasıyla $R(i, 1)$, $R(i, 2)$ ve $R(i, 3)$ temsil edilirken e yöntemin hata toleransını ifade etmektedir.

```

for i=1 to m
  for j=1 to k
    if (R(i, 3)>=dj-e) && (R(i, 3)<=dj+e)
      R(i, 2)= R(i, 1) + dj;
      break;
    end if
  end for
end for

```

Bu adım sonucunda PSNR'ler ve uzaklıklar kullanılarak sahte çerçeve dizisinin nihai yeri belirlenmekte ve R listesi kopyalanıp yapıştırılan çerçeve indislerini içermektedir.

2.2.3. BoW Modeline Dayalı Çerçeve Tekrarlama Sahteciliği Tespiti

Literatürde Çerçeve Tekrarlama sahteciliğinin tespiti için önerilmiş olan yöntemlerin kıyaslanmasında iki önemli kriter kullanılmaktadır: İcra zamanı ve tanılama performansı. Bu yöntemler iki aşamadan oluşmaktadır: Çerçeveler arası benzerliğe bakarak aday benzer video dizilerinin seçimi ve seçilen dizilerdeki çerçeveler arası blok eşleştirme yöntemi ile uzaysal benzerliğin araştırılması. [119, 120]'deki çalışmalarda korelasyon ve renk histogramları benzer video dizilerinin seçiminde kullanılmıştır. Fakat bu yöntemler ciddi çalışma zamanları gerektirmektedir. [121]'deki çalışma ise SVD yönteminden özellik çıkarma aşamasında faydalanmış ve her çerçeveden yalnızca bir değerli özellik çıkarmıştır. Özellik vektör boyutunun küçüklüğü ise yöntemin diğerlerine göre daha hızlı olmasını sağlamıştır. Fakat [121] yönteminin çalışma zamanı başarısı, test olarak kullanılan videoda birden fazla benzer video dizilerinin bulunması ile ters orantılı olarak düşmektedir. Aynı zamanda sahteciliğin ardından uygulanan yeniden kodlama aşaması da çerçeveler arası benzerliği artıracak şekilde de statik sahneleri daha çok benzer hale getireceği için yöntemlerin hem çalışma zamanını yükseltecek hem de tanıma performansını düşürecektir.

Literatürdeki benzer çalışmalara [119-126] kıyasla daha yüksek tanılama performansına ve daha kısa çalışma zamanına sahip yeni bir çerçeve tekrarlama sahteciliği tespit yöntemi geliştirilmiştir. Önerilen yöntem sahteciliğin tespitinde BoW modelini kullanmaktadır. BoW modeli literatürde ilk kez metin dökümanlarının analizinde kullanılmıştır. Daha sonraki yıllarda ise Sivic ve Zisserman BoW modelini görüntü ve video erişim alanlarına uygulamışlardır [163-165]. BoW modeli uygulama aşamasında özellik çıkarımı, sözlük üretimi için k-means öbekleme ve görsel sözcüklerden histogram üretimi olmak üzere üç adımdan oluşmaktadır [165]. Son yıllarda, video olay sınıflandırması, insan eylemi tanıma, sahne ve video sınıflaması gibi video işlemenin çeşitli alanlarında BoW modeli oldukça fazla kullanılmıştır [166-169]. Bununla birlikte literatürde Çerçeve Tekrarlama sahteciliği tespitinde BoW modelinden şimdiye kadar faydalanılmamıştır. Görüntüde çeşitli anahtar noktası çıkarım yöntemleri kullanılarak görüntü üzerinde yapılan sahtecilikleri tespit eden çalışmalar mevcuttur. Bu çalışmalarda anahtar noktası çıkarımı SIFT ile yapılmakta ve anahtar noktalardan elde edilen tanımlayıcıların eşleştirilmesi ile görüntüdeki sahte alanlar işaretlenmektedir. Aynı yaklaşım n çerçeveli bir videoda çerçeve tekrarlama sahteciliğinin tespiti için uygulanmış olsaydı, bir çerçeveden anahtar noktalar çıkarılacak bu çerçeveyi takip eden diğer çerçevelerden çıkarılan anahtar noktalarla sırasıyla eşleştirilecekti. Tekrarlanan çerçevelerin tespit edilebilmesi için her bir çerçeveden çıkarılan anahtar noktaların diğer

çerçevelerle eşleştirilmesi oldukça büyük hesaplama karmaşıklığını beraberinde getirecektir. Videonun n çerçeveden oluştuğu düşünülürse eşleştirme fonksiyonunun $n(n+1)/2$ kez koşması gerekmektedir.

Çalışmada çerçevelerden çıkarılan anahtar noktaların tanımlayıcıları ile bir görsel sözlük oluşturulmuş ve her bir çerçeveye bu sözcükler kullanılarak bir BoW özelliği verilmiştir. Tekrarlanan çerçeveleri belirleyebilmek için çerçevelerden çıkarılan özellik vektörleri arasındaki benzerliğin hesaplanmasında Öklid uzaklığından faydalanılmıştır. Böylece çerçeveler arası benzerlik, daha çok hesaplama karmaşıklığı gerektiren anahtar nokta eşleştirme yerine BoW özelliklerinin karşılaştırılması ile belirlenmiştir.

Önerilen yöntem Çerçeve Tekrarlama sahteciliğinin tespitinde BoW modelini literatürde kullanan ilk çalışmadır. Her çerçeve görsel sözcükler biçiminde temsil edilir ve çerçeveler arasındaki benzerlik Öklid uzaklığı ile hesaplanır. Çalışmada aynı zamanda sıkıştırma kalanlarına karşı daha dayanıklı olan anahtar noktalarını filtrelemek için Ölçek Seçimi adlı yeni bir algoritma önerilmiştir. Algoritma önceden tanımlanmış bir eşik değeri s 'den daha büyük olan ölçek değerlerine sahip olan anahtar noktalarını seçerek yöntemi video kodek kalanlarına karşı daha dayanıklı hale getirir. Ölçek Seçim algoritması s değerini dinamik olarak belirlemektedir. Ardından ölçek ve yön bilgisi eklenerek fazladan iki boyut artırılmış çerçevelerden çıkarılmış SIFT anahtar nokta tanımlayıcıları k-means++ algoritması ile öbeklenmektedir. Oluşturulan öbeklerin merkezleri görsel sözcüklere karşılık düşmekte, çerçeveler ise görsel sözcük histogramları ile temsil edilmektedir. Tüm çerçeveler için histogram oluşturulduktan sonra benzer çerçeve gruplarını yani aday sahte çerçeve çiftlerini belirleyebilmek için histogramlar arasındaki Öklid uzaklığına bakılmaktadır. PSNR dinamik olarak belirlenen bir eşik değerinden büyükse aday çerçeveler sahte olarak işaretlenmektedir. Videonun karakteristik özellikleri, benzerlik karşılaştırması için kullanılan PSNR eşiğini belirlemektedir. BoW modeline dayalı önerilen video sahtecilik tespit yöntemi benzer çalışmalarla kıyaslandığında daha iyi sonuçlar vermektedir. Yöntem ayrıca diğer çalışmalara göre çalışma zamanını oldukça düşürmektedir.

Aşağıda çerçevelerin BoW modeli ile temsili verildikten sonra önerilen yöntemin detayları verilecektir.

2.2.3.1. BoW Modeli

BoW modelinin görüntülere adaptasyonu esnasında uygulanan dört adım aşağıdaki şekilde verilebilir:

- Görüntüden anahtar noktalarının elde edilmesi
- Anahtar noktaları etrafından yerel tanımlayıcıların elde edilmesi
- Tanımlayıcıların kullanılacak olan bir kümeleme yöntemi yardımıyla görsel kelimelere karşılık düşürülmesi.
- Oluşturulan sözlükte var olan görsel kelimelerin, görüntü içerisinde temsil edilme frekanslarını kullanarak görüntüye karşılık düşürülen BoW özelliğinin oluşturulması.

Adımların ayrıntıları aşağıdaki gibi özetlenmektedir.

Varsayalım ki I test görüntüsü olsun. I üzerinde bir anahtar noktası çıkarma algoritması (SIFT gibi) uygulanır ve ilk iki adım ile I 'daki ilgili anahtar noktaları temsil eden yerel tanımlayıcılar elde edilir. $\{d^t | t = 1 \dots T\}$ T çıkarılan tanımlayıcı sayısı olmak üzere d^t , I görüntüsünün t . tanımlayıcısını temsil etmektedir.

BoW modelinde ilk iki adımı eğitim setindeki tüm görüntülere uygulandıktan sonra üçüncü adımda görsel bir sözlük oluşturulur. Eğitim setindeki görüntülerden çıkarılan tanımlayıcılar yerel özellik veri tabanına eklenir. Sonrasında yerel özellik veri tabanına k-means öbekleme algoritması uygulanarak öbek merkezleri elde edilir. Öbek merkezleri $C = \{c_1, c_2, \dots, c_k\}$ ile gösterilmekle beraber k eğitim setindeki görsel kelime sayısını göstermektedir.

Son aşamada ise test görüntüsündeki her bir tanımlayıcı $\{d^t | t = 1 \dots T\}$ kendisine en yakın görsel sözcüğün c_k öbeğine dahil edilip o görsel kelime ile etiketlenilmektedir. (2.51) bağıntısında etiketleme işlemi gösterilmektedir.

$$\underset{k}{\operatorname{argmin}} \quad (\operatorname{dist}(c_k, d^t)) \in C \quad (2.51)$$

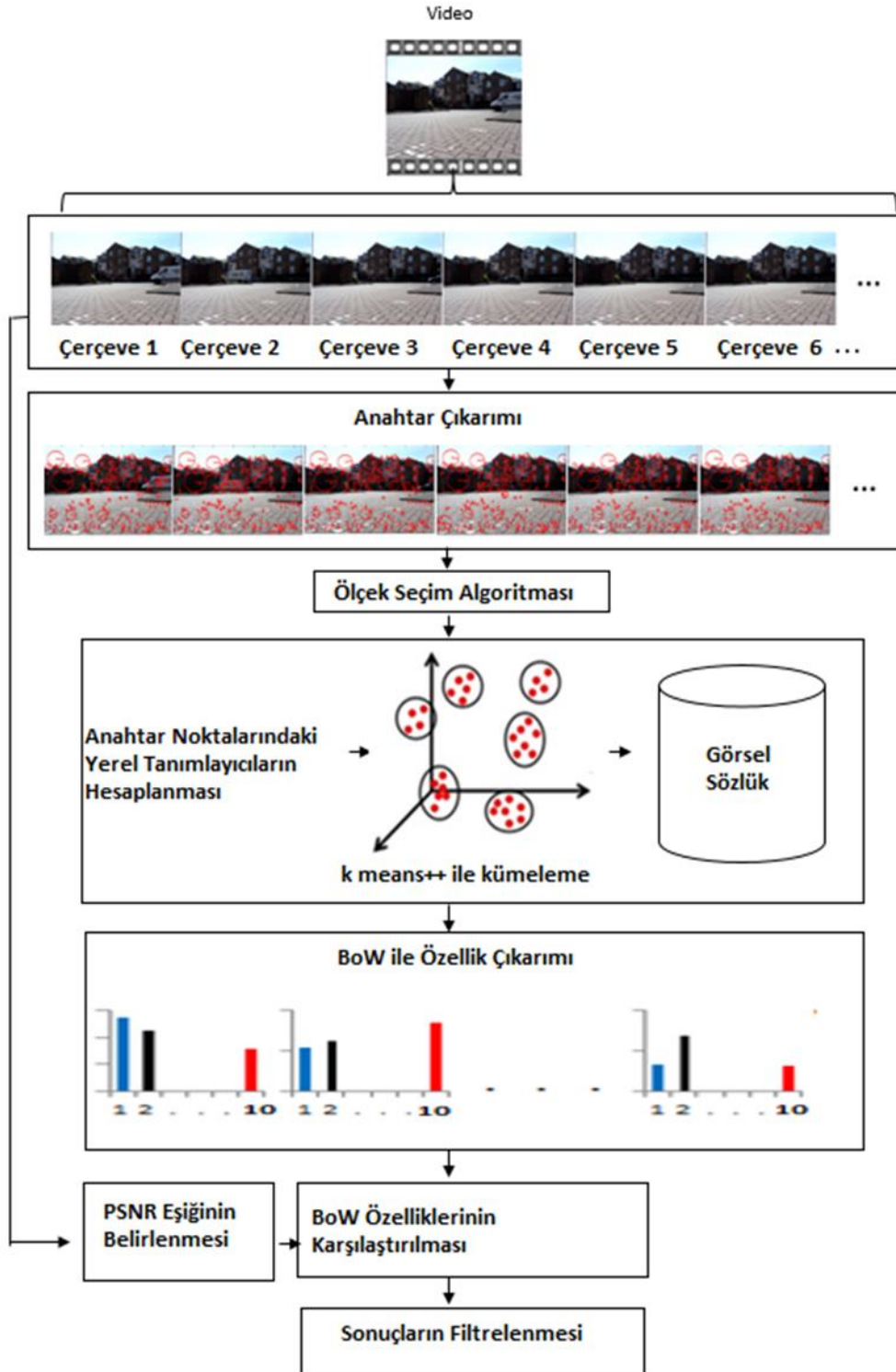
dist fonksiyonu öbek merkezi c_k ile o anki tanımlayıcı d^t arasındaki Öklid uzaklığını geri döndürmektedir. d^t en küçük Öklid uzaklığını veren öbek merkezi yani görsel sözcük ile etiketlenmektedir. Test görüntüsü I 'da ki tüm tanımlayıcılar bir görsel sözcük ile etiketlenildikten sonra görüntüdeki görsel sözcüklerin frekansı hesaplanarak özellik vektörleri üretilmektedir. I görüntüsünden çıkarılan özellik vektörü f 'nin boyutu $1 \times k$ olmaktadır. Dolayısıyla f_1 'de ilk görsel sözcük c_1 ile etiketlenen tanımlayıcı sayısını ifade etmektedir.

Önerilen yöntemde, yukarıda anlatılan görüntü erişim sistemlerinde kullanılan BoW modeli video sahteciliklerinden Çerçeve Tekrarlama sahteciliğinin tespiti için kullanılmıştır. Şekil 2.21.'de önerilen yöntemin genel akış diyagramı verilmektedir. Şekilden de gözlemlenebileceği gibi video çerçeveleri eğitim kümesi olarak kullanılmıştır. Yöntemde öncelikle tüm çerçevelerden anahtar noktalar çıkarılmakta bu noktalar Ölçek Seçim algoritmasına göre filtrelenmekte, ardından üretilen anahtar noktalar için yerel tanımlayıcılar elde edilmektedir. Daha sonra, k-means ++ algoritması ile tüm çerçevelerden elde edilen tanımlayıcılar öbeklenir ve öbek merkezlerin karşılık düştüğü görsel sözcüklerin bulunduğu sözlük üretilir. Her bir video çerçevesi için BoW özellik vektörü oluşturulur. Her çerçeve için elde edilen özellik vektörü, ilgili çerçevedeki görsel sözcüklerin histogramıdır. Özellik vektörlerinin benzerliği ise seçilen bir uzaklık ölçme kriteri ile değerlendirilecektir. Benzer vektörler sahtecilik olduğu düşünülen çerçeveleri temsil etmektedir. Aday çerçevelerin istenilen sayıya düşmemesi durumunda ise ikinci bir benzerlik kriteri devreye sokulmaktadır. Doğruluğu artırabilmek adına kullanılacak olan ikinci eleme kriteri, çerçeve benzerliğinin testi için PSNR'den faydalanmaktadır. Önerilen yöntemin detayları aşağıda verilecektir.

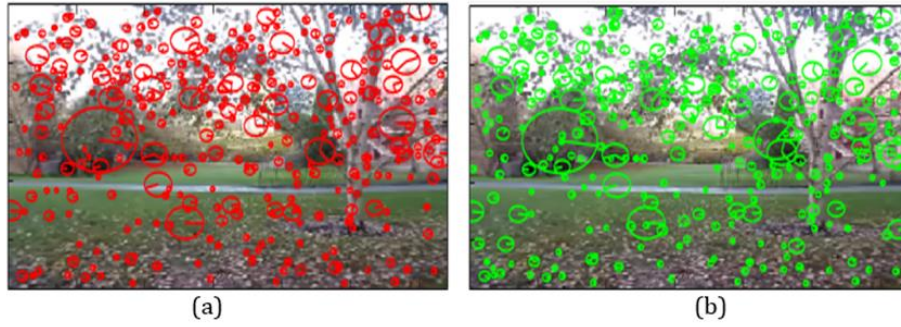
Önerilen yöntem görüntü tanıma alanında oldukça fazla kullanılan SIFT yöntemini kullanarak video çerçevelerinden anahtar noktalarını çıkartmaktadır. SIFT anahtar noktaları rotasyon, ölçekleme ve öteleme ataklarına karşı dayanıklıdır. Yöntemin kopyalanıp yapıştırılan çerçeveleri tespit edebilmesi için, video içerisindeki kopyalanıp yapıştırılan çerçevelerde yer alan anahtar noktalarının aynı olması beklenir. Herhangi iki çerçeve üzerinde var olan anahtar noktaları arasındaki benzerlik ise ilgili çerçevelerin birbirinin tekrarı olabileceği sonucuna varılmasını sağlayacaktır. Çalışmada, video içerisinde yer alan bütün çerçeveler üzerinde DoG yöntemi kullanılarak anahtar noktalar belirlenmektedir. Yöntemin sahteciliği tespit etme olasılığı, kopyalanıp yapıştırılan çerçevelerdeki anahtar noktaları arasındaki tutarlılığa bağlıdır. Ancak sahteciliğin yapılmasının ardından videonun yeniden bir kodekle kodlanıyor olması, çerçeveler aynı olsa bile sıkıştırmaya bağlı farklılaşmaların oluşmasına sebep olacaktır. Şekil 2.22.(a) ve 2.22.(b)'de video içerisinde kopyalanıp yapıştırılan iki çerçeveye ait görüntü verilmektedir. Şekilden de gözlemlenebileceği gibi, iki çerçeve aynı olsa bile farklı anahtar noktaları içermektedir. Bu nedenle, sıkıştırma yöntemlerine en dayanıklı anahtar noktalarının seçilmesi, benzer çerçeveleri tespit edecek olan yöntemin başarısını iyileştirecektir. Bir anahtar noktasının ölçek değeri sıkıştırmaya karşı dayanıklılığını gösterecektir. Çalışmada

yapılan arařtırmalarda, kodeklerden kaynaklanan kalanlara karřı en dayanıklı anahtar noktalarının seilmesi iin yeni bir yntem nerilmiřtir. Test videosundaki erevelerde yer alan anahtar noktalarından belirli bir lek deęeri ve yukarısına sahip olanlarının seilmesi iin, lek Seim algoritmasından yararlanılacaktır. Algoritmaya iliřkin detaylar ařaęıdaki řekildedir.





Şekil 2.21. Sistemin Genel Yapısı



Şekil 2.22. (a) (b) Kopyalanan ve yapıştırılan çerçevelerdeki anahtar noktaları

Ölçek Seçim Algoritması : Algoritma test videosu V için en iyi ölçek değerini belirleme aşamasında **Sahte_Video_Olustur**, **Video_Isle** ve **En_Iyi_Olcek** olmak üzere üç ana fonksiyon kullanılmaktadır. İlk fonksiyon test videolarından sahte videolar oluştururken, ikinci fonksiyon bu sahte videoları işleyerek ölçek değeri hakkında bilgi veren bir histogram döndürmektedir. Son fonksiyon ise her sahte video için en iyi ölçek değerini belirlemektedir. Fonksiyonların detayları sırasıyla aşağıda verilecektir.

Sahte_Video_Olustur: Ölçek Seçim algoritması test videosu V 'den en iyi ölçek değerini belirleyebilmesi için $FV^1 \dots FV^g$ olmak üzere g tane rastgele sahte video oluşturmaktadır. **Sahte_Video_Olustur(.)** fonksiyonu her bir sahte videoyu oluşturur. Çerçeve sayısı n olan V test videosunun çerçeveleri $v^1 \dots v^n$ ile temsil edilsin. Yöntem videodan bir kaynak çerçeve s^k seçer ve rastgele seçilmiş l tane çerçeve $r_1^k \dots r_l^k$ 'lerin yerine sahte video FV^k 'yı oluşturmak için bu çerçeveyi koyar. Yalnız bu kaynak çerçeve l tane yere yazılırken video sıkıştırılmamış formatta kaydedilir. Sahte video oluşturulduktan sonra ise yine orijinal kodek ayarları kullanılarak reencode fonksiyonu ile sıkıştırılmaktadır. Diğer sahte videolarda aynı şekilde oluşturulmakta ancak kaynak çerçevenin yeri ve bu çerçevenin yapıştırıldığı yerler her seferde rastgele olarak belirlenmektedir. **Sahte_Video_Olustur(.)** fonksiyonunun yalancı kodu aşağıda verildiği gibidir.

```

function [ $V^k, r^k, s^k$ ] = Sahte_Video_Olustur ( $V^k$ )
begin
    temp = (rand(n));
     $s^k = v^{temp}$ 
     $r^k = []$ ;
    for i=1:l
        temp = (rand(n));

```

```

         $v^{temp} = s^k;$ 
         $r_i^k = temp;$ 
    end
    return [reencode( $V^k$ ),  $r^k$ ,  $s^k$ ];
end

```

Video_Isle : Ölçek Seçim fonksiyonu Video_Isle fonksiyonunu her bir sahte video FV^k , $k = 1 \dots g$ için çağırır ve $1xscale_{max}$ boyutundaki bir ölçek histogramı olan h 'yi günceller. Bu fonksiyon sahte video ve h histogramı olmak üzere iki parametre alır ve geriye h histogramının güncellenmiş halini döndürür. h histogramı ana fonksiyon olan Ölçek_Seçim fonksiyonunda tanımlanacaktır.

```

function Video_Isle ( $FV^k$ ,  $h$ ,  $s^k$ ,  $r^k$ )
begin
    ind = 0;
     $source_{kp} = extract\_keypoints(s^k)$  ;
    for i=1: l
        begin
            ind = En_Iyi_Olcek( $source_{kp}$ ,  $r_i^k$ );
             $h(ind) = h(ind) + 1$ ;
        end
    return h;
end

```

Varsayalım ki s^k ve r^k sırasıyla FV^k 'nin kaynak ve tekrarlanan çerçeveleri olsun. video_isle fonksiyonu s^k 'dan $extract_keypoints(\cdot)$ fonksiyonunu kullanarak anahtar noktaları çıkaracak ve bu noktaları $source_{kp}$ değişkeninde tutacaktır. best_scale fonksiyonu da FV^k 'nin tüm tekrarlanmış çerçeveleri için çağrılacaktır.

En_Iyi_Olcek($source_{kp}$, r_i^k): Fonksiyon öncelikli olarak r_i^k den anahtar nokta çıkarmakta ve sonrasında $source_{kp}$ ve r_i^k 'dan gelen tüm anahtar noktalarını ölçek değerine göre filtrelemektedir. İlk olarak Ölçek 1 alınır ve $source_{kp}$ ve r_i^k 'daki ölçeği bir ve birden büyük olan anahtar noktaları filtrelendir. $source_{kp}$ 'da filtrelenen anahtar noktaları $source$ filtrelenenler ise dest olarak adlandırılır ve eşleştirilir. Eğer tüm anahtar noktaları eşleştirilmişse fonksiyon geri dönüş değeri olarak eşleşmenin elde edildiği ölçek değerini verir. Aksi takdirde ölçek değeri bir artırılarak filtreleme ve eşleşme işlemleri tekrar ettirilir.

Algoritma tarafından ölçeklerin maksimum değeri için bir $scale_{max}$ değeri belirlenir. Eğer algoritmanın hesapladığı ölçek değeri $scale_{max}$ değerini aşarsa algoritma durur ve geri dönüş değeri olarak $scale_{max}$ 'ı döndürür. *En_Iy1_Olcek* fonksiyonunun işlevi yani source veya dest'deki filtrelenmiş tüm anahtar noktaların eşleştiği ilk ölçek değerini vermektir. Fonksiyonun yalancı kodu aşağıda verildiği gibidir.

```
function best_scale ( $source_{kp}, r_i^k$ )
begin
    scale = 1; temp = Extract_keypoints( $r_i^k$ );
    source = filter( $source_{kp}, scale$ ); dest = filter(temp, scale);
    If match(source, dest)
        return scale;
    else
        scale ++;
        if scale >  $scale_{max}$  return  $scale_{max}$ ;
    end
end
```

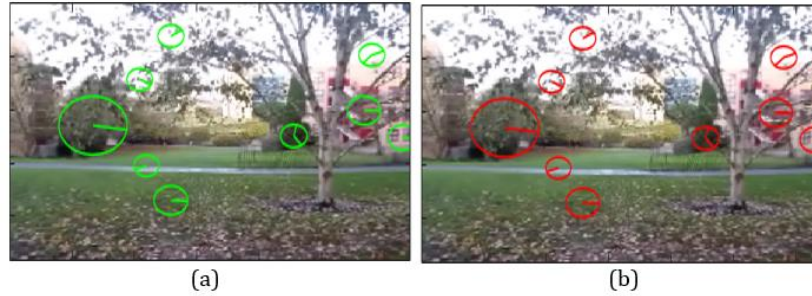
$source_{kp}$ ve r_i^k için best_scale fonksiyonunun geri dönüş değeri h histogramının indisidir. Sonuç olarak Ölçek Seçim fonksiyonu aşağıdaki yalancı kodda görüldüğü gibi üç fonksiyonla özetlenebilir.

```
function Olcek_Secim (V)
begin
    h = zeros(1,  $scale_{max}$ );
    for k=1: g
        begin
            [ $FV^k, r^k, s^k$ ] = Sahte_Video_Olustur(V);
            h = Video_Isle( $FV^k, h, s^k, r^k$ );
        end
    s = ind_of_max(h);
    return s;
end
```

Burada *ind_of_max*(h) fonksiyonu maksimum değerli elemanın indisini döndürmektedir. Ölçek seçim fonksiyonu s değeri olarak h histogramının maksimum değerine sahip olan indisini döndürür. Çünkü maksimum değerli indis, *En_Iy1_Olcek*(·) fonksiyonunda sahte video çiftlerinin çoğunda s'yi döndürdüğünü gösterir. Elde edilen s değeri değerlendirilen test videosu için en iyi ölçek değeri olarak alınmaktadır. Aynı zamanda bu değer sıkıştırılmış sahte videodan elde edildiği için ölçek değeri s veya s'den

büyük anahtar noktaları video sıkıştırma kalanlarına daha dayanıklı olmaktadır. Ölçek Seçim fonksiyonunun karmaşıklığı $O(g \times l)$ olmakla beraber bu değer sahte video sayısı g ve sahte çerçeve sayısı l 'ye bağlıdır.

Önerilen yöntem videodaki çerçevelerden ölçek değeri s ve s' 'den büyük olan anahtar noktalarını SIFT yöntemini kullanarak elde etmektedir. Şekil 2.23., Ölçek Seçim Algoritmasının etkisini göstermektedir. Şekil 2.23.(a) ve 2.23.(b) sırasıyla video içerisinde kopyalanan ve yapıştırılan çerçeveleri temsil etmektedir. Şekil 2.22.'de her iki çerçevede, video sıkıştırmasından kaynaklı olarak farklı yönlenim ve ölçeklerde anahtar noktaları üretildiği gösterilmişti. Bununla birlikte, sadece ölçek değeri 9 ve yukarısındaki anahtar noktaları seçildiğinde, kopyalanan ve yapıştırılan çerçevelerde tespit edilen anahtar noktalarının tutarlılık gösterdiği gözlemlenmektedir. Ölçek Seçim algoritması sayesinde seçilen anahtar noktaları video sıkıştırmalarına daha dayanıklı olmaktadır. Yöntem her çerçeveden s ve s' 'den büyük ölçek değerine sahip anahtar noktalarını belirlemekte ve bu anahtar noktalarının etrafındaki parlaklık değerlerini kullanarak tanımlayıcıları oluşturmaktadır.



Şekil 2.23.(a)(b) Kopyalanan ve yapıştırılan çerçevelerdeki dayanıklı anahtar noktalarının seçilmesi

SIFT tanımlayıcı belirleme algoritması kullanılarak her bir anahtar noktası için 1×130 büyüklüğünde tanımlayıcı vektörleri oluşturulmaktadır. SIFT yöntemi ile beraber 4×4 'lük bir bölge için, her bir gradyan yönündeki 8 bin değeri ile 1×128 'lik bir tanımlayıcı vektör oluşturulur. Önerilen yöntem aynı zamanda ölçek ve yönlenim değerlerini de tanımlayıcının sonuna ekleyerek 1×130 boyutundaki tanımlayıcıyı yapılandırır. Üretilen vektördeki ilk ve son değerler sırasıyla ölçek ve yönlenim değerlerine karşılık düşmektedir. Orijinal SIFT tanımlayıcısı ise orta kısımda yer alır. Önerilen yöntemde açık kaynak kodlu VIFeat kütüphanesi SIFT tanımlayıcı vektörlerinin oluşturulması için kullanılmıştır [170].

Bu kütüphane aracılığı ile üretilen SIFT tanımlayıcı vektörü elemanları [0-255] aralığına normalize edilmiş değerlerden oluşmaktadır. Fakat ölçek ve yönlenim bilgileri ise daha dar bir aralıkta değişmekte hatta yönlenim bilgisi negatif değerler alabilmektedir. Bu nedenle yöntemde ölçek ve yönlenim bilgilerinin değişim aralığı normalize edilerek tanımlayıcı vektörüne yerleştirilir.

Ölçek değeri s ve s' 'den büyük olan n anahtar noktası için yapılandırılan n adet tanımlayıcı vektörü, $d^1 \dots d^n$ ile gösterilsin. $(d_1^1 \dots d_1^n)$ ve $(d_{130}^1 \dots d_{130}^n)$ elemanları sırasıyla ölçek ve yönlenim bilgilerini temsil etmektedir. Algoritma öncelikle yönlenim ve ölçek değerlerini [0-1] aralığına normalize eder ve sonrasında SIFT tanımlayıcılarının maksimum değerli elemanı $\max(d_i^j), i = 2 \dots 129, j = 1 \dots n$ ile bu değerleri çarparak tüm değerleri aynı aralığa düşürmüş olur. K-means öbekleme işleminden önce bu min-max normalizasyon yönteminin uygulanmasıyla kümeleme performansını artırma hedeflenmektedir. [171]'de belirtildiği gibi, öbekleme sırasında Öklid uzaklık metriği kullanıldığında, öbekleme öncesi normalizasyon özellikle gerekmektedir. K-means ++ algoritması tarafından kullanılan Öklid uzaklığı, vektör elemanlarının büyüklüklerindeki değişime duyarlıdır.

Yöntem, video üzerindeki bütün çerçevelerden elde edilmiş olan seçili anahtar noktaları için anlatılan yöntemi kullanarak tanımlayıcı vektörleri elde eder. Çerçevelerden Sözlük oluşumu ve BoW özelliklerinin üretimi için uygulanacak olan aşamalar aşağıda verilmektedir.

Görsel sözlüğün oluşturulması ve BoW özelliklerinin çıkarımı: Yöntem, elde edilen yerel tanımlayıcıların kuantalamasını gerçekleştirerek görsel sözlüğü yapılandırır. Çalışmada tanımlayıcıları kuantalamak ve görsel sözcükleri üretmek için her veri noktasının öbek merkezine olan uzaklığını en aza indirgeyen k-means öbekleme yöntemi kullanılmıştır.

Lloyd algoritması veya k-means öbekleme yöntemi yaklaşık bir çözüm bulur [172], çünkü öbeklemede rastgele giriş NP zorluktur. K-means öbekleme yönteminin yaklaşık çözümü optimal kümeleme ile karşılaştırılmaz. Arthur vd. tarafından önerilen k-means ++ algoritması k-means öbekleme algoritmasının başlangıç değerlerini seçerek k-means algoritması tarafından elde edilen kötü öbekleme performansını elimine etmekte ve bir çözüm garanti etmektedir [173].

Mevcut test videosu için tüm çerçevelerden elde edilen tüm tanımlayıcılar k-means++ algoritması kullanılarak $C = \{c_1, c_2, \dots, c_k\}$ ile temsil edilen k tane öbeğe

ayrıştırılmaktadır. Öbeklerin merkezleri görsel sözlüğü oluşturan görsel sözcüklere karşılık gelmektedir. Dolayısıyla öbek sayısı aynı zamanda görsel sözlüğün boyutunu da göstermektedir. Sözlük içindeki kelimeler çerçeveleri görsel sözcüklerle temsil etmek için kullanılmaktadır. Öbek merkezleri, her tanımlayıcıya görsel bir sözcük atamak için BoW özelliği yapılandırması sırasında kullanılır. Her bir çerçeve için aşağıda gösterildiği gibi çıkarılmaktadır.

- Mevcut çerçevedeki her bir tanımlayıcı kendisine en yakın görsel sözcük c_i ye atanmaktadır.

- Çerçevenin tanımlayıcılarının hepsi görsel bir sözcükle etiketlenildikten sonra o anki çerçevenin görsel sözcük histogramı oluşturulur.

- Çerçeve için elde edilen histogram o çerçeve için BoW özelliğine karşılık gelmektedir. Örneğin özellik vektörünün ilk elemanı ilk görsel kelime olan c_1 'e atanan tanımlayıcı sayısına karşılık gelmektedir.

Tüm çerçeveler için BoW özellik vektörleri yukarıda anlatıldığı gibi oluşturulur. BoW özelliği aslında, videoda yer alan çerçevelerin sözlükte yer alan sözcüklerle ifade şekli olarak da düşünülebilir. Her bir çerçeveden elde edilen özellik vektörü boyutu büyüklüğü, görsel sözlükte yer alan kelime sayısına eş değerdir. Çalışma kapsamında deneysel sonuçlardan faydalanılarak görsel sözlüğün oluşturulmasında kullanılan küme sayısı 10 olarak belirlendiği için, çerçevelerden üretilen özellik vektörü boyutu büyüklüğü 1×10 'dur. Özellik vektörü çıkarıldıktan sonra çerçevelerden üretilen özellik vektörlerinin kullanımı ile sahtecilik tespitinin gerçekleştirilmesine sıra gelmektedir.

Özellik Vektörleriyle Sahtecilik Tespitinin Gerçekleştirilmesi: Yöntem ilk olarak çerçeve özelliklerini karşılaştırmakta ve ardından karşılaştırma sonucu üretilen aday çiftlerdeki hata eliminasyonunu gerçekleştirmektedir. Videodaki benzer çerçevelerin tespiti için öncelikle BoW özellik vektörlerinin kıyaslanması gerçekleştirilir. Ardından, üretilen aday çiftleri arasındaki hatalı eşleşmeleri elimine edebilmek amacı ile filtreleme işlemi uygulanmaktadır.

Özellik vektörleri arasındaki mutlak farkın karelerinin toplamının karekökünün, belirli bir eşik değeri ile karşılaştırılması ile benzerliğe karar verilmektedir. Aynı zamanda çerçeveler arası PSNR değeri kullanılarak da benzerlik kararı desteklenmektedir. $V = v^1 \dots v^n$ ile gösterilen ve n adet çerçeveden oluşan videodan elde edilen özellik vektörleri sırasıyla $B = b^1 \dots b^n$ ile gösterilsin. Her vektör, kendisinden sonraki vektörler ile kıyaslanarak Çerçeve Tekrarlama sahteciliği tespit edilmeye çalışılır. v^i ve b^i sırasıyla i .

çerçeve ve ona karşılık düşen özellik vektörünü temsil etsin. Özellik vektörü kendinden belirli bir komşuluktan sonraki vektörler ile kıyaslanacaktır. Çünkü yakın çerçeveler yakın özellik vektörleri üretmektedir. Varsayalım ki, her vektör kendisinden w adet sonraki vektörler ile kıyaslanacak olsun. Bu durumda, o an değerlendirilmekte olan b^i vektörü ilk olarak b^{i+w+1} vektörü ve geriye kalan $b^{i+w+2} \dots b^n$ vektörleri ile (2.52)'deki gibi kıyaslanacaktır. Burada th eşik değeri benzerlik testinde kullanılmaktadır.

$$\sqrt{\sum_{k=1}^{10} (b_k^i - b_k^{i+w+1})^2} \leq th \quad (2.52)$$

Benzerlik kararında etkili olacak olan PSNR testi ise benzer olduğuna (2.52) ile karar verilen b^i ve b^{i+w+1} özellik vektörlerinin karşılık düştüğü v^i ve v^{i+w+1} çerçeveleri arasındaki görsel benzerliği kontrol edecektir. Yöntem, (2.53)'te verildiği gibi eğer iki çerçeve arasındaki PSNR değeri th_{psnr} ile gösterilen eşik değerinden büyük ise, iki çerçevenin benzer olduğuna karar verecek ve bu ikiliyi aday listesi C 'ye ekleyecektir. Aksi takdirde bir sonraki çerçeveye ait özellik vektörü ile kıyaslamalar devam edecektir.

$$\forall v^i: PSNR(v^i, v^{i+w+1}) > th_{psnr} \text{ ekle } (v^i, v^{i+w+1}, PSNR(v^i, v^{i+w+1}))C \quad (2.53)$$

(2.53)'te verilen PSNR testinin kullanılabilmesi için her bir video için uygun eşiğin th_{psnr} 'in belirlenmesi gerekmektedir. Test videosu V için th_{psnr} değerinin dinamik olarak belirlenmesi aşağıda verildiği gibidir.

PSNR Eşiğinin Belirlenmesi: Yöntem Ölçek Seçim algoritmasının Sahte_Video_Olustur fonksiyonu aracılığıyla oluşturulmuş sahte videolardan $FV^1 \dots FV^g$ birini rastgele seçer. PSNR eşiğinin belirlenebilmesi için varsayalım ki FV^i , $FV^1 \dots FV^g$ arasından rastgele seçilmiş olsun. Bu sahte video Sahte_Video_Olustur(V) fonksiyonu ile kaynak çerçeve v^i nin kopyalanarak video içerisindeki rastgele seçilmiş l tane çerçeve $r^1 \dots r^l$ yerine yapıştırılması ile oluşturulmuştur. Ve sonrasında fonksiyon FV^i 'yi orijinal kodek ayarlarıyla tekrar sıkıştırmıştır. FV^i 'deki kaynak çerçeve ve bu çerçevenin tekrarı olan çerçeveler arasında $PSNR(v^i, r^1)$, $PSNR(v^i, r^2) \dots PSNR(v^i, r^l)$ olarak PSNR hesaplanır. Eğer test videosu birçok statik sahne içeren çerçevelerden oluşuyorsa bu videonun sıkıştırma oranı yüksek olacaktır. Tam tersi olarak videonun sahneleri çok hızlı

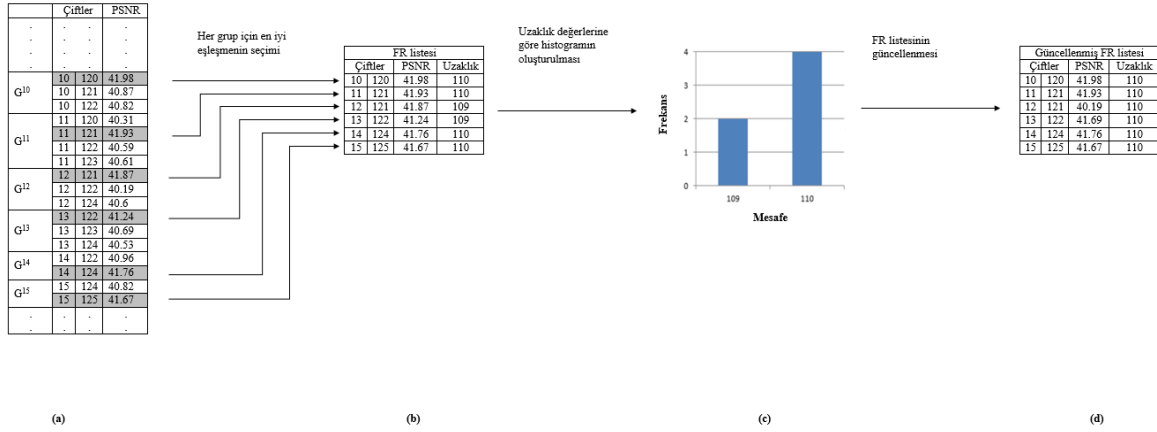
değişiyorsa ilk duruma göre bu videonun sıkıştırma oranı daha düşük olacaktır. Sahte video için kaynak ve tekrarlanan çerçeveler aynı olsa bile, sıkıştırma faktörü nedeniyle aralarındaki PSNR sonsuz olmayacaktır. Bunun için $(v^i, r^1), (v^i, r^2) \dots (v^i, r^l)$ çiftleri için elde edilen PSNR değerleri FV^i videosunun sıkıştırılması hakkında bilgi verecektir. Bu sebeple yöntem $PSNR(v^i, r^1), PSNR(v^i, r^2) \dots PSNR(v^i, r^l)$ değerlerini hesaplar ve (2.54)'te verildiği gibi $1 \times l$ boyutundaki t dizisine kaydeder.

$$t_j = PSNR(v^i, r^j), \quad j = 1 \dots l \quad (2.54)$$

t dizisinin en küçük değeri o anki test videosu için en kötü senaryo olduğu için bu değer $th_{psnr} = \min(t_j), j = 1 \dots l$ olarak seçilmektedir.

Aday listesindeki her satırda kopyalanan ve yapıştırılan çerçevelere ait indis değerleri ile bu çerçeveler arasındaki PSNR değerleri listelenmektedir. Yöntem aday listesini öncelikle sıralar ve listedeki satırları ilk elemanları aynı olacak şekilde gruplara ayırır. Her gruptaki ilk eleman aynı çerçeveyi referanslarken ikinci sütundaki değerler, ilk elemana benzer olan videodaki çerçevelerin indis değerleridir. Şekil 2.24.(a)'da aday listesinde var olan gruplardan biri gösterilmiştir. İlgili örnekte videodaki 10'uncu çerçeve ile benzer olan çerçeve indisleri gösterilmiştir. Gerçekte sahte videodaki 10'uncu çerçeve 120'inci çerçevenin üzerine kopyalanmıştır. Şekil 2.24.(a)'da görülüşü gibi video sıkıştırmasından kaynaklanan hatalı kararlar listede mevcuttur, (10, 121) ve (10, 122) gibi. Bu nedenle çalışma kapsamında önerilen yöntemin sonuçlarını iyileştirmek amacıyla hataları elimine etmek için yeni bir yöntem önerilmiştir.

Videoda yer alan i 'nci çerçeve ile benzer olan çerçevelerin oluşturduğu alt grup G^i ile gösterilsin. Yöntem, her gruptan tek bir satır belirleyecektir. İlgili satır, grupta maksimum PSNR değerini vermiş olan satır olacaktır. Her grubu temsil eden satır FR olarak ifade edilen ve birinci sonuç listesi olarak da adlandırabileceğimiz listeye eklenir.



Şekil 2.24. (a) Aday Listesi (b) FR Listesi (c) Uzaklık Histogramı (d) Güncellenmiş FR listesi

Şekil 2.24.(a)'daki örnek grup için ise en yüksek PSNR değerini veren (10, 120) çifti seçilecektir. FR ile gösterilen listeden seçilen çift, ilgili çift için PSNR değeri ve çiftteki çerçeveler arasındaki uzaysal uzaklık değeri tutulacaktır. İlk sonuç listesi olan FR'de her grup için seçilen en iyi adaylar mevcuttur. Fakat bazı durumlarda yöntem kaynak çerçeveyi sahteciliğin yapıldığı çerçeve ile eşleştirmek yerine bir önceki veya bir sonraki çerçevelerle eşleştirebilmektedir. Varsayalım ki 120 ve 125 arasındaki çerçeveler 10 ve 15 arasındaki çerçeveler ile yer değiştirilerek sahte video elde edilmiş olsun. FR listesi Şekil 2.24.(b)'de görüldüğü gibi yanlış kararları da içermektedir, listedeki üçüncü ve dördüncü girişler. Hatalı girişlerdeki uzaysal uzaklık değerinin 110 olması gerekmektedir. Önerilen yöntemin ilgili hatalı kararları da ortadan kaldırması gerekmektedir. Çalışmada önerilmiş olan Hatalı eşleştirme eliminasyon algoritması aşağıda verilmektedir.

Adım 1. Uzaysal uzaklık vektörü birinci ve ikinci kolonlar arası mutlak fark alınarak hesaplanır ve dördüncü kolon olarak FR listesine eklenir.

Adım 2. Uzaysal uzaklıkların gösterildiği kolonun histogramı elde edilir.

Adım 3. Histogramdaki tepe değerleri bulunur $P = \{p_1 \dots p_n\}$.

Adım 4. FR listesindeki uzaklık değerleri (2.55) ile verilen şartı sağlıyorsa, verilen kural kullanılarak güncellenecektir.

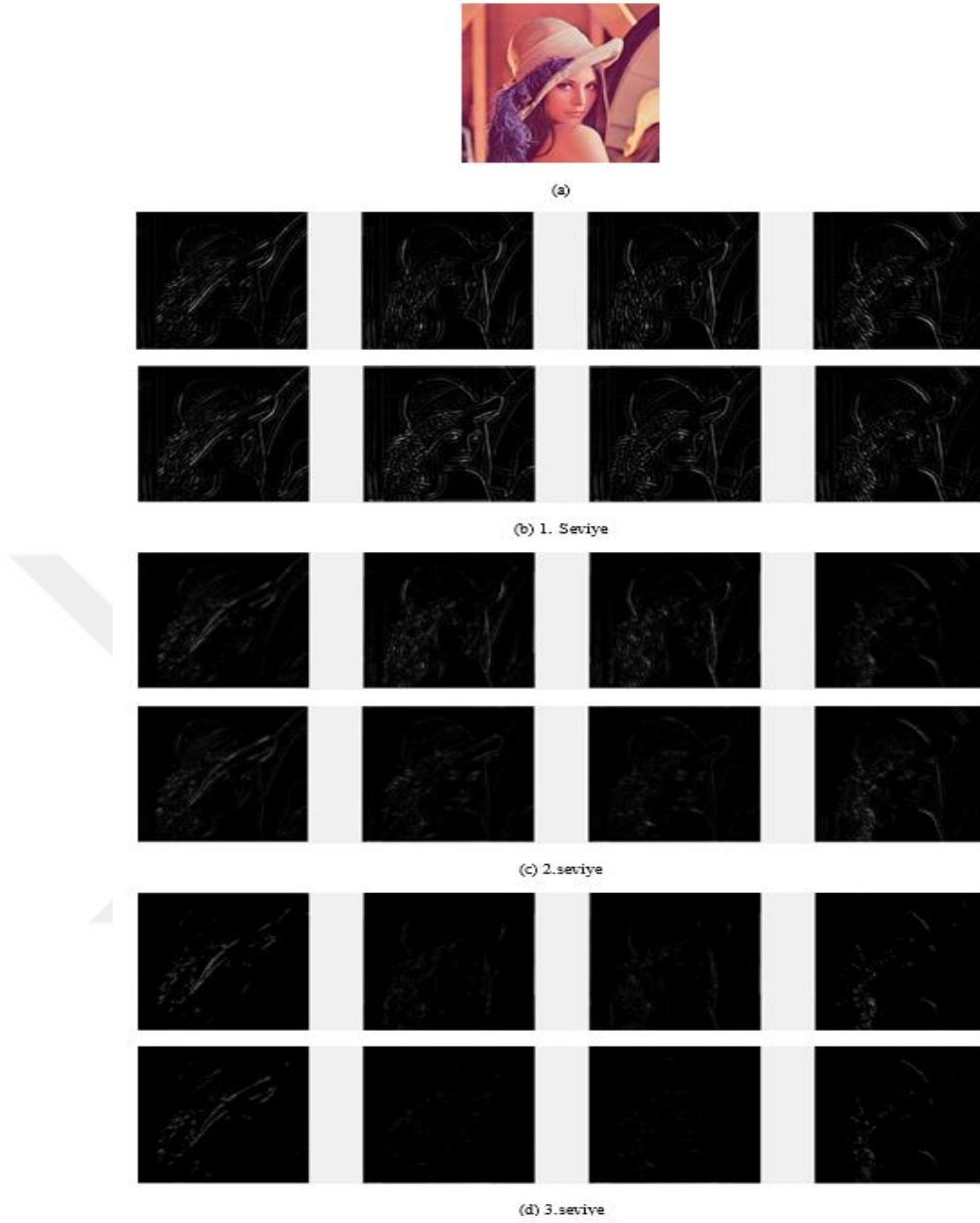
$$\forall i: fr_4^i \in [p_i - d, p_i + d], i = 1 \dots n \Rightarrow fr_4^i = p_i \quad (2.55)$$

Şekil 2.24.(b)'deki örnek için histogramdan elde edilecek olan tepe değeri 110'dur. Dördüncü kolondaki uzaysal uzaklık değerleri Şekil 2.24.(c)'de görüldüğü gibi

değiştirilecektir. Değiştirilmiş olan FR listesi videoda yer alan sahteciliklerin temsil edildiği nihai liste olacaktır. Anlatılan Hatalı eşleştirme eliminasyon ve PSNR eliminasyon yöntemleri, yöntemin tanılama performansını iyileştirebilmek amacıyla önerilmiştir.

2.2.4. Videolarda Bölge Tabanlı Sahtecilik Tespiti

Videoda yapılan bölge tabanlı sahteciliklerde çerçeve üzerine yapıştırılacak olan bölge (aynı çerçeveden veya başka bir video çerçevesinden alınan) yapıştırıldıktan sonra bölge sınırlarına sahtecilik izlerini kapatmak için bulanıklaştırma işlemi uygulanmaktadır. 2013 yılında Liu vd. Alt Örneklenmeyen Contourlet dönüşümü (Non-Subsampled Contourlet dönüşümünü-NSCT) katsayılarını kullanarak görüntü kenarlarını sınıflamıştır [174]. Çalışma görüntüdeki kenarlardan özellikler üretirken de yine aynı dönüşümden faydalanmış ve sonuçlarının tatmin edici olduğunu deneysel sonuçları ile beraber desteklemiştir.



Şekil 2.25. Seviyelere göre sekiz yönde elde edilen görüntüler

Tez kapsamında literatürdeki yöntemler göz önüne alındığında, bölgesel seviyede sahteciliğin yapıldığı yerlerdeki kenar bilgisindeki bulanıklaştırmayı tespit edebilmek amacıyla [174]'te görüntü için önerilen yöntemin, video alanındaki probleme özgü olarak modifiye edilip kullanılmasına karar verilmiştir. Bu amaçla önerilen çalışmada videodaki bulanık kenarların tespiti ile bölgesel sahtecilik izleri belirlenmektedir.

Orijinal görüntüler üç farklı kenar tipi barındırabilmektedir: Adım, Çatı ve Konveks. Görüntü üzerinde gerçekleştirilen bir bulanıklaştırma işlemi, adım kenar bilgisini azaltırken konveks kenar bilgisini artıracaktır (kenarlardaki geçişlerin yumuşaması sebebi

ile). Kenarlardan bilgi çıkarabilmek için kullanılacak olan dönüşüm fonksiyonu olan NSCT kendi içerisinde iki ayrı filtre bankasını kullanmaktadır: Alt örneklenmeyen piramid filtre bankası (non-sampled pyramid filter bank-NSPFB) ve yönlü filtre bankası (non-sampled directional filter bank-NSDFB). Bahsi geçen ilk filtre bankası çok ölçekli ayrıştırma yaparken, diğeri yönlü ayrıştırma gerçekleştirmektedir. NSPFB sonucu elde edilen alçak frekans görüntü üzerinde iteratif olarak dönüşüm yöntemi uygulanabilmektedir. Yönlü filtre bankası, piramid filtre bankasının çıkışı olan yüksek frekans görüntüsünü çeşitli yönlerde alt bantlara ayırmaktadır. Şekil 2.25.'te örnek bir test görüntüsü için üç seviyede sekiz yönlü NSCT çıktı görüntüleri verilmiştir. NSCT dönüşümü kaymadan bağımsız, çok ölçekli ve çok yönlü bir dönüşüm olup yüksek frekans bileşenleri hakkında daha fazla detayı orijinal görüntü ile aynı büyüklükte olacak şekilde farklı yönlerde verebilmektedir. NSCT dönüşümüne ilişkin detaylar aşağıda verilmiştir. NSCT dönüşümünün ardından, görüntüdeki kenarlara ait özellik çıkarımı ve çıkarılan özelliklerin etiketlenmesinde kullanılan yöntemler anlatılacaktır.

2.2.4.1. NSCT Dönüşümü

Contourlet dönüşümü (Counturlet-CT), Do ve Vetterli tarafından önerilen bir ayrıştırma yöntemidir [175]. CT, ayrık Dalgacık dönüşümüne göre görüntüdeki farklı yönlerdeki yumuşak düzgün kontür (dış hatları) temsil etmede daha etkindir. Yönlü ve çok ölçekli bir dönüşüm olan Contourlet dönüşümü, Laplacian piramidi (LP) ve yönlü filtre bankalarının (Directional Filter Bank-DFB) birleşiminden oluşmaktadır. Fakat gerek LP gerekse DFB esnasındaki alt örnekleme ve üst örneklemlerden kaynaklı olarak ötelemeden bağımsız değildir. Cunha vd. tarafından 2005'te ötelemeden bağımsızlığı gerçekleştirebilmek amacı ile NSCT yöntemi önerilmiştir [176]. NSCT dönüşümü iki kısım halinde düşünülebilir: (1) Alt örneklenmeyen piramid yapısı (2) Alt örneklenmeyen DFB yapısı. Her iki kısma ilişkin detaylar aşağıda verilecektir.

NSCT için Piramid filtre bankası: Alt örneklenmeyen piramid filtre bankasının temel kısmı iki kanallı alt örneklenmeyen filtre bankasıdır. Alt örneklenmeyen filtre bankası ne alt örnekleme ne de üst örnekleme sahiptir ve bu nedenle de ötelemeden bağımsızlık özelliği kazanmıştır. (2.56) ile verilen ifadede mükemmel yeniden yapılandırma durumu gösterilmiştir.

$$H_0(Z)G_0(Z) + H_1(Z)G_1(Z) = 1 \quad (2.56)$$

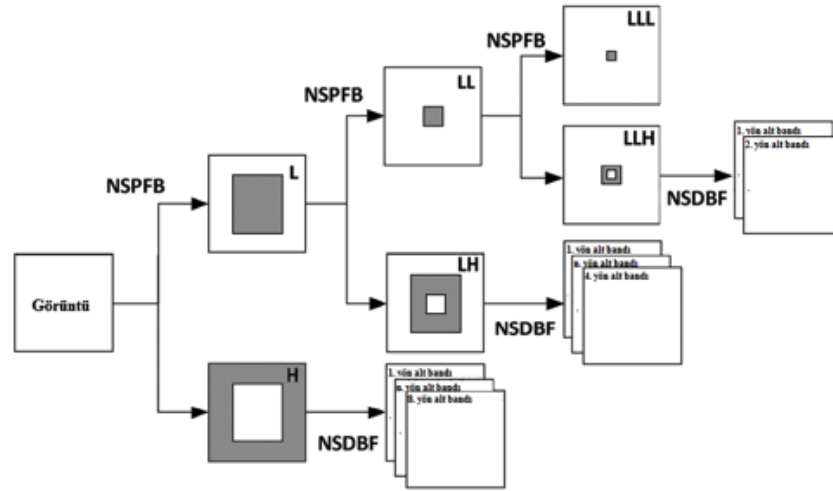
Verilen ifadede $H_0(Z)$, alçak geçiren ayrıştırıcı filtreyi temsil ederken, $H_1(Z)$ ise yüksek geçiren ayrıştırıcı filtreyi temsil etmektedir. $G_0(Z)$ ve $G_1(Z)$ ise sırası ile, alçak geçiren ve yüksek geçiren yeniden yapılandırma filtrelerini temsil etmektedir. Çok ölçekli ayrıştırmayı sağlayabilmek amacı ile iterasyonlu alt örneklenmeyen filtre bankaları ile alt örneklenmeyen piramitler oluşturulacaktır. Gelecek her seviye için, bütün filtreler her iki yönde iki ile üst örneklenir. Böylece, mükemmel yeniden yapılandırma da garantilenmiş olacaktır. 1 seviyesi için alt örneklenmeyen piramide karşılık gelen filtreler (2.57) ile ifade edilmiştir.

$$H_d^{eq} = \begin{cases} H_1(z^{2^{d-1}}) \prod_{j=0}^{d-2} H_0 z^{2^j}, & 1 \leq d \leq 2^l \\ \prod_{j=0}^{d-1} H_0 z^{2^j}, & d = 2^l \end{cases} \quad (2.57)$$

NSCT için yönlü filtre bankaları: Alt örneklenmeyen yönlü filtre bankalarının da temel taşı yine iki kanallı alt örneklenmeyen filtre bankalarıdır. Alt örneklenmeyen yönlü filtre bankaları itere edilerek daha keskin yönlü ayrıştırmaların gerçekleştirilmesi sağlanır. Her gelecek seviye için, bütün filtreler verilen Q (quincunx) matrisi (2.58) ile beraber üst örneklenir.

$$Q = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (2.58)$$

NSCT dönüşümü, alt örneklenmeyen piramit filtre bankaları (NSPFB) ile alt örneklenmeyen yönlü filtre bankalarının (NSDFB) birleşiminden oluşmaktadır. Şekil 2.26.'da çok ölçekli ve çok yönlü NSCT dönüşümüne bir örnek verilmiştir. Şekilden de gözlemlenebileceği gibi, dönüşüm NSPFB'nin alçak geçiren alt bandı üzerinde iteratif olarak devam ettirilebilmektedir. NSPFB öncelikle görüntüyü alçak ve yüksek geçiren alt bantlara ayırmaktadır. Ardından, NSDFB, yüksek geçiren alt bandı çok yönlü alt bantlara ayrıştırmaktadır. Anlatılan yöntem iteratif olarak devam ettirilebilmektedir. NSCT dönüşümü, öteleme bağımsız, çok ölçekli ve çok yönlü hızlı bir dönüşüm yöntemidir.



Şekil 2.26. NSCT dönüşümüne bir örnek

Video görüntüsünün alt sahne içermediği düşünülürse video çerçevesine aynı çerçeveden ya da başka bir videodan alınıp eklenecek olan bölgenin, videonun bütünlüğünü koruma açısından bütün çerçevelere eklenmesi gerekmektedir. Bu düşünceden hareketle test videosu çerçevelere ayrıştırıldıktan sonra ilk çerçeve referans çerçeve olarak alınmakta ve bu çerçevede bulanık kenar bilgisi araştırılmaktadır. Bulanık kenar bilgisinin varlığı, o çerçeve üzerinde bölge tabanlı sahteciliğin gerçekleştirilmiş olabileceğine dair bilgi taşımaktadır. Referans çerçevede sahteciliğin yapılmış olması da videoda bölge tabanlı sahtecilik yapıldığını göstermektedir. Hedef doğrultusunda izlenmesi gereken iki adım mevcuttur: İlk çerçevedeki kenarların sınıflandırılması, kenarlardan özelliklerin elde edilmesi ve özelliklere göre kenarların etiketlenmesi. Her bir adıma ilişkin detaylar aşağıdaki şekildedir.

Referans çerçevedeki kenarların sınıflandırılması: Çerçevedeki kenarların sınıflandırılmasında NSCT'den faydalanılacaktır. Kenarların etiketlenmesi aşağıda adımlar halinde verilmektedir.

Adım 1. Referans çerçeve üzerinde Canny kenar algılama algoritması uygulanır ve kenarların bulunduğu koordinatlar E kümesine yerleştirilir. Anahtar çerçeve üzerinde üç seviyeli ve her seviyesinde 8 yön ile NSCT uygulanır.

Adım 2. Çerçevenin tüm kenarları için aşağıdaki adımları uygulanır.

Adım 3. Frekans dönüşüm görüntülerinde, ilgili kenar maksimum değeri üçüncü seviyedeki sekiz görüntünün birinden aldı ise kenar “Adım” olarak sınıflanır ve S kümesine yerleştirilir ve tüm kenarlar işlem görmedi ise Adım 2’ye dönlür.

Adım 4. Frekans dönüşüm görüntülerinde, ilgili kenar maksimum değeri ikinci seviyedeki sekiz görüntünün birinden aldı ise kenar “Çatı” olarak sınıflanır ve M kümesine yerleştirilir ve tüm kenarlar işlem görmedi ise Adım 2’ye dönülür.

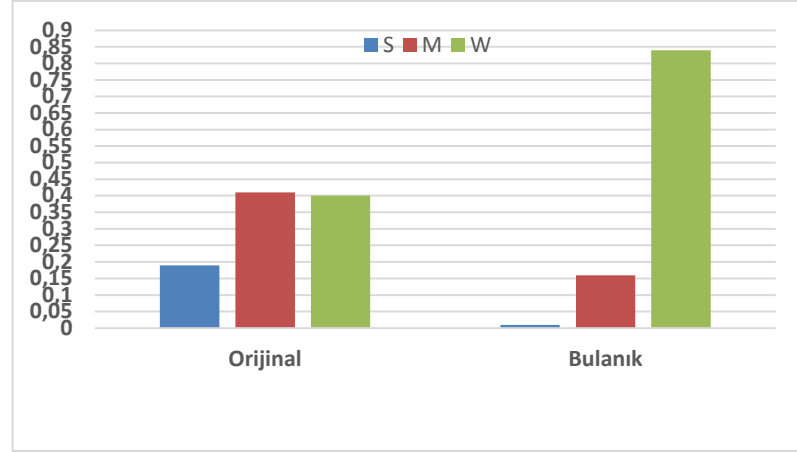
Adım 5. O anki kenar Konveks olarak sınıflanır ve W’ye yerleştirir ve tüm kenarlar işlem görmedi ise Adım 2’ye dönülür.

Adım 6. O anki çerçevedeki kenarlar, sınıfları ile beraber çıkışa aktarılır: S, M, W.

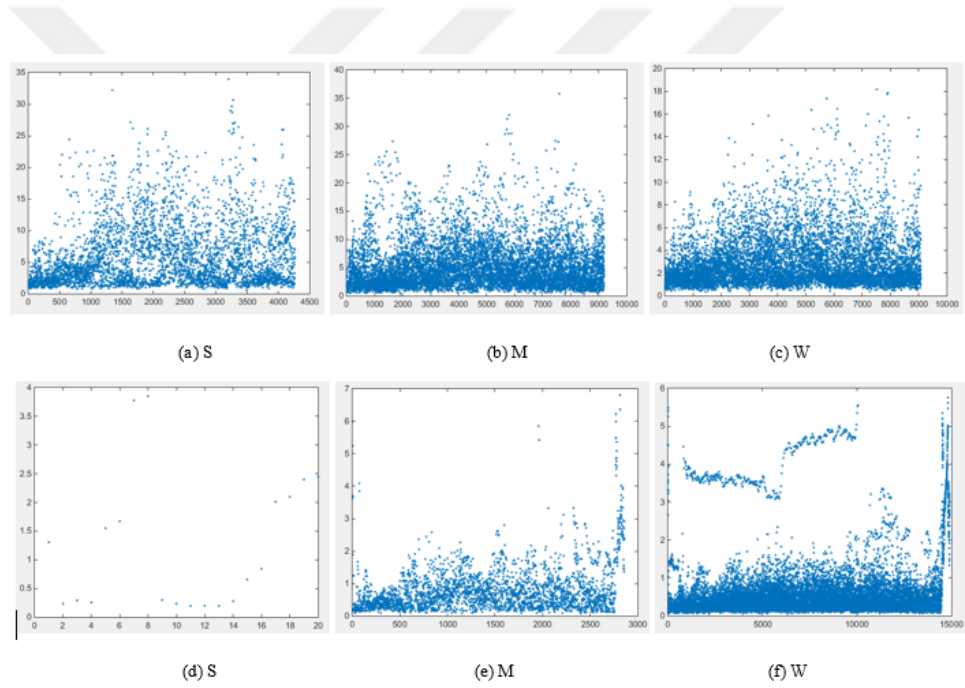
Şekil 2.27.’de Şekil 2.25.(a)’da verilen test görüntüsü için ve aynı görüntünün 5x5 çerçeve $\sigma = 2$ olan bir Gauss bulanıklaştırma uygulanması ile elde edilen adım, konveks kenar sayılarının değişimi gösterilmektedir. Şekilden de gözlemlenebileceği gibi, konveks kenarlar bulanıklaşmanın ardından artış gösterirken adım kenarların sayısında azalış mevcuttur. Kenar türlerinin belirlenmesinin ardından, bir sonraki aşama olarak kenarlardan özellik çıkarımı yapılacaktır.

Sınıflanan Kenarlardan Özellik Çıkarımı ve Etiketlendirme: Bu amaçla kullanılan özellik, 3 seviyeli ve her seviyesinde 8 dönüşüm olan NSCT’nin üçüncü seviyede ilgili kenar için üretmiş olduğu maksimum frekans değeridir. Burada maksimum frekans tepkisinin belirlenmesinde mutlak katsayılar kullanılmıştır. Üçüncü seviyedeki maksimum katsayının kullanılmasının sebebi, orijinal görüntüdeki kenarlardan elde edilecek yüksek frekans tepkisinin, bulanık bir görüntüye nazaran daha yüksek olacak olmasıdır. Şekil 2.28’de aynı görüntüdeki ve 5x5 çerçeve $\sigma = 2$ ile Gauss bulanıklaştırması uygulanmış halindeki kenarlardan elde edilen üçüncü seviye mutlak maksimum frekans katsayısı dağılımları, farklı kenar tipleri için ayrı ayrı gösterilmektedir. Şekilden de gözlemlenebileceği gibi, bulanık olmayan bir görüntüde, kenarlardan elde edilen yüksek frekans katsayıları çok daha düşük olmaktadır. Fakat yine de kenar tiplerine göre katsayı dağılımları değişiklik gösterdiğinden dolayı, kenarların tipine bağlı olarak eşiklendirme yapılmış ve bulanık kenarlar tespit edilmiştir.

Çerçevedeki üç farklı kenar tipine ayrılan kenar noktalarının her biri için bir elemanlı özellik vektörü çıkarılmıştır. Her bir kenar noktasının özelliğinin belirli bir eşik değeri ile kıyaslanması sonucunda bu kenarın bulanıklığı hakkında karar verilmiştir. Bulanık kenarlar görüntü ile aynı boyuttaki bir B matrisinde işaretlenmiştir.



Şekil 2.27. Bulanıklaştırmadan önce ve sonraki kenar tipleri yoğunlukları



Şekil 2.28. (a)(b)(c) Orijinal (d)(e)(f) Bulanık kenar tiplerine göre mutlak maksimum frekans katsayısı dağılımları

İşaretlenmiş Bulanık Kenarların Kullanımı ile Sahteciliğin Tespiti: Bulanık kenarları temsil eden B görüntüsü üzerinde yanlış tespitleri ortadan kaldırmak için bu matris üzerinde son işlemler uygulanacaktır. Bu amaçla B üzerindeki her bir piksel merkez kabul edilerek 3x3'lük komşuluğundaki kenar sayısının 3'ü geçmesi durumunda ilgili piksel, bulanık olarak etiketlendirilecek aksi durumda işareti kaldırılacaktır. Bahsedilen işlemin B için uygulanış şekli ifade (2.59)'da verilmiştir.

$$B'(i, j) = \sum_{p=-1}^1 \sum_{q=-1}^1 B(i + p, j + q) \quad (2.59)$$

Son işlem sonrası üretilen B' ikili görüntülerde beyaz piksel bulunması, karşılık düşen çerçevenin bulanık kenar içerdiğini göstermektedir. Problemliliğin kullanıcı tarafından daha net olarak gözlemlenebilmesi amacı ile çalışmada her beyaz piksel ile ona en yakın olan beyaz piksel arasında doğru çizdirilmiştir. Böylelikle sahte olan bölgenin doğrular ile işaretlenebilmesi mümkün kılınmıştır.

2.2.5. Pasif Video Doğrulama Sistemi Yazılımı

Video sahteciliği tespiti için yapılan çalışmalar Matlab simülasyon ortamı kullanılarak denenmiş ve önerilen teknikler hazır paketler olmaksızın kodlanarak gerçekleştirilmiştir. Geliştirilen yöntemler tek bir çatı altında toplanarak, Pasif Video Doğrulama Sistemi isimli bir yazılım da tez kapsamında geliştirilmiştir. Geliştirilen yazılım aynı zamanda dll biçiminde dağıtılarak herhangi bir programlama dili tarafından kullanılabilir hale getirilmiştir.

3. BULGULAR VE İRDELEME

Bu bölümde tez kapsamında gerçekleştirilen çalışmalara ait bulguların, literatürde var olan çalışmalara ait sonuçlarla kıyaslanması gerçekleştirilerek, önerilen yöntemlerin üstünlükleri ve getirdikleri dezavantajlar irdelenecektir. Yapılan çalışmalar kısmında üzerinde durulduğu gibi önerilen yöntemler, görüntü ve video sahteciliklerinin tespitindeki belirli problemlere çözüm getirmeyi hedeflemektedir. Literatürdeki çeşitli problemlere çözüm getirmek için önerilen tekniklerden elde edilen bulgular ve kendi alanlarındaki diğer çalışmalarla kıyaslanması sırasıyla ilerleyen bölümlerde verilmektedir. Öncesinde görüntü ve video için kullanılan veritabanları ve yapılan çalışmaların performansını değerlendirmek için kullanılan ölçütler verilecektir.

3.1. Veritabanı

Görüntü sahteciliklerinin tespiti için yapılan çalışmalarda GIMP 2.8 kullanılarak üç ayrı veritabanı hazırlanmıştır. Bunlardan CoMoFoD veritabanı [177] sahte görüntüler içeren hazır bir veri tabanı olmasına rağmen , veri tabanında kopyalanan bölge büyüklüğüne göre (32x32, 48x48, 64x64, 96x96) kopyala yapıştır yapılmadığı için, veri tabanının orjinal görüntülerinden faydalanılarak kopyalanan bölge büyüklüğü 32x32, 48x48, 64x64, 96x96 olan sahte görüntüler oluşturulmuştur. Oluşturulan sahte görüntüler 512x512 büyüklüğündedir. İkinci bir veri tabanı olarak 768x512 büyüklüğünde PNG formatında 24 görüntü içeren araştırma amaçlı kullanılan Kodak veri tabanından [178] faydalanılmıştır. Görüntüler orjinal görüntüler olup sahte görüntüler tarafımızdan oluşturulmuştur. Üçüncü veri tabanı olarak ise 1024x768 büyüklüğünde Google görsellerden [179] görüntü alınarak sahte görüntüler oluşturulmuştur. Yöntemin ataklara karşı dayanıklılığını ölçmek için de oluşturulan sahte görüntülere JPEG sıkıştırma, Gauss bulanıklaştırma ve Beyaz Gauss gürültüsü işlemleri uygulanmıştır.

Medikal görüntüdeki sahteciliklerin tespiti için yapılan çalışmada hazırlanan veri tabanına çeşitli ataklar (Dönme, Ölçekleme, Bulanıklaştırma ve Beyaz Gauss gürültüsü) uygulanmış ve toplamda 420 oynanmış gri seviye medikal görüntü oluşturulmuştur. Sahtecilik yapılan orijinal medikal görüntüler internet ve [180]'den alınmıştır. Senaryo olarak;

- Medikal görüntüden tek bir ya da birden çok bölge kopyalanarak başka bir bölge ya da bölgelere yapıştırılmıştır. Bu şekilde 58 tane oynanmış medikal görüntü oluşturulmuştur.

- Medikal görüntünün bir bölgesi kopyalanmış ve bu bölge [20, 90, 180, 220] açıları ile döndürüldükten sonra bir başka bölgeyi tekrarlama ya da gizleme amacıyla o bölge üzerine yapıştırılmıştır. Bu şekilde 48 adet döndürme atağı uygulanmış oynanmış görüntü oluşturulmuştur.

- Kopyalanan bölge yapıştırılmadan önce 80, 90, 110 ve 120 ölçekleme faktörleriyle yeniden boyutlandırılmıştır. Bu şekilde de 59 adet oynanmış görüntü oluşturulmuştur.

- 58 oynanmış görüntüye $w = 5$ ve $\sigma [1.5-3]$ aralığında Gauss bulanıklaştırma uygulanmıştır.

- Tepe Gürültü Oranları (Signal Noise Ratio-SNR) değerleri 30 ve 60 dB olacak şekilde 58 oynanmış tıbbi görüntüye Beyaz Gauss gürültüsü eklenmiştir.

Hazırlanan tüm sahte görüntüler açık kaynaklı GIMP görüntü editörü aracılığıyla oluşturulmuştur.

Video sahteciliklerinin tespiti için yapılan çalışmalarda sahte video oluşturmak için 25'i SULFA veri tabanından alınan 31 video kullanılmıştır. SULFA veritabanı orijinal videolardan oluşmaktadır. SULFA'dan seçilen rasgele videolar: can_220_book, can_220_flap(1), can_220_flap(2), can_220_garden(1), can_220_garden(4), can_220_hallway(2), can_220_man(2), can_220_road(1), can_220_road(5), can_220_room(3), can_220_street(3), can_220_street, fuji_2800_busstop(4), fuji_2800_man(2), fuji_2800_outdoor(4), fuji_2800_road(2), fuji_2800_road(5), fuji_2800_stair_outdoor, fuji_2800_street(1), nik_s3000_ball, nik_s3000_bridge(1), nik_s3000_bridge, nik_s3000_door(2) ve nik_s3000_indoor_stairs 320x240 çözünürlükte). Sahte videolar açık kaynaklı Virtual Dub video editörü aracılığıyla oluşturulmuştur. SULFA haricinde oluşturulan 6 sahte video da farklı içerikteki sinema filmlerinden alınarak oluşturulmuştur. Oluşturulan sahte video veri tabanındaki 16 video hareketli kameradan elde edilirken 15'i sabit kameradan alınmıştır. Test videolarının özellikleri Tablo 3.1'de verilmektedir.

Video için yapılan çalışmalarda yöntemin Çerçeve Yansıma sahteciliğini de tespit edebildiğini gösterebilmek için bazı çerçeve dizilerinin ayna simetrik yansıması alınarak videoya eklenmesiyle sahte videolar oluşturulmuştur. Bunun için oluşturulan video veri tabanına ek olarak orijinal videolardan alınarak 10 çerçeve tekrarlama-FD, 10 tane çerçeve

yansıma-FM ve 10 tane de hem çerçeve tekrarlama hem de çerçeve yansıma-FD+FM sahteciliği yapılmış 30 sahte video oluşturulmuştur. Tablo 3.2’de videoların özellikleri ve videolar üzerinde yapılan sahteciliklerin detayları verilmektedir.

Tablo 3.1. Test videolarının özellikleri

Test video	Orijinal çerçeve sayısı	Sahtecilik sonrası çerçeve sayısı	Çerçeve oranı	Sahtecilik işlemi
Video 1	357	407	29.97fps	170-220 arası 10’dan sonraya
Video 2	369	429	29.97fps	100-160 arası 315’den sonraya
Video 3	332	372	29.97fps	60-100 arası 280’den sonraya
Video 4	353	403	29.97fps	200-250 arası 10’dan sonraya
Video 5	321	361	29.97fps	100-140 arası 180’den sonraya
Video 6	122	142	29.97fps	40-60 arası 80’den sonraya
Video 7	340	390	29.97fps	110-160 arası 10’dan sonraya
Video 8	301	371	29.97fps	150-220 arası 240’dan sonraya
Video 9	296	356	29.97fps	160-220 arası 130’dan sonraya
Video 10	396	436	29.97fps	220-260 arası 130’dan sonraya
Video 11	286	326	29.97fps	30-70 arası 210’dan sonraya
Video 12	361	421	29.97fps	85-145 arası 220’den sonraya
Video 13	300	355	30fps	5-60 arası 190’dan sonraya
Video 14	330	380	30fps	110-160 arası 20’den sonraya
Video 15	270	310	30fps	30-70 arası 130’dan sonraya
Video 16	330	380	30fps	5-55 arası 187’den sonraya
Video 17	420	460	30fps	223-263 arası 60’dan sonraya
Video 18	330	370	30fps	40-80 arası 215’den sonraya
Video 19	330	390	30fps	110-170 arası 200’den sonraya
Video 20	326	363	30fps	138-175 arası 5’den sonraya
Video 21	344	394	30fps	120-170 arası 220’den sonraya
Video 22	293	323	30fps	250-280 arası 10’dan sonraya
Video 23	314	374	30fps	130-190 arası 258’den sonraya
Video 24	188	228	30fps	10-50 arası 105’den sonraya
Video 25	330	360	30fps	179-209 arası 220’den sonraya
Video 26	387	584	25fps	187-384 arası 386’dan sonraya
Video 27	174	222	31.27fps	54-102 arası 173’den sonraya
Video 28	461	540	25fps	51-130 arası 133’den sonraya
Video 29	213	266	25fps	73-126 arası 212’den sonraya
Video 30	160	190	23.97fps	10-40 arası 70’den sonraya
Video 31	401	481	25fps	189-269 arası 270’den sonraya

Tablo 3.2. Test videolarının özellikleri

Test video	Orijinal çerçeve sayısı	Sahtecilik sonrası çerçeve sayısı	Çerçeve oranı	Atak	Sahtecilik işlemi
Video 1	369	429	29.97fps	FD	100-160 arası 315’ten sonraya

Video 2 332 372 29.97fps FD 60-100 arası 280'den sonraya
Tablo 3.2. 'nin devamı

Video 3	353	403	29.97fps	FD	200-250 arası 10'dan sonraya
Video 4	122	142	29.97fps	FD	40-60 arası 80'den sonraya
Video 5	301	371	29.97fps	FD	150-220 arası 240'dan sonraya
Video 6	270	310	30 fps	FD	30-70 arası 130'dan sonraya
Video 7	330	390	30 fps	FD	110-170 arası 200'den sonraya
Video 8	188	228	30 fps	FD	10-50 arası 105'den sonraya
Video 9	344	394	30 fps	FD	120-170 arası 220'den sonraya
Video 10	293	323	30 fps	FD	250-280 arası 10'dan sonraya
Video 11	369	398	29.97fps	FM	60-89 arası 1'den sonraya
Video 12	332	371	29.97fps	FM	60-99 arası 280'den sonraya
Video 13	353	403	29.97fps	FM	5-55 arası 75'den sonraya
Video 14	122	142	29.97fps	FM	40-60 arası 80'den sonraya
Video 15	301	361	29.97fps	FM	5-65 arası 105'den sonraya
Video 16	270	320	30 fps	FM	100-150 arası 200'den sonraya
Video 17	330	400	30 fps	FM	2-72 arası 102'den sonraya
Video 18	188	218	30 fps	FM	20-50 arası 70'den sonraya
Video 19	293	353	30 fps	FM	90-150 arası 180'den sonraya
Video 20	344	403	30 fps	FM	1-60 arası 81'den sonraya
Video 21	369	438	29.97fps	FD+FM	60-89 arası 1'den ve 200-240 arası 280'den sonraya
Video 22	332	391	29.97fps	FD+FM	60-99 arası 280'den ve 10-30 arası 325'den sonraya
Video 23	353	433	29.97fps	FD+FM	5-55 arası 75'den ve 160-190 arası 230'dan sonraya
Video 24	122	162	29.97fps	FD+FM	40-60 arası 80'den ve 10-30 arası 100'den sonraya
Video 25	301	386	29.97fps	FD+FM	5-65 arası 105'den ve 225-250 arası 270'den sonraya
Video 26	270	350	30 fps	FD+FM	100-150 arası 200'den ve 10-40 arası 60'dan sonraya
Video 27	330	430	30 fps	FD+FM	2-72 arası 102'den ve 235-265 arası 290'dan sonraya
Video 28	188	238	30 fps	FD+FM	20-50 arası 70'den ve 125-145 arası 165'ten sonraya
Video 29	293	378	30 fps	FD+FM	90-150 arası 180'den ve 230-255 arası 270'den sonraya
Video 30	344	438	30 fps	FD+FM	1-60 arası 81'den ve 180-215 arası 250'den sonraya

Video için yapılan çalışmalarda yöntemin bölgesel tabanlı sahtecilik tespit performansını test edebilmek için ise videolarda bölge tabanlı oynama yapılarak sahte videolar oluşturulmuştur. Bunun için 20 tane orjinal test videosu üzerinde Kopyala yapıştır

(KY) veya Video Birleştirme (VB) sahtecilikleri uygulanarak sahte videolar oluşturulmuştur. Tablo 3.3'te bu videoların özellikleri verilmektedir.

Tablo 3.3. Test videoları özellikleri

Test video	Çerçeve Sayısı	İşlem
Video 1	118	KY
Video 2	353	VB
Video 3	353	VB
Video 4	330	KY
Video 5	314	KY
Video 6	321	KY
Video 7	130	VB
Video 8	120	VB
Video 9	120	VB
Video 10	353	VB
Video 11	270	VB
Video 12	314	VB
Video 13	314	VB
Video 14	314	KY
Video 15	270	KY
Video 16	270	KY
Video 17	353	KY
Video 18	353	KY
Video 19	321	KY
Video 20	314	KY

3.2. Kullanılan Ölçütler

Görüntü için geliştirilen blok tabanlı sahtecilik tespit yöntemlerinin başarısını değerlendirebilmek için p doğruluk oranı ölçütü tanımlanmıştır. Doğruluk oranı yöntemin tespit gücü hakkında bilgi vermektedir. Bu değer [0-1] aralığında değişmektedir. Eğer yöntem kopyalanıp yapıştırılmış bölgelerin tamamını bulursa p değeri 1 olacak ancak yöntem sahte bölge yerine sahte olmayan bölgeyi sahte olarak işaretlerse doğruluk oranı azalacaktır. D_1 ve D_2 sahte görüntüde kopyala yapıştır yapılmış bölgeler olsun. R_1 ve

R_2 'de algoritma tarafından kopyala yapıştır yapılmış bölgeler olarak tespit edilmiş bölgeler olsun. Algoritmanın doğruluk oranı (3.1) kullanılarak hesaplanmaktadır.

$$p = \frac{|D_1 \cap R_1| + |D_2 \cap R_2|}{|D_1| + |D_2|}, \quad f = \frac{|D_1 \cup R_1| + |D_2 \cup R_2|}{|D_1| + |D_2|} - p \quad (3.1)$$

Doğruluk oranı p 'nin yanında görüntüde sahte olmadığı halde sahte olarak algılanan bölgelerin miktarını belirlemek için yanlış negatif f değeri tanımlanmıştır. f değerinin sıfıra gitmesi algoritmanın doğruluk oranının iyileştiği anlamına gelmektedir.

Anahtar nokta tabanlı önerilen yöntemlerin sahte bölge tespit performansını değerlendirmede ise (3.2) bağıntısında verilen Tespit oranı (Detection Rate-DR) metriği kullanılmıştır. Metriğin ilk kısmını oynanmış bölge içerisindeki eşleşmiş anahtar nokta sayısının K_F , oynanmış bölgeler içerisindeki toplam piksel sayısı F 'ye oranı oluşturmaktadır. İkinci kısmını ise oynanmış bölgeler dışındaki eşleşmiş anahtar nokta sayısının K_B , sahte bölgeler dışında kalan toplam piksel sayısı B 'ye oranı oluşturmaktadır. Görüntü boyutundan bağımsızlığı sağlamak için $N \times M$ boyutundaki bir test görüntüsü için bu metrik $NM/100$ ile çarpılmıştır. DR değeri ne kadar yüksek elde edilirse, yöntem oynanmış bölge tespitini o kadar yüksek doğrulukta yapabiliyor demektir.

$$DR = \left(\frac{K_F}{|F|} - \frac{K_B}{|B|} \right) \frac{NM}{100} \quad (3.2)$$

Video için yapılan çalışmaların performans değerlendirmesinde ise Duyarlılık (Precision Rate-PR), Seçicilik (Recall Rate-RR) ve tanılama doğruluğu (Detection Accuracy-DA) ölçütlerinden yararlanılmıştır. PR, RR ve DA'ya ilişkin tanım ifadeleri (3.3)'de sırasıyla verilmiştir. TP, TN, FP ve FN sırasıyla orijinal çerçevenin orijinal, sahte olan çerçevenin sahte, orijinal olan çerçevenin sahte ve sahte olan çerçevenin orijinal olarak etiketlenmesi olarak ifade edilir.

$$PR = \frac{TP}{TP+FP} \quad RR = \frac{TP}{TP+FN} \quad DA = \frac{TP+TN}{TP+TN+FP+FN} \quad (3.3)$$

Eğer elde edilen sonuçlarda yöntemin üretmiş olduğu PR ve RR değerleri daha yüksekse, yöntemin tanılama performansının daha yüksek olduğu söylenecektir. Toplam tanıma sayısı $(TP+TN)$ ile hesaplanırken, $(TP+TN+FP+FN)$ ise videodaki toplam çerçeve

sayısını göstermektedir. Daha yüksek DA değeri yöntemin tanılama oranının daha iyi olduğunu gösterecektir.

3.3. CLD ve Renk Momentlerine Dayalı Kopyala-Yapıştır Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar

Çalışmada AKD tabanlı yöntemlerin [12, 33, 72] son işleme operasyonlarına karşı sağlamlığını artırmak için Renk Momentleri ve Renk Düzeni Tanımlayıcılarına dayanan yeni bir yöntem önerilmektedir. Literatürde ilk kez Kopyala-yapıştır sahteciliği tespitinde Renk Momentleri ve Renk Düzeni tanımlayıcısından yararlanılmıştır. Önerilen yöntemde görüntü öncelikle örtüşen alt bloklara ayrıştırılmaktadır. Sonrasında üç renk kanalı için her bloğun ilk renk momentleri hesaplanmakta ve bloklar, renk momentlerine göre 13 kümeye gruplanmaktadır. Ardından, her kümedeki bloklar için özellik vektörü çıkarımı, işlemi hızlandırmak için eşzamanlı olarak ayrı iş parçacıkları (threads) tarafından yapılmaktadır. Yönteme ilişkin detaylar yapılan çalışmalar kısmında verilmektedir. Önerilen yöntemin etkinliğini gösterebilmek ve AKD tabanlı yöntemlerle kıyaslamasını gerçekleştirebilmek amacıyla yapılan testler ve elde edilen deneysel sonuçlar aşağıda verilmektedir.

Yapılan çalışmada kullanılan değerler $B = 8$, $t_s = 0.5$, $t_{nb} = 100$, $t_d = 40$, $q = 32$, $p = 1/11$, $t_{sc} = 90$ olarak alınmıştır.

Önerilen yöntemin etkinliğini, dayanıklılığını test edebilmek ve literatürdeki benzer çalışmalarla kıyaslabilmek için aşağıdaki deneyler gerçekleştirilecektir.

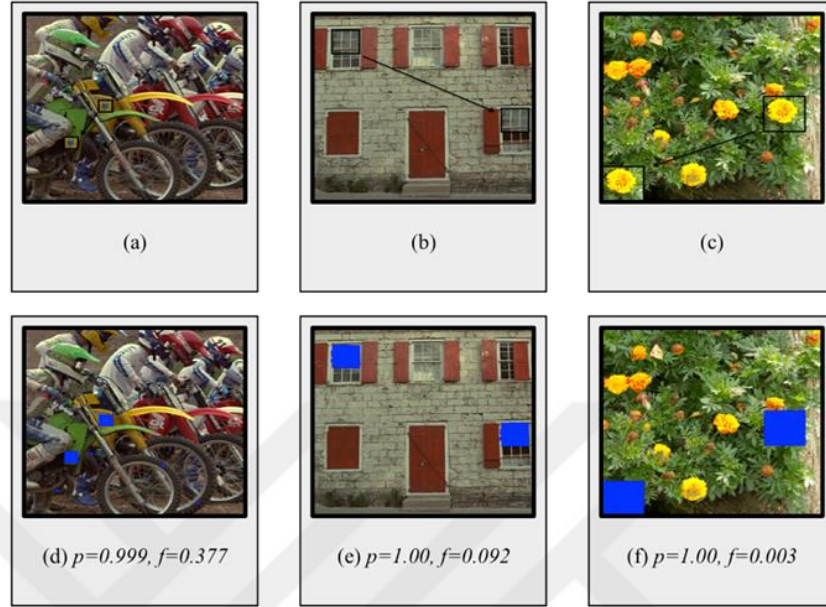
- İlk deneyde sahtecilik sonrasında son işlem uygulanmadığı zaman elde edilen görsel sonuçlar yöntemin doğruluk değerleriyle birlikte verilecektir. Deneylerde kopyalanıp yapıştırılan bölgeler düzenli ve düzensiz olarak seçilmiştir. Aynı zamanda ilk deneyde kopyala-yapıştır işleminin birden çok yapılması durumundaki sonuçlarda değerlendirilecektir.

- İkinci deney olarak sahte görüntülere JPEG sıkıştırma, Gauss bulanıklaştırma ve Beyaz Gauss gürültüsü eklenmesi sonucunda yöntemin etkinliği gösterilecektir.

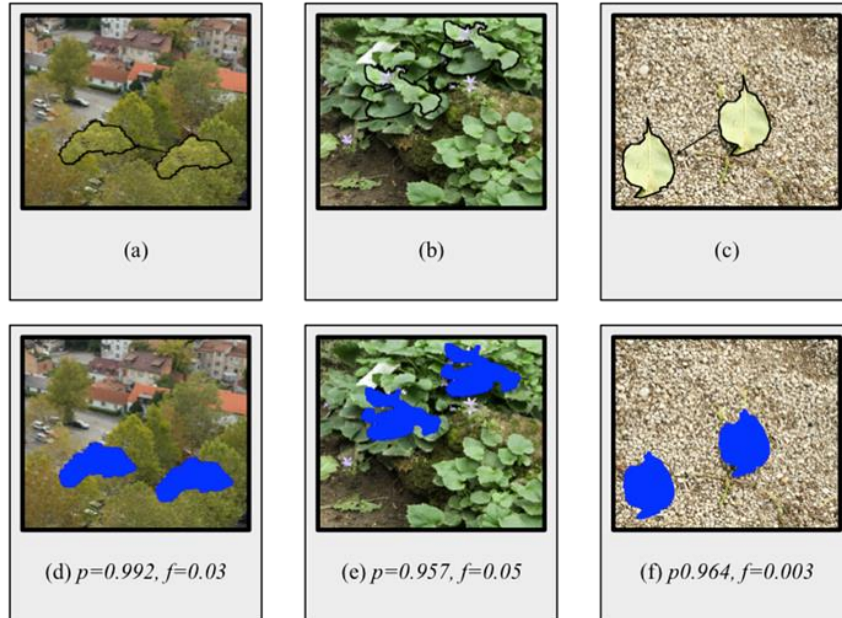
- Son deneyde ise önerilen yöntem literatürdeki benzer çalışmalarla kıyaslanacaktır.

İlk deneyde kopyala-yapıştır yapılan bölgenin düzgün olması, düzgün olmaması ve çoklu kopyala-yapıştır yapılması durumundaki sonuçlar verilecektir. Şekil 3.1.'de kopyalanan bölgenin büyüklüğü 32x32, 64x64 ve 96x96 seçildiğinde yöntemin düzgün kopyalama sonuçları verilmektedir. Bu test görüntüleri için algoritmanın doğruluğu,

şekilde gösterildiği gibi yaklaşık 1'dir. Bu da gösteriyor ki önerilen algoritma kopyalanan ve yapıştırılan bölgelerin neredeyse tamamını tespit etmektedir.



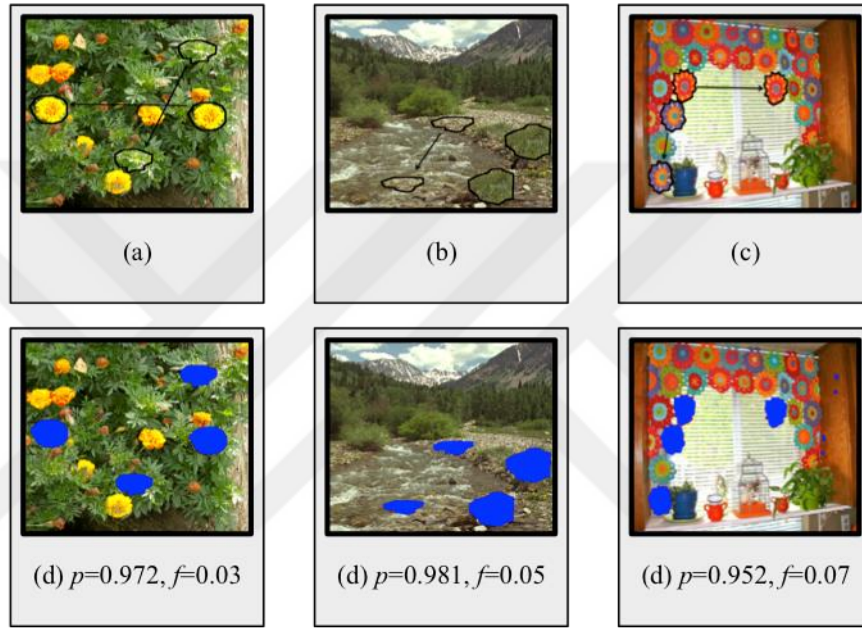
Şekil 3.1.(a) (b) (c) 32x32, 64x64 ve 96x96 boyutlarında kopyalama yapılmış sahte görüntüler (d) (e) (f) Önerilen yöntemin doğruluk oranları ve yanlış negatif değerleri



Şekil 3.2. (a) (b) (c) Düzgün olmayan bölgelerin kopyalanması oluşan sahte görüntüler (d) (e) (f) Önerilen yöntemin doğruluk oranları ve yanlış negatif değerleri

Şekil 3.2.'de düzgün olmayan kopyalama sonucundaki yöntemin görsel çıktılarıyla birlikte doğruluk oranları ve yanlış negatif değerleri verilmektedir. Şekilden de görüldüğü üzere kopyala-yapıştır yapılan bölge düzgün olmasa bile yöntemin doğruluk oranı 0.95'in üzerindedir.

Algoritma, Şekil 3.3.'te görüldüğü gibi çoklu kopyala-yapıştır yapıldığında tekrarlanan bölgeleri yüksek doğruluk oranlarıyla tespit edebilmektedir.

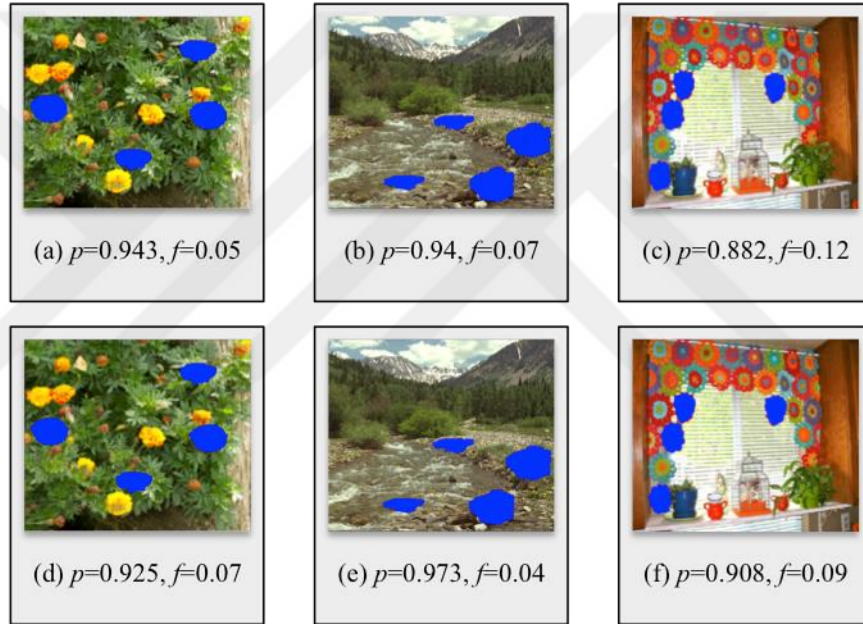


Şekil 3.3. (a) (b) (c) Çoklu kopyalama yapılmış sahte görüntüler (d) (e) (f) Önerilen yöntemin doğruluk oranları ve yanlış negatif değerleri

İkinci deneyde oluşturulan sahte görüntülere son işlem uygulanması durumunda yöntemin etkinliği test edilmektedir. Sahtecilik işlemi sırasında oluşan görsel izlerinin belli olmaması için görüntülere son işlem operasyonları uygulanmaktadır. JPEG sıkıştırma, Gauss bulanıklaştırma ve Beyaz Gauss gürültüsü sahtecilik izlerini kapamak için görüntüye tek başlarına ya da birlikte en çok uygulanan son işlem operasyonlarıdır. Bu operasyonlar yöntemin etkinliğini göstermek için Şekil 3.4.'te çoklu kopyala-yapıştır yapılmış sahte görüntülere uygulanmıştır. Yöntemin Gauss bulanıklaştırma işlemine dayanıklılığını test etmek için görüntüye 5x5 penceresinde standart sapma $\sigma = 1.3$ alınarak bulanıklaştırma işlemi uygulanmıştır. Şekil 3.4.(a) ve 3.4.(d)'de işlemin görsel sonuçları görülmekte ve elde edilen doğruluk oranları sırasıyla 0.94 ve 0.92'dir.

Beyaz Gauss gürültüsü işlemine dayanıklılık testinde ise sahtecilik izlerini kapatmak için görüntüye 25 ve 30 dB gürültü eklenmiştir. Şekil 3.4.(b) ve 3.4.(e)'de verilen görsel sonuçlardan görüldüğü üzere SNR değeri arttıkça yöntemin doğruluk oranı artmaktadır. Yöntemde 25 dB'de bile yaklaşık olarak 0.94 doğruluk oranı elde edilmiştir.

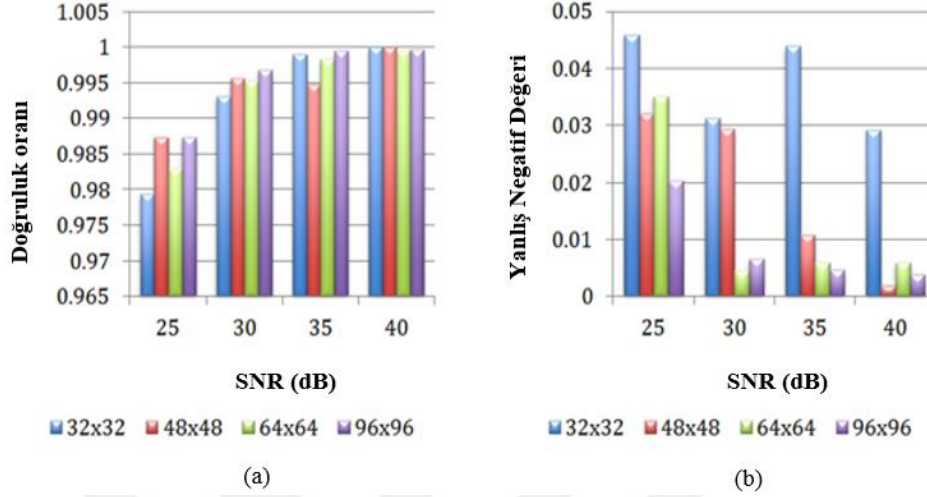
Ayrıca yöntemin JPEG sıkıştırma karşı dayanıklılığı test edilmiştir. Bunun için sahte görüntü 65 ve 75 kalite faktörüyle sıkıştırılmıştır. Şekil 3.4.(c) ve 3.4.(f)'de yöntemin görsel çıktısı ile birlikte doğruluk oranları verilmektedir. Yöntemden sahte görüntünün 65 kalite faktörü ile sıkıştırılması durumunda dahi 0.88 doğruluk oranı elde edilmektedir. Yöntemin JPEG sıkıştırma karşı sağlamlığı, sonuçlarda belirtildiği gibi oldukça iyidir.



Şekil 3.4. Gauss Bulanıklaştırma, $w = 5 \times 5, \sigma = 1$ (b) AWGN (Beyaz Gauss Gürültüsü) (SNR = 25dB) (c) JPEG qf=65 (d) Gauss Bulanıklaştırma $w = 5 \times 5, \sigma = 3$ (e) AWGN (SNR=30dB) (f) JPEG qf=75

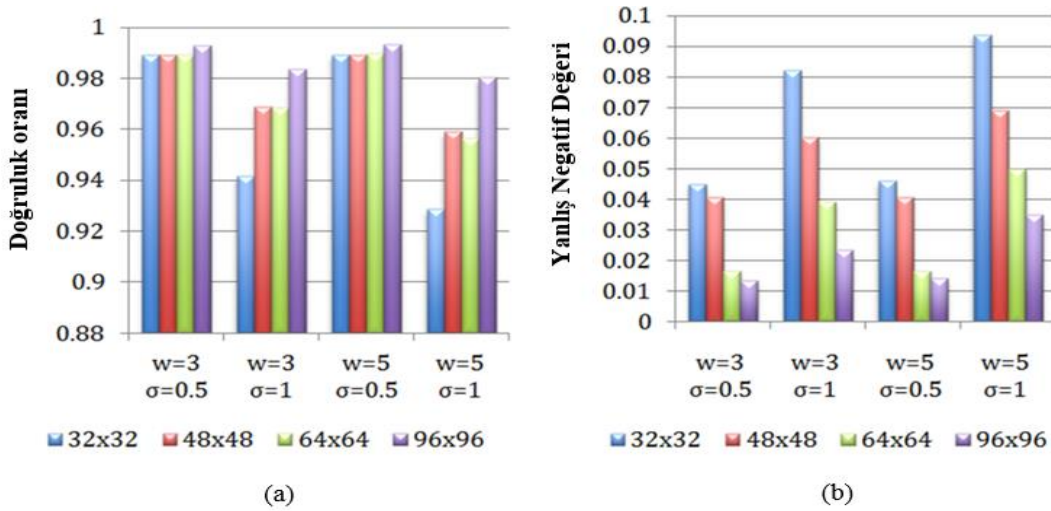
Çalışma ayrıca, çeşitli son işlem operasyonları altında yöntemin sağlamlığını test etmek için niceliksel olarak değerlendirilir. Bu test, sonuçların literatürdeki diğer çalışmalarla kıyaslanabilmesi için yapılmaktadır. Bunun için CoMoFod veritabanından alınan 100 orijinal görüntüden 32×32 , 48×48 , 64×64 ve 96×96 büyüklüğündeki karesel bölgeler kopyalanıp yapıştırılarak 400 sahte görüntü oluşturulmuştur. Oluşturulan sahte görüntülere AWGN, JPEG sıkıştırma ve Gauss bulanıklaştırma uygulanmıştır. Şekil 3.5.(a) ve 3.5.(b)'de bu veri setine AWGN uygulanması sonucu elde edilen doğruluk oranı

ve yanlış negatif değerleri verilmektedir. Şekilden de görüldüğü üzere bölge büyüklüğü arttıkça yöntemin doğruluğu artmaktadır. Yöntem 25 dB gürültüde dahi 0.978'den daha yüksek doğruluk oranına sahiptir.



Şekil 3.5. 25, 30, 35, 40 dB AWGN için (a) doğruluk oranları (b) yanlış negatif değerleri

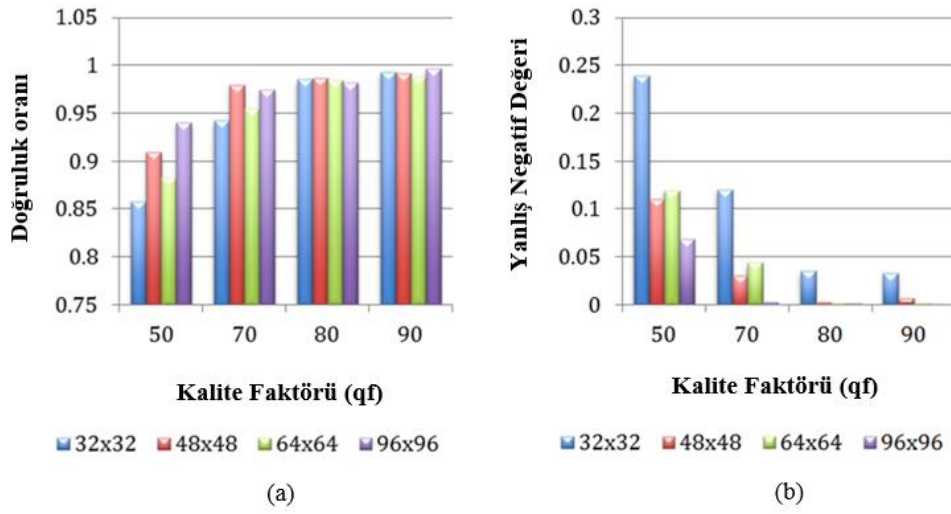
Farklı parametrelerle elde edilen Gauss bulanıklaştırma sonucu elde edilen doğruluk oranları ise Şekil 3.6.'da verilmektedir.



Şekil 3.6. Pencere büyüklüğü 3, 5 ve standart sapma $\sigma=0.5, 1$ Gauss bulanıklaştırması sonucu (a) Doğruluk oranları (b) Yanlış negatif değerleri

Şekilden de görüldüğü üzere 32x32'lik oldukça küçük bir bölge ile kopyala-yapıştır sahteciliği yapılmış görüntülere 5x5 çerçevede 1 standart sapma ile bulanıklaştırma işlemi

uygulanmış olma durumunda bile 0.92 ‘den daha yüksek bir doğruluk oranı elde edilmiştir. Şekil 3.7.’de de sahte görüntüler farklı kalite faktörlerinde sıkıştırılarak yöntemin JPEG sıkıştırmaya olan dayanıklılığı test edilmiştir. Kalite faktörü arttıkça yöntemin doğruluğu da artmaktadır. Bunun yanında önerilen yöntem 50 kalite faktörü ile sıkıştırılmış sahte görüntüde bile 0.85 doğrulukla tekrarlanan bölgeleri işaretleyebilmektedir.



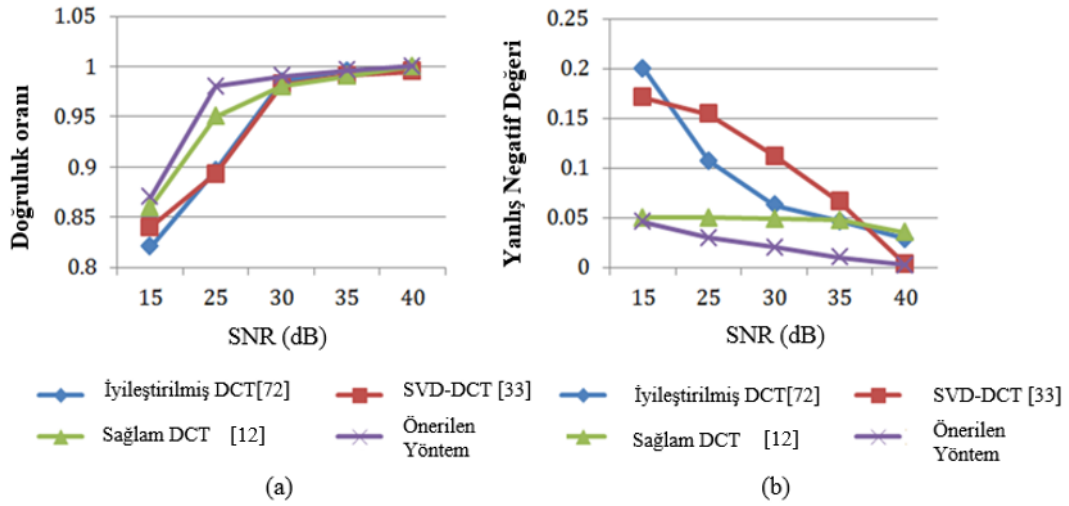
Şekil 3.7. Kalite faktörü $q=50,70,80,90$ ’da JPEG sıkıştırma sonucu elde edilen (a) Doğruluk oranları (b) Yanlış negatif değerleri

Üçüncü deneyde önerilen yöntemin sonuçları literatürdeki benzer üç çalışma ile kıyaslanmaktadır. Önerilen yöntem özellik çıkarımında CLD’yi kullandığı için kıyaslama yapılacak çalışmalar AKD tabanlı çalışmalardan seçilmiş olup bu çalışmalar İyileştirilmiş AKD [72], Sağlam AKD [12] ve AKD-SVD [33]’tür. CLD ile özellik çıkarımında AKD temel teşkil etmektedir.

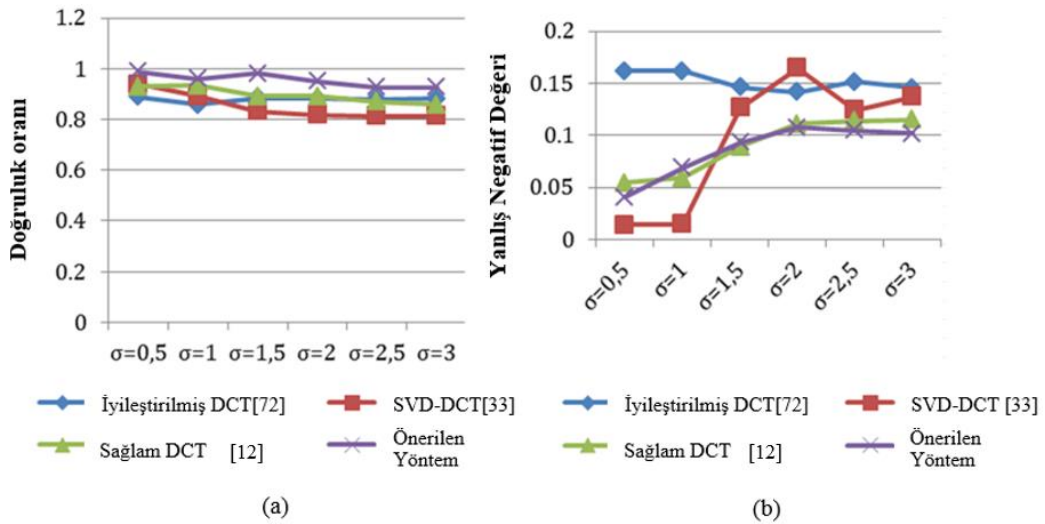
İlk olarak sahte görüntülere 25, 30, 35, 40 dB’lik beyaz Gauss gürültüsü eklenmiş ve elde edilen doğruluk oranı ve yanlış negatif değerlerinin ortalaması alınmıştır. Elde edilen doğruluk oranı ve yanlış negatif değerleri Şekil 3.8.’de verilmektedir. Görüldüğü üzere önerilen yöntem diğer çalışmalara göre yüksek doğruluk oranına sahip olmakla beraber daha düşük de yanlış negatif değerleri vermektedir.

İkinci olarak sahte görüntülere Şekil 3.9.’da görüldüğü gibi 5x5 penceresinde [0.5-3] aralığındaki standart sapma ile bulanıklaştırma işlemi uygulanmıştır. Şekil 3.9.(a)’da verildiği gibi önerilen yöntem diğer yöntemlere göre daha yüksek doğruluğa sahiptir.

Bununla birlikte standart sapmanın 3 olduğu yüksek bulanıklaştırma durumunda bile diğer yöntemler 0.8 civarında doğruluk oranı verirken önerilen yöntemin doğruluk oranı 0.9'dur.



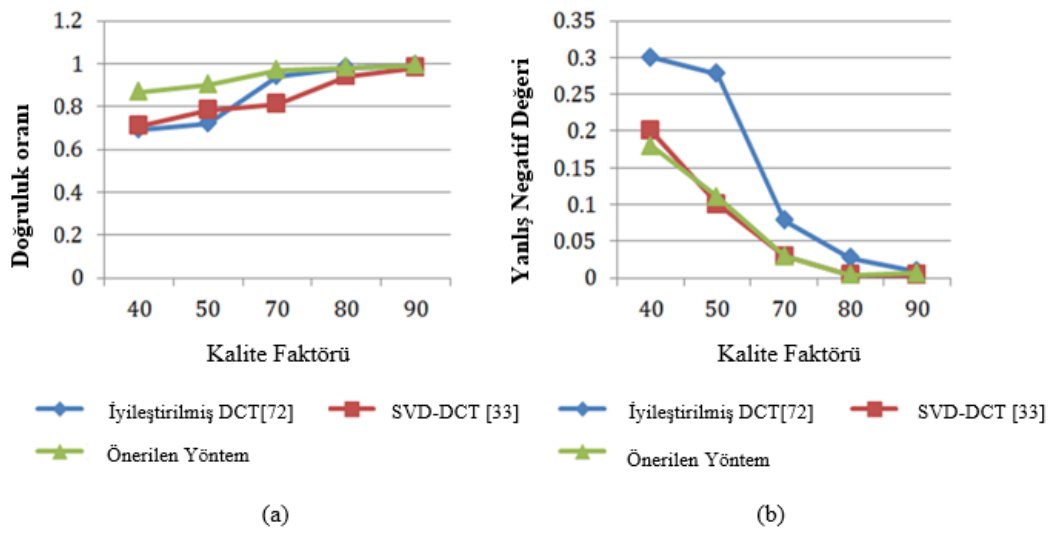
Şekil 3.8. SNR=15, 25, 30, 35, 40 dB AWGN sonucu yöntemlerin doğruluk oranları (b) Yanlış negatif değerleri



Şekil 3.9. (a) Gausss bulanıklaştırma sonucu yöntemlerin doğruluk oranları (b) Yanlış negatif değerleri

Şekil 3.9.(b)'de ise önerilen yöntemin standart sapmanın 1.5'dan yüksek olduğu değerlerde diğer yöntemlere göre daha düşük f değerine sahip olduğu gözükmemektedir. Bu da gösteriyor ki önerilen yöntem Gausss bulanıklaştırma işlemine karşı diğer yöntemlere göre oldukça dayanıklıdır.

Son olarak ise yöntem diğer yöntemlerle JPEG sıkıştırma dayanıklılık açısından kıyaslanmıştır. Sahte görüntüler kalite faktörü 40, 50, 70, 80 ve 90 ile sıkıştırılmıştır. Şekil 3.10.'da yöntemin doğruluk oranları ve yanlış negatif değerleri verilmektedir. Görüntü kalite faktörü 40 ile sıkıştırılırsa dahi yöntemin doğruluk oranı 0.86 olarak elde edilmektedir. Bununla birlikte yöntemin yanlış negatif değerleri diğer yöntemlerden düşüktür. Bu deneye Sağlam AKD [12] çalışması JPEG sıkıştırma karşı dayanıklı olmadığı için dahil edilmemiştir.



Şekil 3.10. Kalite faktörü 40, 50, 70, 80, 90 iken JPEG sıkıştırma sonucu elde edilen Doğruluk oranları (b) Yanlış negatif değerleri

Önerilen yöntem yapılan çalışmalar kısmında anlatıldığı gibi benzerlik araştırması aşamasının paralel yapılabilmesi için bir görüntüdeki blokların renk dağılımlarına göre kümelenebileceğini varsaymaktadır. Blokların kümelendirilmesi benzer renk dağılmış blokların eşzamanlı iş parçacıkları tarafından işlenebilmesini sağlar. Böylece yöntem özellik çıkarımı ve benzer blokların eşleştirilmesi için harcanan süreyi bu şekilde oldukça düşürmektedir. Tablo 3.4.'te eşleştirme algoritmasının öbekleme öncesi ve sonrasında farklı test görüntüsü boyutları için koşma zamanlarını göstermektedir. Tablodan da görüldüğü üzere öbekler üzerindeki benzerlik araştırmasının paralel yapılması algoritmanın koşum zamanını düşürmektedir. Görüntü boyutu arttıkça çalışma zamanı da artmaktadır. Bununla birlikte eşleştirme algoritmasının tek bir iş parçacığı tarafından koşulması yerine paralel koşulması büyük bir zaman kazancı getirmektedir. Tablodan da

görüldüğü üzere 512 x 512 boyutundaki bir test görüntüsü için tek iş parçacığı ile algoritmanın çalışma zamanı yaklaşık 476 saniye iken paralel koşum sonucunda bu süre yaklaşık 299 saniyeye kadar düşmüştür. Bu da özellikle büyük boyutlu görüntülerde yöntemin zaman etkin olduğunu göstermektedir.

Tablo 3.4. Blokların sınıflandırılmasının koşum zamanına etkisi

	116x116	200x200	512x512
Öbekleme yapılmadan	20.5	67.3	476.7
Öbekleme yapılarak	11.8	44.7	299.6

Çalışmada Renk Momentleri ve Renk düzeni tanımlayıcısına dayalı bir yöntem önerilmiştir. Yapılan test sonuçlarından da görüldüğü üzere önerilen yöntem AWGN, Gauss bulanıklaştırmaya ve JPEG sıkıştırma karşı yüksek dayanıklılık göstermekte ve literatürdeki diğer AKD tabanlı çalışmalara göre daha yüksek doğruluk oranı ve daha düşük yanlış negatif değeri vermektedir. Ayrıca blokların özellik çıkarımından önce sınıflandırılması ve sınıfların iş parçacıklarına dağıtılması sonucu işlem zamanında iyileştirilmeye gidilmiştir.

3.4. Eşik Değerinin Otomatik Olarak Belirlendiği Kopyala-Yapıştır Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar

Çalışmada AKD-faz terimleri kullanılarak görüntü bloklarından özellik çıkarılmış ve Benford'un Genelleştirilmiş yasası ile test görüntüsünün sıkıştırma geçmişi hakkında bilgi edinilerek Kopyala-yapıştır sahteciliğini tespit eden yeni bir yöntem önerilmiştir. Kopyala-yapıştır sahteciliğinin tespiti için literatürde önerilmiş olan yöntemler görüntüyü örtüşen alt bloklara bölerek blokların özellik çıkarımında çeşitli yöntemler kullanmaktadır. Elde edilen özellik vektörleri arasındaki benzerliğin araştırılması ile sahtecilik tespiti gerçekleştirilmektedir. Ancak bu yöntemler farklı test görüntüleri farklı eşik değeri gerektirmesine rağmen benzerlik araştırmasında önceden belirlenmiş bir eşik değerini kullanmaktadır [12, 14, 33, 35, 70, 72]. Özellik vektörü elemanlarının aralığı önceden belirlenemediği için en iyi eşik değerin belirlenmesi başka bir problemdir. Bu çalışmada AKD-faz terimleri kullanılarak özellik vektörleri elemanlarının değer aralığı sınırlanmış ve Benford'un Genelleştirilmiş yasası ile test görüntüsünün sıkıştırma geçmişi hakkında bilgi

edinilmiştir. Yönteme ilişkin detaylar yapılan çalışmalar kısmında verilmektedir. Önerilen yöntemin etkinliğini gösterebilmek ve literatürdeki benzer yöntemlerle kıyaslamasını gerçekleştirebilmek amacıyla yapılan testler ve elde edilen deneysel sonuçlar aşağıda verilmektedir.

Deneylerde algoritmada kullanılan parametreler $q_t=16$, $t_f=100$, $t_c = 2.8$, $t_d = 40$, $t_{shift} = 100$ olarak belirlenmiştir.

Yöntemin etkinliğini göstermek adına yapılan deneyleri dört grup altında toplayabiliriz.

İlk deneyde; hazırlanan sahte görüntü veri tabanı üzerinde kalite faktörü belirleme algoritmasının etkinliği gösterilecektir.

İkinci deneyde; sahte bölgenin düzgün/düzgün olmayan şekilde ve sahte bölgenin birden çok olması durumunda yöntemin etkinliği gösterilecektir.

Üçüncü deneyde; sahte bölgenin çeşitli ataklar (JPEG sıkıştırma, Gauss Bulanıklaştırma ve Beyaz Gauss gürültüsü) sonucundaki dayanıklılığı görsel sonuçlar halinde gösterilecektir.

Son deneyde ise yöntem literatürdeki benzer çalışmalarla [12, 14, 33, 35, 70, 72] kıyaslanacaktır.

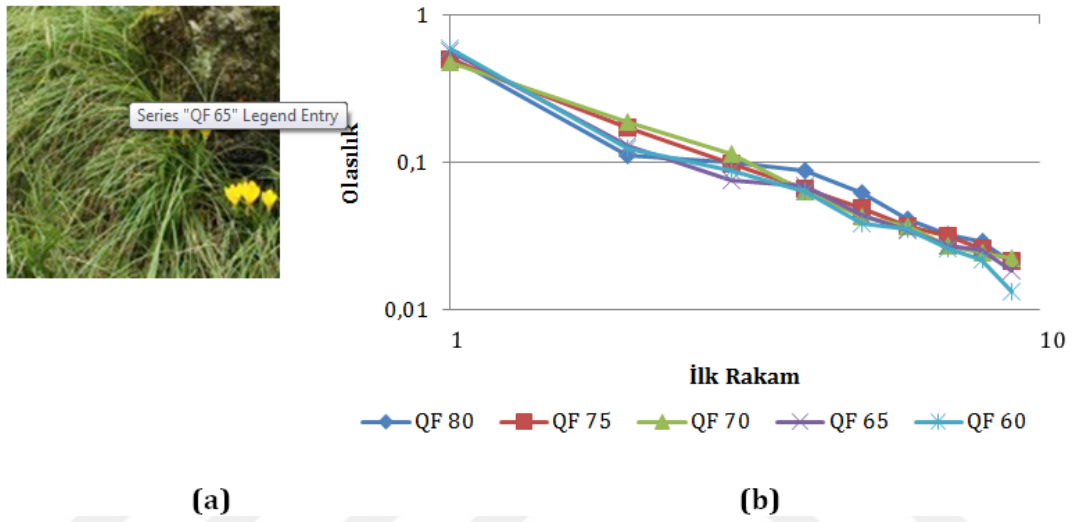
Kalite Faktörünün Belirlenmesi: Önerilen yöntem kalite faktörünün belirlenebilmesinde Benford'un İlk Rakam Dağılımı Yasasını kullanmaktadır. Bu yöntem tarafımızdan oluşturulan veri tabanı ile test edilmektedir. Şekil 3.11.(a) veri tabanından alınan ve 75 kalite faktörü ile sıkıştırılmış görüntüyü göstermektedir.

Tablo 3.5. En iyi beş uydurma için model parametreleri sonuçları

Q-factor	Model Parametleri			Uyum Derecesi (SSE)
	N	q	s	
80	0.2925	0.6545	-0.9826	0.00136
75	0.9976	1.36	-0.5346	1.474e-06
70	1.532	1.609	-0.05068	1.4119e-04
65	0.4588	1	-0.9408	2.1330e-04
60	0.4679	1.033	-0.9418	3.2541e-04

Bu testteki amaç sıkıştırmada kullanılan kalite faktörünün tahmin edilebilmesidir. Bunun için test görüntüsü kalite faktörü 30'dan 100'e kadar 5'er artımlarla tekrar sıkıştırılmaktadır. Yeniden sıkıştırılmış görüntüler Benford'un genelleştirilmiş yasasına

uydurulmaktadır. Tablo 3.5, test görüntüsü için en iyi beş uydurma sonucunu göstermektedir. Tablodan da görüldüğü üzere en küçük uyum derecesi değeri SSE kalite faktörü 75 için elde edilerek, test görüntüsünün kalite faktörü 75 olarak belirlenmiştir. Şekil 3.11.(b)'de de bu en iyi sıkıştırılmış beş görüntünün ilk rakam dağılımları log-log koordinat sisteminde gösterilmiştir. Şekilden de görüldüğü üzere 75 kalite faktöründeki eğri Benford eğrisine en iyi uydurulan eğridir.

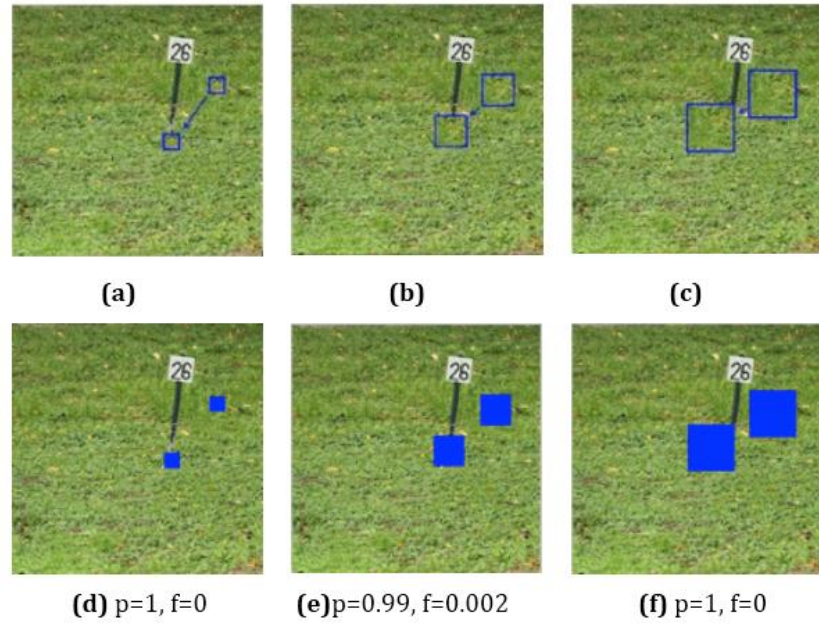


Şekil 3.11. (a) Kalite faktörü 75 ile sıkıştırılmış görüntü (b) En iyi yeniden sıkıştırılmış beş görüntünün ilk rakam dağılımları

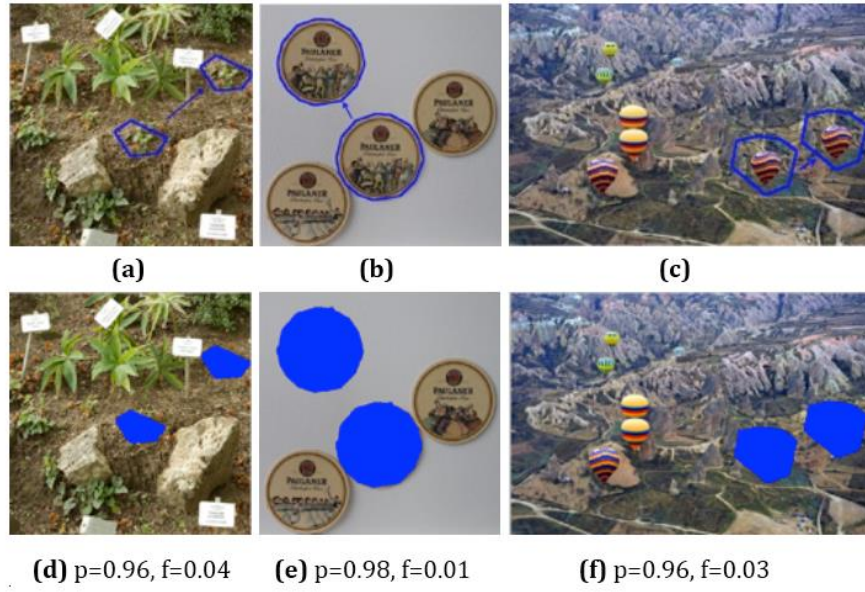
Düzenli/Düzensiz Bölgeler ve Çoklu Kopyala Yapıştır Sahteciliği için Sonuçlar: Blok tabanlı yöntemler sahte bölgeleri test etmek için görüntüyü birbiri ile örtüşen alt bloklara ayrıştırmaktadır. Sahte bölge düzgün olmayan bir şekle sahip olduğunda doğruluk ve yanlış negatif oranları değişmektedir. Bu sebeple literatürdeki çalışmalar sahte bölgenin düzgün/düzensiz olmayan şekle sahip olması durumundaki sonuçlarını rapor etmektedir. İlk deney olarak kopyalanıp yapıştırılan bölgenin düzgün bir şekle sahip olması durumundaki sonuçlara bakılmıştır. Bunun için aynı görüntüde kopyalanan bölgenin boyutu 32x32, 64x64 ve 96x96 alınarak üç ayrı sahte görüntü oluşturulmuştur. Şekil 3.12.(a), 3.12.(b) ve 3.12.(c) sahte görüntüleri göstermektedir. Mavi kareler kopyalanıp yapıştırılan bölgeleri belirtmektedir. Bu karelerin boyutu sırasıyla 32x32, 64x64 ve 96x96'dır. Önerilen yöntemin görsel sonuçları sırasıyla Şekil 3.12.(d), 3.12.(e) ve 3.12.(f)'de verilmektedir. Şekillerden de görüldüğü üzere yöntem kopyalanıp yapıştırılan bölgeleri oldukça iyi bir

şekilde tespit edebilmektedir. Şekil 3.12.(d) ve 3.12.(f)'nin doğruluk oranlarının 1 olması tüm bölgelerin yanlışsız tespit edildiğini göstermektedir.

İkinci deney olarak sahte bölgenin düzgün olmayan bir şekle sahip olması durumundaki performansı test edilmiştir. Sahte görüntüleri oluşturmak için üç test görüntüsü kullanılmıştır. Şekil 3.13.(a), 3.13.(b) ve 3.13.(c) sahte görüntüleri göstermektedir.



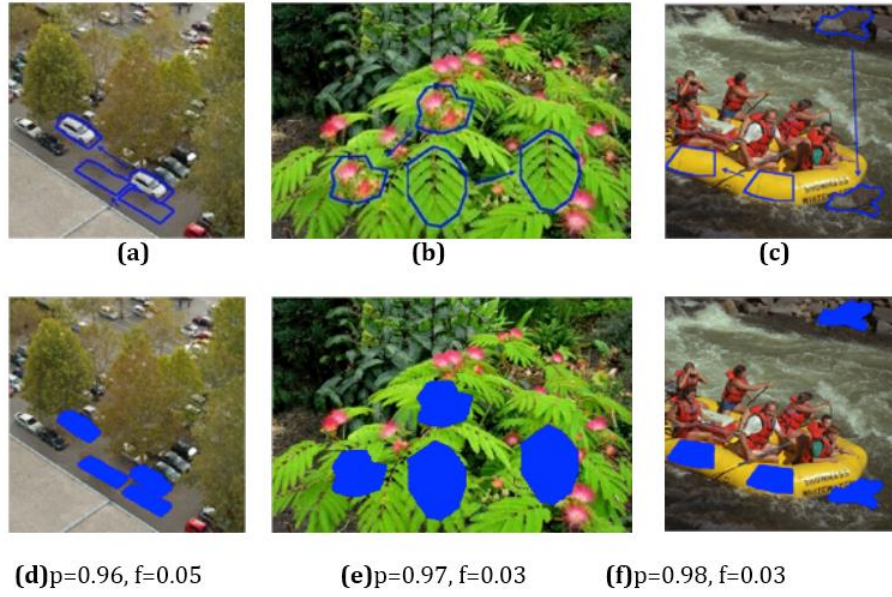
Şekil 3.12. (a) (b) (c) 32x32, 64x64 ve 96x96 büyüklüğünde kopyala-yapıştır yapılmış sahte görüntüler (d) (e) (f) Yöntemin görsel sonuçları



Şekil 3.13. (a)(b)(c) Sahte görüntüler (d)(e)(f) Yöntemin görsel sonuçları

Algoritmanın görsel sonuçları ise sırasıyla Şekil 3.13.(d), 3.13.(e) ve 3.13.(f)'de verilmektedir. Doğruluk ve yanlış negatif oranlarına bakıldığında doğruluk oranı 0,96'dan büyük, yanlış negatif oranı ise 0.04'den küçüktür. İkinci görüntüdeki nesnelerin arka plandan farklı olması sebebiyle en iyi doğruluk oranı 0.98 olarak bu resimden elde edilmiştir.

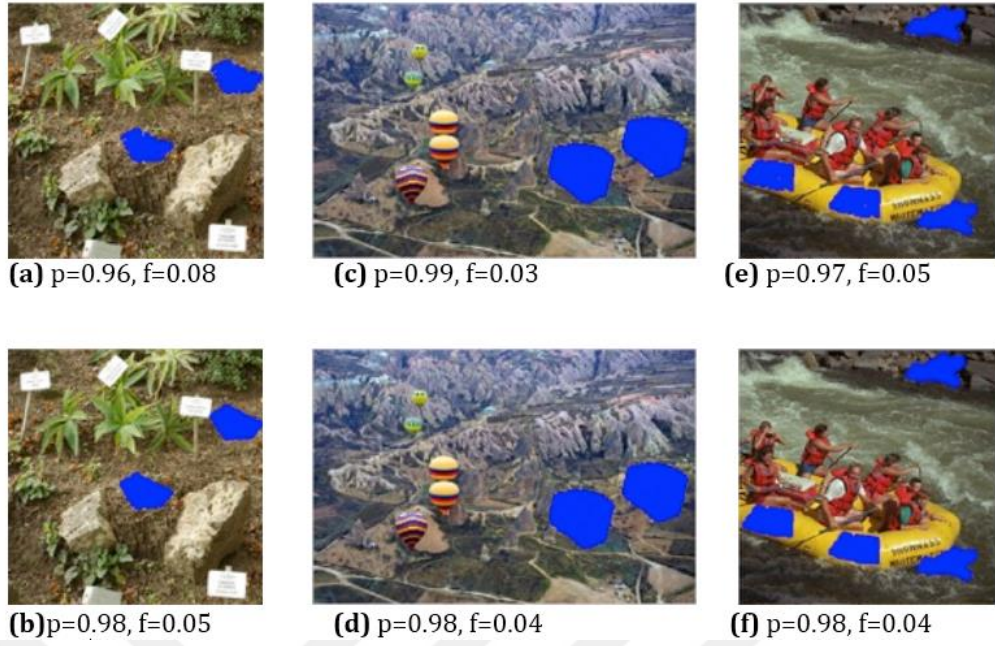
Üçüncü deney olarak ise sahte bölgenin görüntü içerisinde birden çok olması durumunda yöntemin performansı test edilmiştir. Üç test görüntüsü oynanarak sahte görüntüler oluşturulmuştur. Şekil 3.14.(a), 3.14.(b) ve 3.14.(c) sahte görüntüleri göstermektedir. Algoritmanın ürettiği görsel sonuçlar ise Şekil 66(d), 66(e) ve 66(f)'de sırasıyla gösterilmektedir. Görüldüğü üzere görüntülerden elde edilen doğruluk oranı 0.96'nın üzerindeyken yanlış negatif oranı 0.05'nin altındadır. Sonuçlar yöntemin çoklu kopyala-yapıştır sahteciliğini de tespit edebildiğini göstermektedir.



Şekil 3.14. (a)(b)(c) Sahte görüntüler (d)(e)(f) Yöntemin görsel sonuçları

Ataklara karşı yöntemin görsel sonuçları: Yöntemin son işlem operasyonlarına dayanıklılığını test etmek için sahte görüntülere bu işlemler uygulanmıştır. Bunun için Şekil 3.13.(a), Şekil 3.13.(c) ve Şekil 3.14.(c) 'deki sahte görüntüler alınmıştır. Şekil 3.13.(a)'ya Şekil 3.15.(a) ve 3.15.(b)'de görüldüğü üzere kalite faktörü 30 ve 70 olarak JPEG sıkıştırma uygulanmıştır. Şekil 3.13.(c)'deki görüntü $w=5$, $\sigma = 1$ ve $\sigma = 3$ ile Gaussian filtresi ile bulanıklaştırılarak sırasıyla Şekil 3.15.(c), 3.15.(d) elde edilmiştir. Son olarak da Şekil 3.14.(c)'deki görüntüye SNR=25 ve 30 dB'lik beyaz Gauss gürültüsü eklenmiş ve sırasıyla Şekil 3.15.(e), 3.15.(f) elde edilmiştir. Bu görüntülerin sonuçları sırasıyla Şekil 3.15.(a)-3.15.(f)'de görülmektedir.

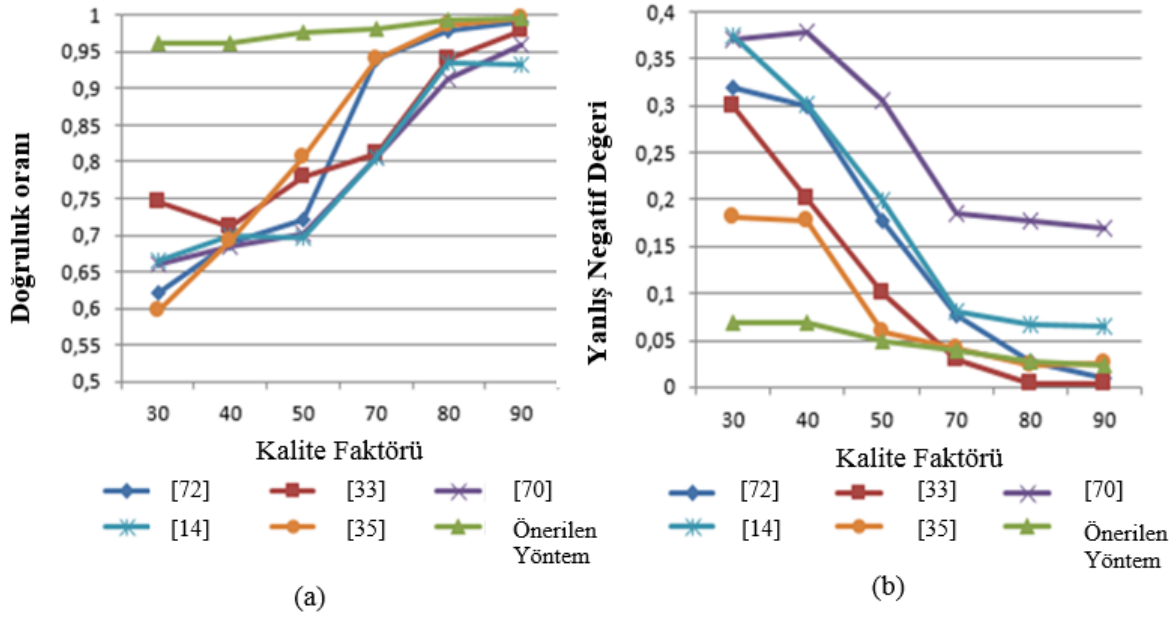
Görsel sonuçlardan da görüldüğü üzere görüntülere çeşitli son işlem operasyonları uygulansa dahi yöntem sahte bölgeleri tespit edebilmektedir. Şekil 3.15.(a) ve 3.15.(b) yöntemin JPEG sıkıştırma sonucundaki performansını göstermektedir. Kalite faktörü arttıkça yöntemin doğruluk oranı artmaktadır. Bununla birlikte kalite faktörü 30'da dahi 0.96 doğrulukla sahte bölgeler tespit edebilmektedir. Yine $\sigma = 1$ ve 3 bulanıklaştırmaları sonucunda Şekil 3.15.(c) ve Şekil 3.15.(d)'de görüldüğü üzere yöntem sahte bölgeleri doğruluk oranı 0.98'den daha yüksek tespit edebilmektedir. Gürültü sonucunda ise de yaklaşık olarak doğruluk oranı 0.97 gibi yüksek bir değer elde edilmektedir.



Şekil 3.15. (a) (b) Kalite faktörü $QF=30$ ve 70 olan görüntüler (c) (d) $w=5$, $\sigma = 1$ ve 3 ile Gauss bulanıklaştırması uygulanmış görüntüler (e) (f) $SNR = 25$ ve 30 dB'lik gürültü eklenmiş görüntüler

Karşılaştırma Testleri: Bu kısımda önerilen yöntem literatürdeki benzer çalışmalarla kıyaslanacaktır. Önerilen yöntem özellik çıkarımında AKD faz yönteminden faydalandığı için, kıyaslama yapılacak olan çalışmalar literatürdeki AKD tabanlı çalışmalardan seçilmiştir. Kopyala-yapıştır sahteciliği tespitinde literatürde AKD tabanlı olarak üç çalışma ön plana çıkmaktadır [12, 33, 72]. Bu çalışmalarla önerilen yöntem doğruluk oranı ve yanlış negatif değerlerine göre kıyaslanacaktır. Aynı zamanda önerilen yöntem sonuçları özellik çıkarımında TBA, FMT ve DWT'yi kullanan diğer çalışmalarla [14, 35, 70] kıyaslanmıştır.

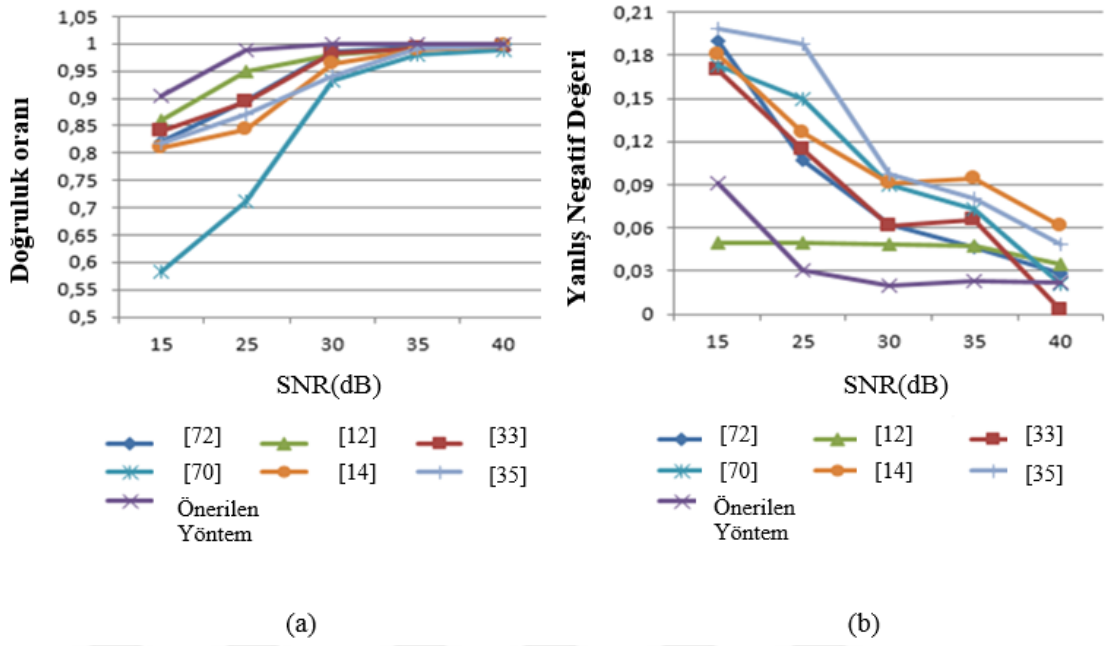
İlk deneyde, yöntemin ve diğer çalışmaların JPEG sıkıştırma sonuçları kıyaslanmıştır. Bu deneyde Cao vd. [12] çalışması JPEG sıkıştırmalara dayanıklı olmadığı ve sonuçlarında sıkıştırma sonuçlarına yer vermedikleri için, bu çalışma yer almayacaktır. Şekil 3.16.(a)'da görüldüğü üzere önerilen yöntemin doğruluk oranı tüm kalite faktörleri için 0.96 'nın üstündedir. Yöntemin üstünlüğü kalite faktörü 70 'den küçük olduğunda görülmektedir. Şekil 3.16.(b)'de verilen yanlış negatif değerleri incelendiğinde, özellikle kalite faktörü düştükçe yöntem daha başarılı sonuçlar vermektedir.



Şekil 3.16. JPEG sıkıştırma uygulandığında yöntemin [12, 14, 33, 35, 71, 73] çalışmaları ile (a) doğruluk oranına göre (b) Yanlış negatif değerlerine göre kıyaslanması

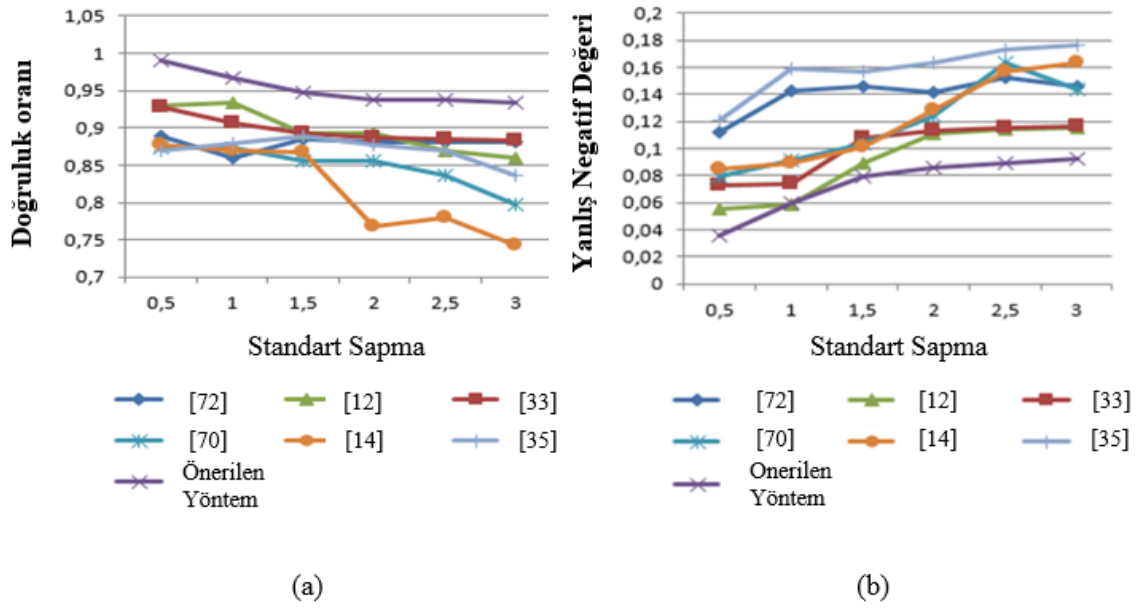
Yapılan ikinci deneyde yöntemlerin Gaussss gürültü sonuçları kıyaslanmıştır. Sahte görüntülere 15, 25, 30, 35 ve 40 dB'lik beyaz Gaussss gürültüsü eklenmiştir. Ortalama doğruluk oranları ve yanlış negatif değerleri sırasıyla Şekil 3.17.(a) ve 3.17.(b)'de verilmektedir. Şekil 3.17.(a)'dan da görüldüğü üzere SNR değeri 30'dan küçük olsa dahi yöntem sahte bölgeleri tespit edebilmektedir.

Son deney olarak ise Gaussss bulanıklaştırma uygulandığında yöntemin diğer yöntemlerden farkı gösterilmiştir. Şekil 3.18.(a) ve Şekil 3.18.(b) yöntemlerin sırasıyla doğruluk oranları ve yanlış negatif değerlerini göstermektedir. Sahte görüntüler $w=5$ ve $\sigma = 0.5, 1, 1.5, 2, 2.5, 3$ parametleri uygulanarak Gaussss filtresinden geçirilmiştir. Şekil 3.18.(a) ve 3.18.(b)'den görüldüğü üzere yöntem diğer çalışmalara göre daha yüksek doğruluk oranı ve daha düşük yanlış negatif değeri vermektedir.



Şekil 3.17. Görüntülere AWGN uygulanması sonucunda önerilen yöntem ile [12, 14, 33, 35, 71, 73] çalışmalarının (a) doğruluk oranına göre (b) Yanlış negatif değerlerine göre kıyaslanması

Son işlem operasyonlarından JPEG sıkıştırmaya, Gauss bulanıklaştırma ve beyaz Gauss gürültüsüne dayanıklı bir Kopyala-yapıştır sahteciliği tespit yöntemi önerilmiştir. Yöntem Benford'un genelleştirilmiş yasasını kullanarak test görüntüsünün sıkıştırma geçmişini belirlemekte ve AKD faz ile özellik çıkararak özellik vektörlerinin değer aralığını sınırlamaktadır. Bunun sonucunda yöntem iki vektörün kıyaslamasında eşit eleman sayılarını değerlendirmekte ve test görüntüsünün sıkıştırma geçmişini kullanarak otomatik eşik değeri belirleyebilmektedir. Yöntem literatürdeki diğer yöntemlerden farklı olarak her bir test görüntüsü için farklı eşik değeri otomatik olarak seçebilmektedir. Deneysel sonuçlardan da görüldüğü üzere önerilen yöntem literatürdeki benzer çalışmalara kıyasla daha yüksek doğruluk oranı ve daha düşük yanlış negatif değeri vermektedir.



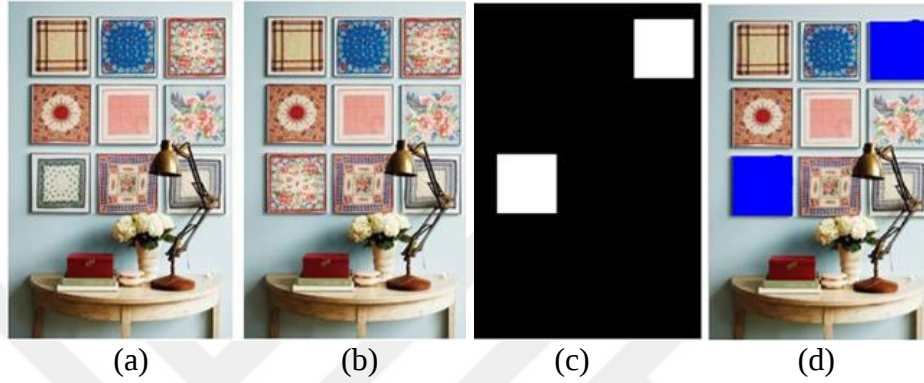
Şekil 3.18. Görüntülere Gauss Bulanıklaştırma uygulandığında önerilen yöntemle [12, 14, 33, 35, 71, 73] çalışmalarının (a) doğruluk oranına göre (b) Yanlış negatif değerlerine göre karşılaştırılması

3.5. LBP-AKD'ye Dayalı Kopyala-Yapıştır Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar

Literatürde kopyala-yapıştır sahteciliğinin tespiti için önerilen AKD tabanlı yöntemlerde amaç yöntemi JPEG sıkıştırma ataklarına karşı dayanıklı hale getirmektir. Önerilen çalışmada LBP ve AKD'ye dayalı yeni bir kopyala-yapıştır sahteciliği tespit yöntemi gerçekleştirilmiştir. Literatürde sahtecilik tespitinde ilk defa LBP ve AKD yöntemleri birlikte kullanılarak diğer AKD tabanlı yöntemlere göre daha etkin sonuçlar elde edilmiştir. Çalışmada öncelikle görüntü birbiri örtüşen alt bloklara parçalanmakta ve her bir bloktan LBP kullanılarak özellik çıkarılmaktadır. Sonrasında LBP ile etiketlenen bloklar AKD yöntemiyle frekans uzayına geçilmiştir. AKD'si alınmış her bir bloğun zigzag taraması sonucu elde edilen ilk 15 katsayının işaret bilgisi ile birlikte bloğun Y, Cb, Cr değerlerinin ortalaması o blok için özellik olarak çıkarılmıştır. Yönteme ilişkin detaylar yapılan çalışmalar kısmında verilmektedir. Önerilen yöntemin etkinliğini gösterebilmek ve literatürdeki benzer yöntemlerle kıyaslamasını gerçekleştirebilmek amacıyla yapılan testler ve elde edilen deneysel sonuçlar aşağıda verilmektedir.

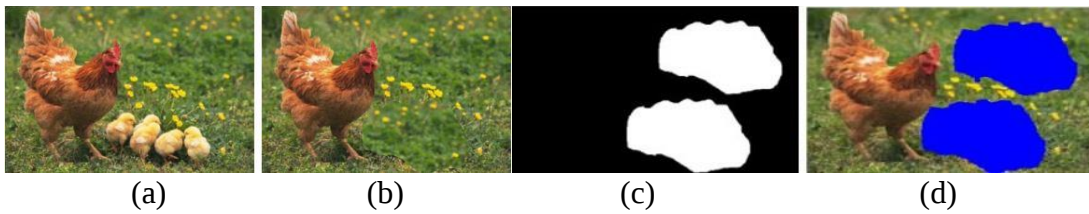
Testler esnasında kullanılan parametreler $t_n = 50$, $t_c = 2.8$, $t_d = 40$, $t_s = \frac{13}{15} = 0.86$, $t_{\text{shift}} = 200$ olarak alınmıştır.

Yapılan ilk testte kopyala-yapıştır yapılan bölgenin düzgün bir şekle sahip olması durumundaki önerilen yöntemin görsel sonuçlarına bakılmıştır. Şekil 3.19.(a)'da verilen orijinal görüntü Şekil 3.19.(c)'deki maske kullanılarak Şekil 3.19.(b)'deki sahte görüntü oluşturulmuştur. Şekil 3.19.(d)'den de görüldüğü üzere yöntem kopyala-yapıştır yapılmış bölgeleri doğruluk oranı ve yanlış negatif oranı sırasıyla 1 ve 0.01 olarak tespit etmiştir.



Şekil 3.19. (a) Orijinal görüntü (b) Sahte görüntü (c) Maske (d) Tespit sonucu

İkinci testte ise kopyala-yapıştır yapılan bölgenin şeklinin düzgün olmama durumundaki sonuç değerlendirilmiştir. Orijinal görüntü, sahte görüntü ve maske görüntüsü sırasıyla Şekil 3.20.(a), 3.20.(b) ve 3.20.(c)'de verilmiştir. Şekil 3.20.(d)'den görüldüğü üzere sahte bölgenin şeklinin düzgün olmaması durumunda dahi önerilen yöntem kopyala-yapıştır yapılmış bölgeleri tespit edebilmektedir. Tespit sonucu doğruluk oranı ve yanlış negatif oranı yaklaşık olarak sırasıyla 0.99 ve 0.04 olarak hesaplanmıştır.



Şekil 3.20.(a) Orijinal görüntü (b) Sahte görüntü (c) Maske (d) Tespit sonucu

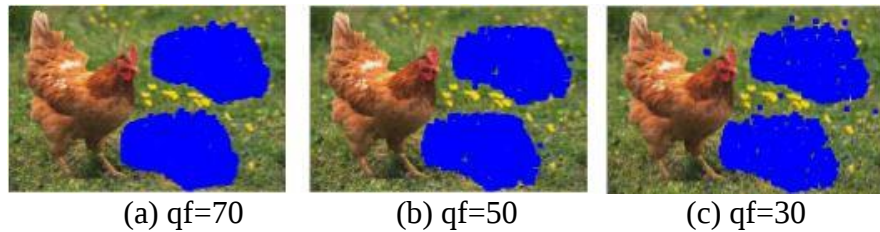
Bir başka deney olarak kopyalanan bölgenin aynı görüntü içerisindeki birden çok alana yapıştırılması durumu yani çoklu kopyala-yapıştır sahteciliği gerçekleştirildiğindeki önerilen yöntem sonuçlarına bakılmıştır. Şekil 3.21.(a)'daki orijinal görüntüden iki bölge

alınarak bu bölgeler yine aynı görüntü içerisindeki başka alanlara yapıştırılarak Şekil 3.21.(b)'deki sahte görüntü oluşturulmuştur. Çoklu kopyala-yapıştır yapılması durumunda ise önerilen yöntemin tespit sonuçları Şekil 3.21.(d)'de verilmektedir. Orijinal görüntüye çoklu kopyala yapıştır sahteciliği yapılsa dahi önerilen yöntem bu görüntüyü 0.97 doğruluk oranı ile tespit edebilmektedir. Önerilen yöntem sahte görüntüye JPEG sıkıştırma uygulanması durumunda da test edilmiştir. JPEG sıkıştırmada çeşitli kalite faktörleri kullanılarak yöntemin etkinliği gösterilecektir.



Şekil 3.21. (a) Orijinal görüntü (b) Sahte görüntü (c) Maske (d) Tespit sonucu

Şekil 3.22'de JPEG sıkıştırma sonucu yöntemin tespit sonuçları görsel olarak verilmiştir. Sonuçlardan da görüldüğü üzere sahte görüntü 30 kalite faktörü ile dahi sıkıştırılmış olsa da, yöntem bu görüntüdeki sahte bölgeleri 0.92 doğruluk oranı ile tespit etmiştir. Bu üç kalite faktörü için elde edilen yanlış negatif değerleri ise 0.1 değerinden küçüktür.



Şekil 3.22. Çeşitli kalite faktörlerinde elde edilen tespit sonuçları

Önerilen yöntem literatürdeki benzer AKD tabanlı yöntemlerle [12,72] kıyaslanarak yöntemin çeşitli ataklar altındaki etkinliği gösterilmiştir. Karşılaştırma amacıyla üç test gerçekleştirilmiştir. İlk deney olarak Tablo 3.6'da sahte görüntünün çeşitli kalite faktörlerinde sıkıştırıldığı zamanki yöntem sonuçları verilmiştir. Sonuçlardan görüldüğü üzere önerilen yöntem [72] ile kıyaslandığında daha yüksek doğruluk oranı ve daha düşük

yanlış negatif değeri vermiştir. Tabloda [12] çalışmasının JPEG sıkıştırmaya dayanıklı olmaması sebebiyle sonuçlarına yer verilmemiştir. Bir başka test olarak önerilen yöntem literatürdeki benzer çalışmalarla Gauss bulanıklaştırma atağı uygulanması sonucunda karşılaştırılmıştır. Bunun için sahte görüntülere pencere boyutu $w=5$ ve standart sapma $\sigma = 0.5, \sigma = 1$ olacak şekilde Gauss filtresi uygulanmıştır. Tablo 3.7’de benzer çalışmalar [12,72] ve önerilen yöntemin p ve f değerleri verilmektedir. Önerilen yöntem oldukça büyük bulanıklaştırma seviyelerinde dahi diğer yöntemlerle kıyaslandığında daha yüksek doğruluk değeri vermiştir. Son testte ise önerilen yöntem gürültü eklenmiş sahte test görüntüleri üzerinde test edilmiştir. Sahte görüntülere $SNR = 25$ dB ve 40 dB’lik beyaz Gauss gürültüsü eklenmiştir. Önerilen yöntem Tablo 3.8’de görüldüğü üzere sahte görüntüye 25 dB’lik gürültü eklense dahi yüksek doğruluk değerine sahip olmaktadır.

Tablo 3.6. Yöntemlerin JPEG sıkıştırma sonucunda karşılaştırılması

<i>QF</i>	<i>Önerilen yöntem</i>		<i>[72]</i>	
	<i>p</i>	<i>f</i>	<i>p</i>	<i>f</i>
Q=90	0.999	0.008765	0.996	0.009
Q=80	0.987	0.02135	0.98	0.027
Q=50	0.969	0.05955	0.729	0.277
Q=40	0.9513	0.1121	0.729	0.277

Tablo 3.7. Yöntemlerin Gauss bulanıklaştırma sonucunda karşılaştırılması

	<i>Önerilen yöntem</i>		<i>[72]</i>		<i>[12]</i>	
	<i>p</i>	<i>f</i>	<i>p</i>	<i>f</i>	<i>p</i>	<i>f</i>
w=3 $\sigma=0.5$	0.993	0.081	0.922	0.091	0.9584	0.0516
w=3 $\sigma=1$	0.998	0.053	0.915	0.089	0.9464	0.0546
w=5 $\sigma=0.5$	0.998	0.062	0.888	0.162	0.93	0.055
w=5 $\sigma=1$	0.984	0.064	0.86	0.162	0.934	0.059

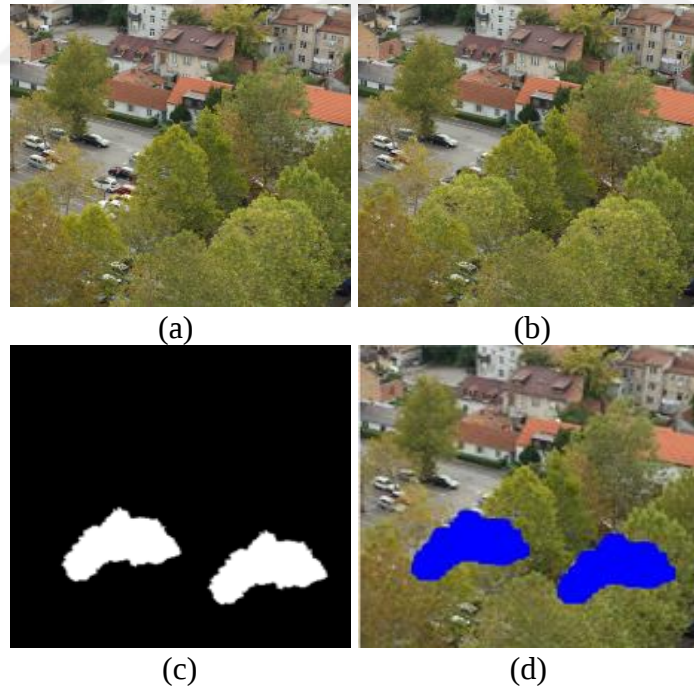
Tablo 3.8. Yöntemlerin Beyaz Gauss gürültüsü sonucunda karşılaştırılması

	<i>Önerilen yöntem</i>		<i>[72]</i>		<i>[12]</i>	
	<i>p</i>	<i>f</i>	<i>p</i>	<i>f</i>	<i>p</i>	<i>f</i>
25 dB	0.9924	0.0368	0.8967	0.1068	0.95	0.05
40 dB	1	0.0399	0.9987	0.0282	1	0.0348

Literatürde ilk olarak Kopyala-yapıştır sahteciliği tespitinde LBP ile AKD yöntemi birleştirilerek yeni bir yöntem önerilmiştir. Önerilen yöntem literatürdeki AKD tabanlı çalışmalarla kıyaslandığında çeşitli ataklar altında dahi kopyalanıp yapıştırılan bölgeleri daha yüksek doğruluk oranı ve daha düşük yanlış negatif değerleri ile tespit etmektedir. Bir başka sonuç olarak ise sahtecilik tespitinde AKD katsayılarının işaret bilgisinin yeterli olduğu ortaya konulmaktadır.

3.6. LPQ'ya Dayalı Kopyala-Yapıştır Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar

Görüntüde kopyalanan bölge yapıştırıldıktan sonra yapıştırılan bölge sınırlarının belli olmaması için bulanıklaştırma işlemi uygulanmaktadır. Bu nedenle çalışmada görüntü bloklarının özellik çıkarımında bulanıklaştırma işleminden bağımsız LPQ yönteminden faydalanılmıştır. LPQ yöntemi literatürde ilk kez Kopyala-yapıştır sahteciliği tespitinde kullanılmıştır.

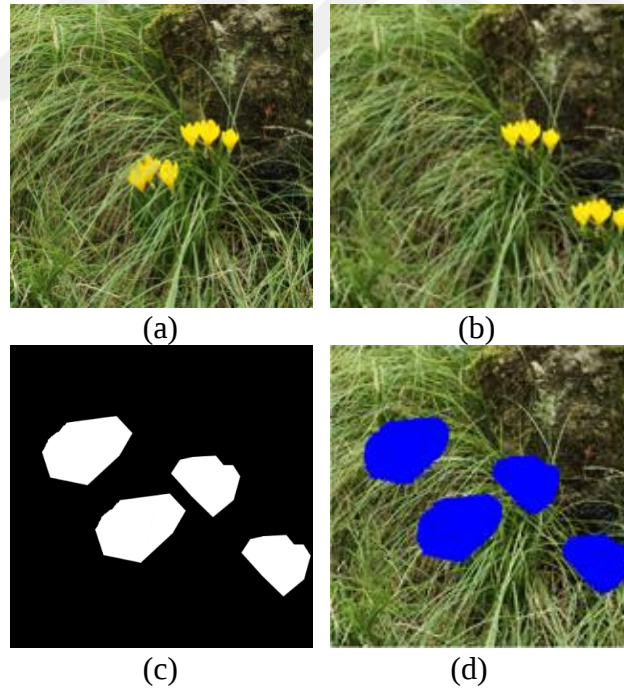


Şekil 3.23. (a) Orijinal görüntü, (b) Sahte görüntü, (c) Maske görüntüsü, (d) Algoritmanın tespit sonucu

Çalışmada LPQ ile etiketlenen blokların sütunlarının ortalama değerleri özellik vektörü olarak çıkarılmıştır. Benzer özelliklerin araştırılması ile de tekrarlanan bölgeler tespit edilmiştir. Yönteme ilişkin detaylar yapılan çalışmalar kısmında verilmektedir. Önerilen yöntemin etkinliğini gösterebilmek ve literatürdeki benzer yöntemlerle kıyaslamasını gerçekleştirebilmek amacıyla yapılan testler ve elde edilen deneysel sonuçlar aşağıda verilmektedir.

Yapılan testlerde kullanılan parametreler $t_s=1.5$, $t_d=32$, $t_n=100$ ve $qt=10$ olarak alınmıştır.

İlk deney olarak sahte görüntüye herhangi bir son işlem operasyonu uygulanmadığı zaman elde edilen sonuçlar verilmiştir. Şekil 3.23.(a) ve 3.23.(b) sırasıyla orijinal ve sahte görüntüyü göstermektedir. Sahte görüntünün maskesi ve algoritmanın tespit sonucu ise sırasıyla Şekil 3.23.(c) ve 3.23.(d)'de verilmektedir. Şekilden de görüldüğü üzere sahte bölge düzgün olmayan bir şekle sahip olsa dahi önerilen algoritma bu bölgeleri 0.98 gibi yüksek bir doğruluk oranıyla tespit edebilmektedir.

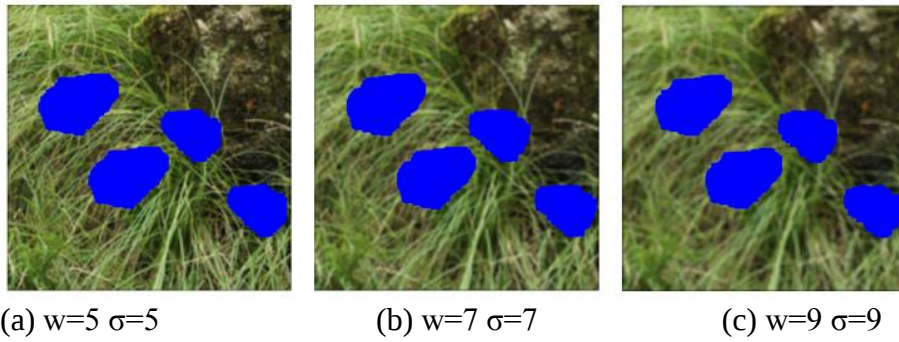


Şekil 3.24. (a) Orijinal görüntü(b) Sahte görüntü(c) Maske görüntüsü(d) Algoritmanın tespit sonucu

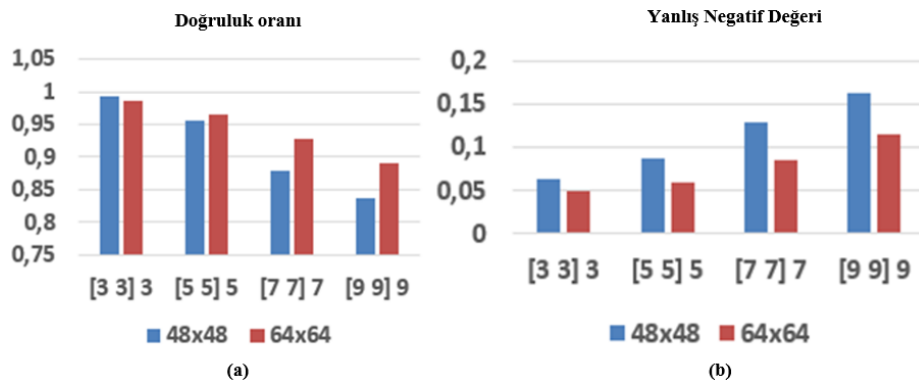
Bir başka deney olarak kopyala-yapıştır işleminin birden çok yapılması yani çoklu kopyala yapıştır yapılması durumundaki algoritmanın etkinliği test edilmiştir. Şekil

3.24.(a)'da verilen orijinal görüntüdeki bölgeler alınarak sahte görüntü oluşturulmuştur. Şekil 3.24(c)'deki maske kullanılarak Şekil 3.24.(b)'de verilen çoklu kopyala-yapıştır ile yapılmış sahte görüntü oluşturulmuştur. Bu sahte görüntünün algoritmaya verilmesi sonucu ise Şekil 3.24.(d)'deki program çıktısı elde edilmiştir. Algoritma tekrarlanan bu bölgeleri 0.98 doğruluk oranı ve 0.04 yanlış negatif değeriyle tespit etmektedir.

Önerilen yöntem bulanıklaştırma işleminden bağımsızdır. Sahte görüntü $w = 5$ $\sigma = 5$, $w = 7$ $\sigma = 7$ ve $w = 9$ $\sigma = 9$ değerleriyle bulanıklaştırıldığında elde edilen program çıktıları Şekil 3.25.(a)-3.25.(c)'de verilmektedir. Sonuçlardan da görüldüğü üzere yöntem $w = 9$ $\sigma = 9$ gibi oldukça yüksek seviyede bulanıklaştırılmış sahte görüntüdeki tekrarlanmış bölgeleri 0.92 doğruluk oranı ile bulabilmektedir. Bu üç sahte görüntü için elde edilen yanlış negatif değerleri ise 0.08'den daha küçüktür.



Şekil 3.25. Farklı bulanıklaştırma seviyelerinde elde edilen program çıktıları

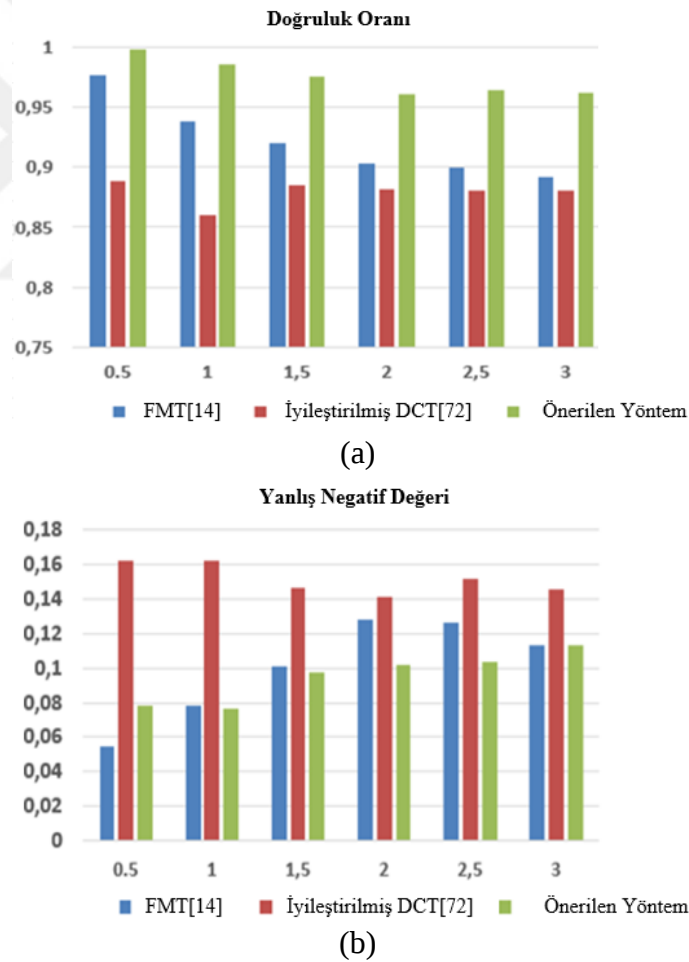


Şekil 3.26.(a) Doğruluk oranları (b) Yanlış negatif değerleri

Diğer bir deney olarak ise hazırlanan veri tabanından kopyala-yapıştır yapılan bölgelerin büyüklüğü 48x48 ve 64x64 olan 30 sahte görüntü alınmıştır. Bu sahte görüntülere Gauss bulanıklaştırma işlemi uygulanmıştır. Bu sahte görüntülerin

algortimaya verilmesi sonucu elde edilen ortalama doğruluk oranları ve yanlış negatif değerleri Şekil 3.26.(a) ve 3.26.(b)'de sırasıyla verilmektedir.

Son deney olarak ise görüntülere bulanıklaştırma işlemi uygulanması durumunda yöntemin performansı literatürdeki benzer yöntemlerle [14, 72] kıyaslanmıştır. Karşılaştırmada kullanılan sahte görüntüler kopyala-yapıştır yapılan bölge büyüklüğü 48x48 olan görüntülerden alınmıştır. Şekil 3.27.(a) ve (b)'de sırasıyla yöntemlerin doğruluk oranları ve yanlış negatif değerleri verilmektedir. Sahte görüntülerin bulanıklaştırılmasında $w=5$ penceresinde $\sigma = 0.5, 1, 1.5, 2, 2.5, 3$ olarak alınmıştır. Şekilden de görüldüğü üzere yöntem diğer çalışmalara oranla daha yüksek doğruluk oranı (0.96'nın üzerinde) vermektedir. Bunun yanında yöntemin yanlış negatif değerleri (0.01'nin altı) de diğer çalışmalara göre düşüktür.

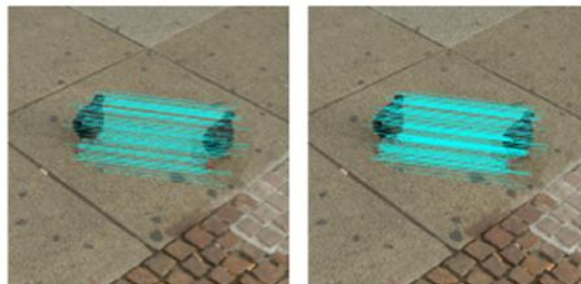


Şekil 3.27. Önerilen yöntemin farklı bulanıklaştırma seviyelerinde diğer yöntemlerle (a) Doğruluk oranına (b) Yanlış Negatif değerine göre kıyaslanması

3.7. Renkli SIFT Yöntemine Dayalı Kopyala-Yapıştır Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar

Literatürde Kopyala-yapıştır sahteciliği tespiti alanında yapılmış olan SIFT tabanlı çalışmalar gri seviye görüntü üzerinde çalışmışlardır. Bu çalışmada anahtar nokta tabanlı yöntemlerden renkli SIFT kullanılarak sahte görüntünün tek bir kanalından anahtar nokta çıkarmak yerine her bir kanalı için anahtar noktalar çıkarılmıştır. Bunun sonucunda SIFT ve renkli SIFT yöntemleri arasında karşılaştırma yapılmıştır. Karşılaştırma sonucunda renkli SIFT yöntemi ile SIFT yöntemine göre daha fazla eşleşen nokta elde edilerek daha etkin sahtecilik tespiti yapıldığı görülmüştür. Eşleşen noktalarla da tekrarlanan bölgeler tespit edilmiştir. Önerilen yöntem, kopyalanan bölgenin döndürme, ölçekleme, görüntünün JPEG sıkıştırmaya tabi tutulmasında dahi tekrarlanan bölgeleri tespit edebilmektedir. Yönteme ilişkin detaylar yapılan çalışmalar kısmında verilmektedir. Önerilen yöntemin etkinliğini gösterebilmek ve literatürdeki benzer yöntemlerle kıyaslamasını gerçekleştirebilmek amacıyla yapılan testler ve elde edilen deneysel sonuçlar aşağıda verilmektedir.

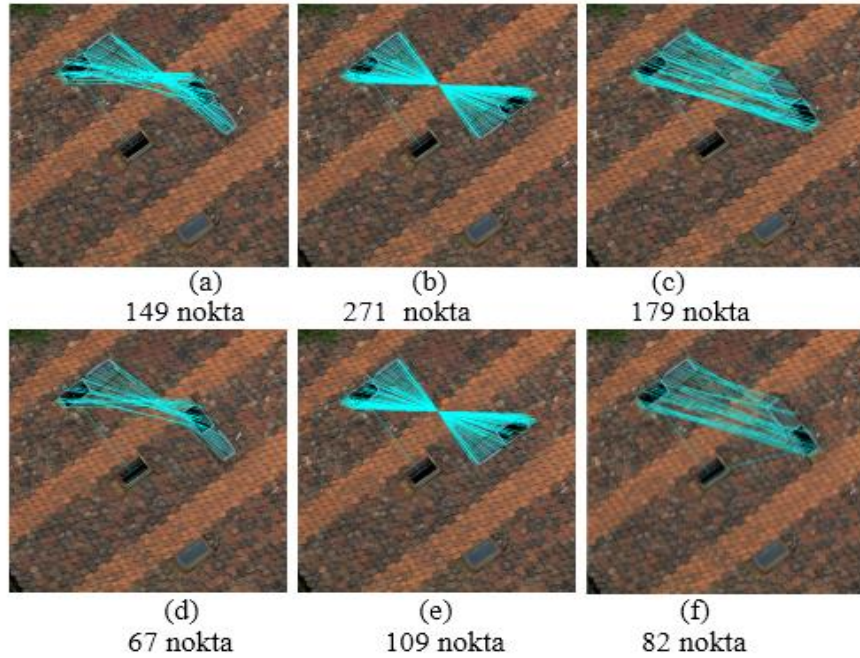
Gerçekleştirilen ilk deneyde ataksız görüntü üzerinde SIFT ve renkli SIFT yöntemleri ile anahtar noktalar çıkarılmış ve eşleştirilmiştir. Şekil 3.28.(a)'da görüntünün SIFT sonucunda Şekil 3.28.(b)'de ise renkli SIFT sonucunda eşleşen anahtar noktaları gösterilmiştir. Deney sonucunda SIFT yöntemiyle 98 eşleşen nokta elde edilirken, renkli SIFT yöntemi sonucunda 208 nokta elde edilmiştir. Şekilden de görüldüğü üzere Renkli SIFT yöntemiyle daha fazla eşleşen nokta elde edilerek kopyalanmış bölge daha etkin olarak tespit edilmiştir.



(a) SIFT 98 eşleşme (b) Renkli SIFT 208 eşleşme

Şekil 3.28.(a) SIFT sonucunda (b) Renkli SIFT sonucunda eşleşen noktalar

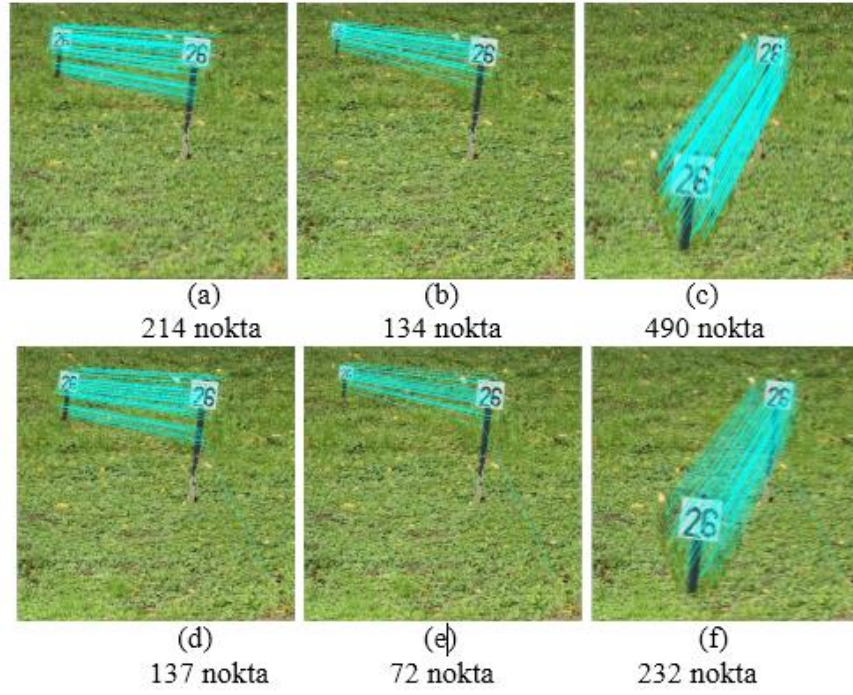
İkinci deney olarak sahte görüntü oluşturulurken kopyalanan bölge farklı açılarda 90, 180 ve 270 derece döndürüldükten sonra yapıştırılmış ve önerilen yöntem ile SIFT'in dönme atağına karşı dayanıklılığı test edilmiştir. Şekil 3.29.(a), 3.29.(b) ve 3.29.(c) sırasıyla kopyalanan bölgenin 90, 180 ve 270 derece ile döndürülmesi sonucu oluşturulmuş sahte görüntülerin Renkli SIFT sonucunda, Şekil 3.29.(d) 3.29.(e) ve 3.29.(f) ise bu görüntülerin SIFT sonucunda eşleşen nokta sayısını göstermektedir. Renkli SIFT ve SIFT yönteminin sonuçları kıyaslandığında, kopyalanmış bölgeler arasında renkli SIFT yöntemi ile SIFT yöntemine göre neredeyse iki katı kadar eşleşmiş nokta bulunmuştur.



Şekil 3.29. (a) 90 derece (b)180 derece (c)270 derece döndürme sonucunda renkli SIFT ile(d) 90 derece(e) 180 derece (f) 270 derece döndürme sonucunda SIFT ile eşleşen noktalar

Bir başka deney olarak renkli SIFT ile SIFT yönteminin ölçekleme atağına karşı dayanıklılığı kıyaslanmıştır. Bunun için sahte görüntü oluşturulurken kopyalanan bölge büyültme veya küçültme operasyonu yapıldıktan sonra yapıştırılmıştır. Şekil 3.30.(a), 3.30.(b), 3.30.(c) sırasıyla 0.3, 0.5 ve 1.5 ölçek sonucunda oluşturulan sahte görüntülerin renkli SIFT ile Şekil 3.30.(d), 3.30.(e), 3.30.(f) ise bu görüntülerin SIFT ile eşleşen nokta sayısını göstermektedir. Şekillerden de görüldüğü üzere renkli SIFT yöntemi ölçekleme

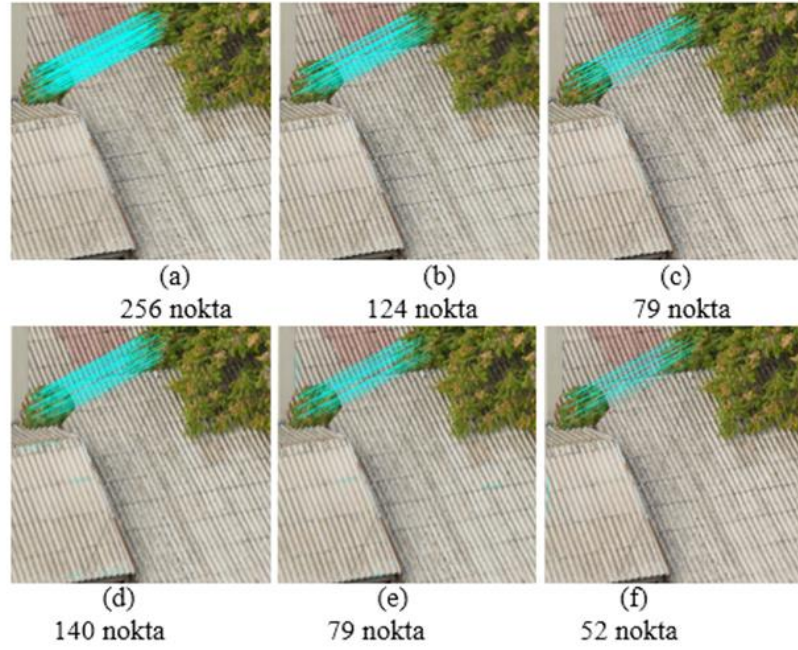
atağına karşı oldukça etkin tespit yapabilmekte ve döndürme işleminde olduğu gibi SIFT yöntemine göre neredeyse iki katı kadar eşleşmiş nokta üretmektedir.



Şekil 3.30. (a) Ölçek 0.3 (b) Ölçek 0.5 (c) Ölçek 1.5 sonucunda renkli SIFT ile (d) Ölçek 0.3 (b) Ölçek 0.5 (c) Ölçek 1.5 sonucunda SIFT ile elde edilmiş eşleşen noktalar

Yapılan en son deney olarak renkli SIFT yönteminin JPEG ataklara olan dayanıklılığı test edilmiştir. Bunun için oluşturulan sahte görüntü farklı kalite faktörlerinde $KF=90, 50, 30$ sıkıştırılarak renkli SIFT ve SIFT yöntemi sonucunda elde edilen eşleşen nokta sayılarına bakılmıştır. Görsel sonuçlar Şekil 3.31.'de gösterilmektedir. Şekillerden de görüldüğü üzere sahte görüntüye JPEG sıkıştırma uygulanması sonucunda da Renkli SIFT ile SIFT'e göre daha çok eşleşen nokta elde edilmiştir.

Çalışmada, Kopyala-yapıştır sahteciliğinin tespitinde, görüntü özelliklerinin çıkarımında literatürde ilk olarak renkli SIFT algoritmasının kullanılması gerçekleştirilmiştir. Aynı zamanda renkli SIFT'in SIFT'e göre daha etkin sahtecilik tespiti yaptığı gösterilmiştir.



Şekil 3.31. (a) $KF=90$ (b) $KF=50$ (c) $KF=30$ ile kaydedilmiş sahte görüntülerin renkli SIFT sonucunda (d) $KF=90$ (e) $KF=50$ (f) $KF=30$ ile kaydedilmiş görüntülerin SIFT sonucunda eşleşen noktaları

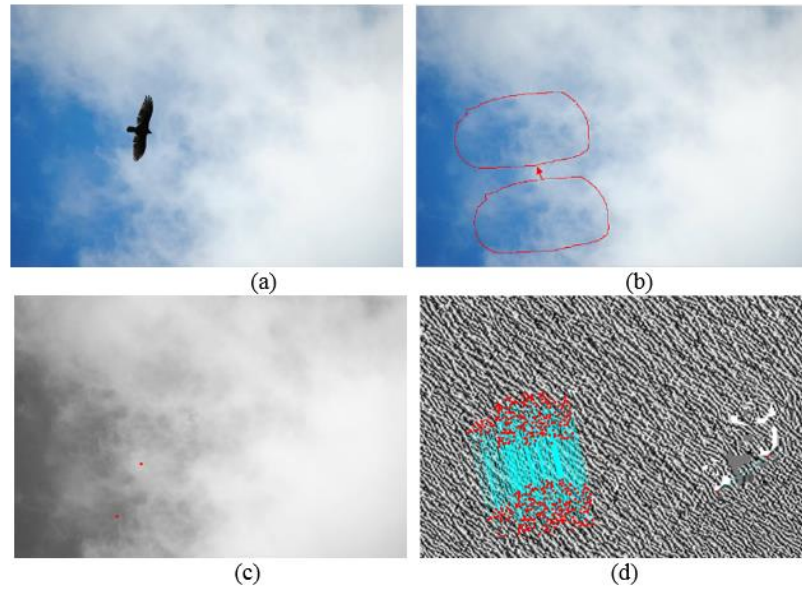
3.8. LPQ ve SIFT Yöntemlerine Dayalı Anahtar Nokta Tabanlı Kopyala-Yapıştır Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar

Çalışmada LPQ ile SIFT yöntemleri kullanılarak düz geçişsiz bölgelerde dahi yapılan kopyala-yapıştır sahteciliğini tespit eden yeni bir yöntem önerilmiştir. Yöntemde anahtar nokta çıkarımından önce LPQ yöntemiyle doku çıkarılmıştır. SIFT yönteminden LPQ ile elde edilen doku görüntüsünden anahtar nokta çıkarmada faydalanılmıştır. Anahtar noktaların eşleştirilmesi ile tekrar eden bölgeler tespit edilmektedir. Çalışmada anahtar nokta çıkarımından önce doku çıkarımının yapılması ile anahtar nokta tabanlı pasif görüntü doğrulama yöntemleri iyileştirilmiştir. Elde edilen sonuçlar yöntemin son işlem operasyonlarına da dayanıklı olduğunu göstermektedir. Yönteme ilişkin detaylar yapılan çalışmalar kısmında verilmektedir. Önerilen yöntemin etkinliğini gösterebilmek ve literatürdeki benzer yöntemlerle kıyaslamasını gerçekleştirebilmek amacıyla yapılan testler ve elde edilen deneysel sonuçlar aşağıda verilmektedir.

Yapılan testlerde kullanılan veri tabanının oluşturulmasında Google görsellerden 640x420 ve 1200x800 boyutlarında görüntüler alınmış ve GIMP programı yardımıyla sahte

görüntüler oluşturulmuştur. Sahte görüntüdeki kapatılan bölgenin çevre sınırlarını gizlemek için bölge kenarlarına sınır yumuşatma işlemi uygulanmıştır.

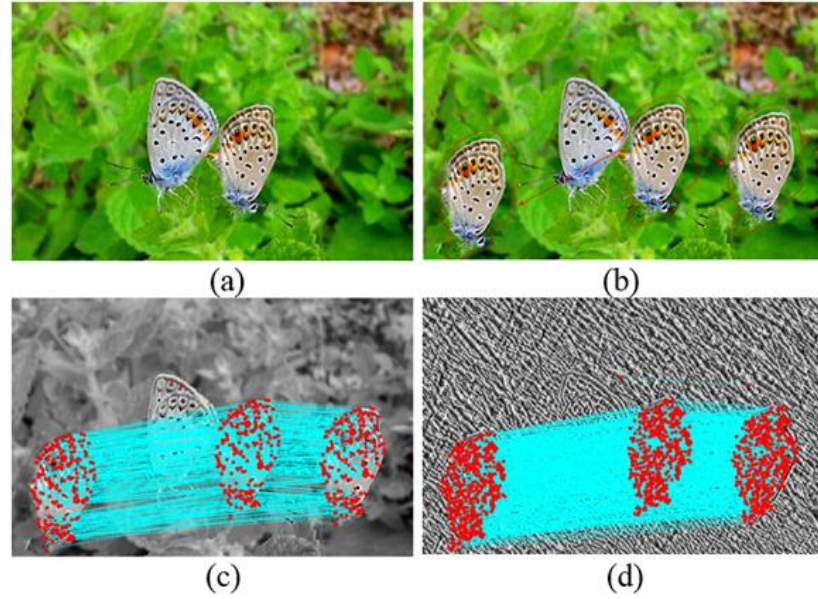
İlk deney olarak tekli ve çoklu kopyala-yapıştır yapılarak sahte görüntüler oluşturulmuştur. Şekil 3.32.(b) tekli kopyala-yapıştır yapılarak oluşturulmuş sahte görüntüye bir örnektir. Şekil 3.32.(a)'daki kuşlu bölge yine aynı görüntü içerisindeki düz bir bölge ile kapatılarak Şekil 3.32.(b)'deki sahte görüntü oluşturulmuştur. SIFT yöntemi Şekil 3.32.(c)'de görüldüğü üzere bu bölge düz bir bölge olduğu için 2 anahtar nokta bulmakta ve bu noktaları eşleştirememektedir. Bununla birlikte önerilen yöntem bu sahte görüntü üzerinde 11600 anahtar nokta çıkarıp Şekil 3.32.(d)'de görüldüğü üzere bunların 412 tanesini eşleştirebilmektedir. Önerilen yöntem sonucunda sahte bölgeler üzerinde daha fazla eşleşen nokta bulunmaktadır. Sahte görüntü oluşturma esnasında düz bir bölge ile kapatılma yapılacağı zaman, SIFT tabanlı yöntemler bu bölgeler üzerinde anahtar nokta bulamayacaktır.



Şekil 3.32. (a) Orijinal Görüntü (b) Sahte görüntü (c) SIFT yöntemi sonucu (Anahtar nokta: 2 Eşleşen nokta: 0)(d) Önerilen yöntem sonucu (Anahtar nokta: 11600 Eşleşen anahtar nokta: 412)

Sahte görüntü oluşturulmasında kopyalanan bölge birden çok bölgeye yapıştırılarak çoklu kopyala yapıştır yapılması durumu da bu deney içinde irdelenmiştir. Şekil 3.33.(a)'da görülen kelebek bölgesi kopyalanarak aynı görüntü içerisindeki farklı iki alana

yapıştırılmış Şekil 3.33.(b)'deki kırmızı çizgilerin gösterdiği şekilde sahte görüntü oluşturulmuştur. Bu görüntü için önerilen yöntemin ve SIFT in Şekil 3.33.(c) ve 3.33.(d) görüldüğü üzere toplam eşleşen anahtar nokta sayıları sırasıyla (858, 359) iken anahtar nokta sayıları (15922, 2520)'dir. Önerilen yöntem kopyalanan bölge karmaşık olsa da SIFT ile karşılaştırıldığında daha fazla eşleşen nokta bulmaktadır.

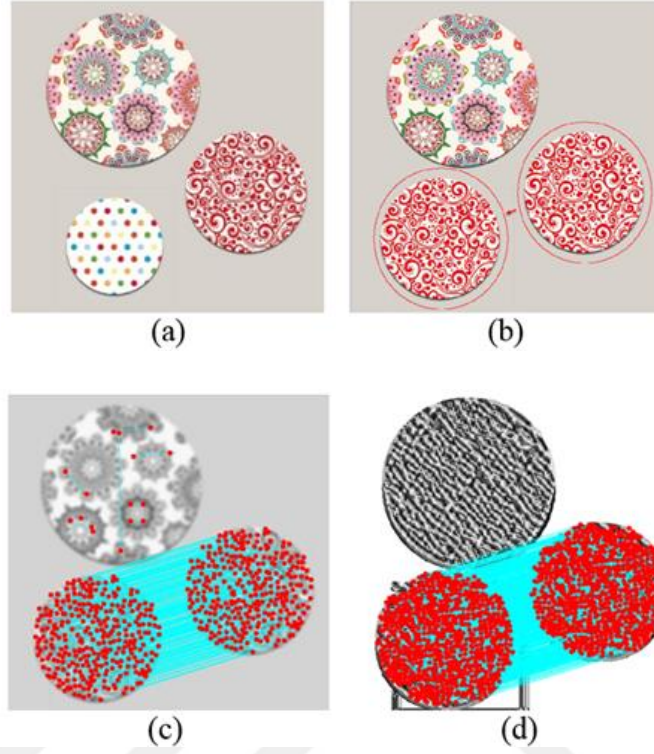


Şekil 3.33.(a) Orijinal görüntü (b) Sahte görüntü (c) SIFT yöntemi sonucu (Anahtar nokta: 2520 Eşleşen anahtar nokta: 359)(d) Önerilen yöntem sonucu (Anahtar nokta: 15922 Eşleşen anahtar nokta: 852)

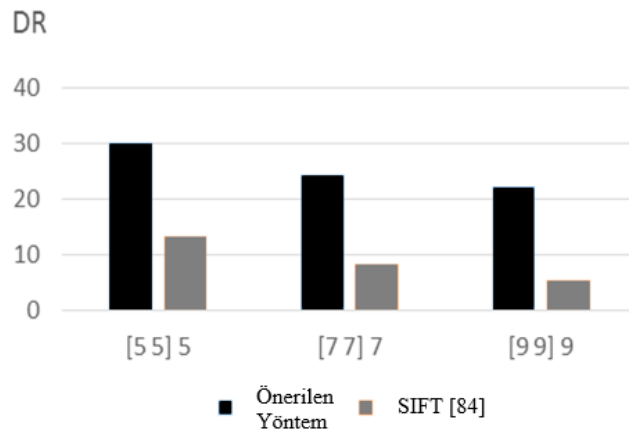
İkinci bir deney olarak bulanıklaştırma işlemi sahte görüntüye uygulanarak önerilen yöntemin bulanıklaştırmaya karşı dayanıklılığı test edilmiştir. Şekil 3.34.(a)'daki kırmızılı desen kopyalanmış diğer desen üzerine yapıştırılarak Şekil 3.34.(b)'deki sahte görüntü oluşturulmuştur. Bu sahte görüntüye $w = 9$ $\sigma = 9$ Gauss filtresi uygulanarak görüntü bulanıklaştırılmıştır. Önerilen yöntem ve SIFT sırasıyla bu görüntü için 5153 ve 1163 anahtar noktası çıkarırken Şekil 3.34.(c) ve 3.34.(d)'de görüldüğü üzere bu noktaların sırasıyla 1789'u ve 637'si eşleşmiştir. Önerilen yöntem SIFT ile kıyaslandığında büyük bulanıklaştırma seviyelerinde dahi daha yüksek tespit oranına sahiptir.

Üçüncü deney olarak önerilen yöntemin son işlem operasyonlarına olan dayanıklılığı test edilmiştir. Bunun için öncelikli olarak 640x420 piksel boyutlarında 50 test görüntüsü alınarak 5x5, 7x7 ve 9x9 pencerelerinde standart sapma $\sigma = 5$, $\sigma = 7$ ve $\sigma = 9$ olacak

şekilde bulanıklaştırma uygulanmıştır. Şekil 3.35'te yöntemlerin ortalama tespit oranları verilmektedir. Sonuçlara bakıldığında önerilen yöntem SIFT'e göre daha yüksek DR değerlerine sahiptir.

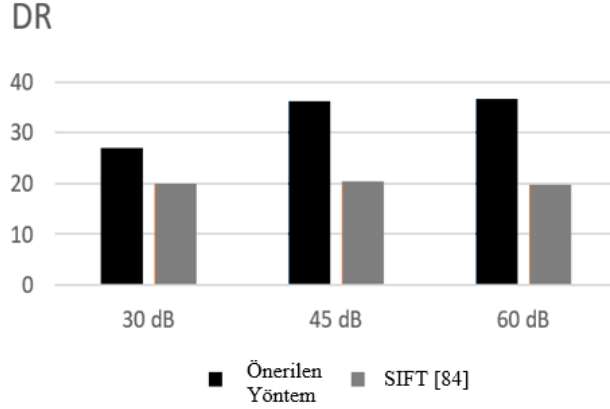


Şekil 3.34. Orijinal görüntü (b) Sahte görüntü (c) SIFT sonucu (Anahtar nokta: 1163 Eşleşen anahtar nokta: 637)(d) Önerilen yöntem sonucu (Anahtar nokta: 5153 Eşleşen anahtar nokta: 1789)



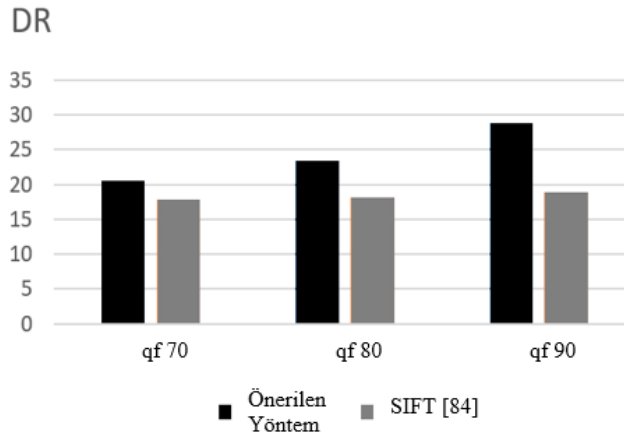
Şekil 3.35. Bulanıklaştırma sonucu DR değerleri

Önerilen yöntemin gürültüye dayanıklılığını test etmek için ise 50 sahte görüntüye 30 dB 45 dB ve 60 dB’lik gürültüler eklenmiştir. Şekil 3.36.’da gürültü karşısında önerilen yöntem ve SIFT’in ortalama DR değerleri gösterilmektedir. Grafikten de görüldüğü üzere önerilen yöntem SIFT’e göre gürültü karşısında da daha yüksek DR değerine sahiptir.



Şekil 3.36. Gürültü sonucu DR değerleri

Çalışmada son test olarak ise 1200x800 piksel boyutlarında 40 sahte görüntü alınarak kalite faktörü $qf=90$, 80 ve 70 olacak şekilde bu görüntülere JPEG sıkıştırma uygulanmıştır. Şekil 3.37.’den de görüldüğü üzere JPEG sıkıştırma uygulansa dahi önerilen yöntem SIFT’e göre daha yüksek doğruluk oranına sahiptir.



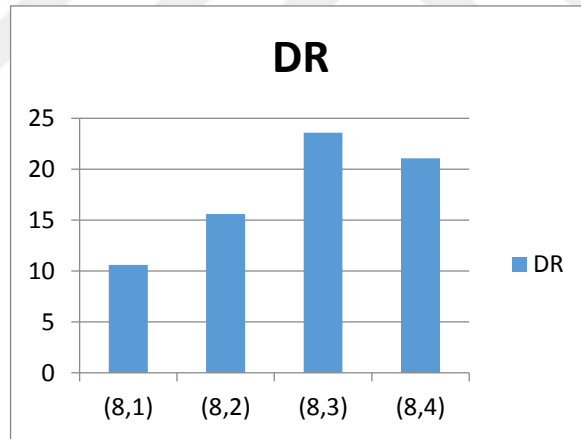
Şekil 3.37. JPEG sıkıştırma sonucu DR değerleri

Çalışmada Kopyala-yapıştır sahteciliği tespiti için yeni bir anahtar nokta tabanlı görüntü doğrulama yöntemi önerilmiştir. Önerilen yöntem anahtar nokta seçimine dayalı olup SIFT uygulanmadan önce LPQ uygulayarak doku bilgisini çıkarmaktadır. LPQ

görüntünün görünüşte düz bölgelerinden doku bilgisi çıkarmaktadır. Bu sayede anahtar nokta çıkarım algoritmaları düz bölgelerden de anahtar nokta çıkarabilmektedir. Önerilen yöntemle blok tabanlı ve anahtar nokta tabanlı yöntemlerin düz bölgelerde gerçekleştirilen sahteciliklerdeki eksikliği giderilmiştir.

3.9. Pasif Görüntü Doğrulama Yöntemlerine Dayalı Medikal Görüntülerde Sahtecilik Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar

Önerilen yöntem literatürde ilk olarak medikal görüntülerde oynanmış bölge tespitinde pasif görüntü doğrulama yöntemlerinden faydalanmaktadır. Yöntemde anahtar nokta çıkarım yöntemlerini daha başarılı hale getirebilmek için medikal görüntüden yapısal doku bilgisi çıkarılmaktadır. Doku bilgisinin çıkarımında rotasyondan bağımsız LBPROT operatöründen faydalanılırken anahtar nokta çıkarımında SIFT kullanılmıştır. Yönteme ilişkin detaylar yapılan çalışmalar kısmında verilmektedir. Önerilen yöntemin etkinliğini gösterebilmek ve literatürdeki benzer yöntemlerle kıyaslamasını gerçekleştirebilmek amacıyla yapılan testler ve elde edilen deneysel sonuçlar aşağıda verilmektedir.

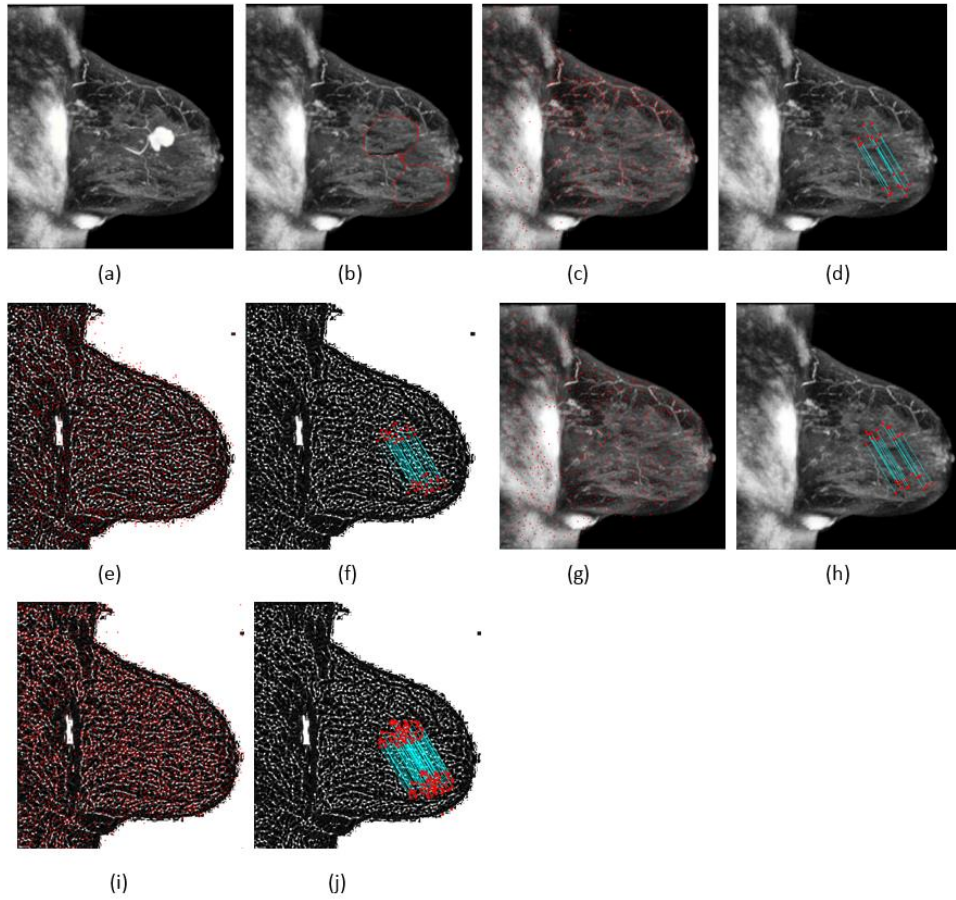


Şekil 3.38. R değerlerine göre elde edilen DR'ler

Medikal görüntülerde oynanmış bölgelerin tespitinde SIFT ile anahtar nokta çıkarımından önce görüntüye LBPROT operatörü uygulanarak doku görüntüsü oluşturulduğundan, DR değerinin maksimize edilmesinde LBPROT operatöründeki R değerinin seçimi önem arz etmektedir. Optimum R değerinin bulunabilmesi için R değeri 1 ile 4 arasında alınarak test görüntülerinden elde edilen ortalama DR değerleri hesaplanmıştır. Şekil 3.38.'den de görüldüğü üzere en yüksek ortalama DR değeri R=3'de elde edildiği için bu değer optimum değer olarak alınmıştır.

Önerilen yöntemin üstünlüğü üç farklı senaryo altında gösterilecektir. İlk iki senaryoda; iki popüler anahtar nokta çıkarım yöntemi SIFT ve SURF kıyaslanarak önerilen yöntemde SIFT anahtar çıkarım yöntemi tercih edilmesi sebebi ortaya konulacaktır. Son olarak ise SURF yönteminin doku bilgisi üzerindeki performansı değerlendirilecektir.

Etkinlik Testi: Medikal görüntüye tekli ve çoklu atak olmak üzere iki atak uygulanarak oynanmış görüntü elde edilmiştir. Şekil 3.39.(b)'de tekli atağa örnek gösterilmiştir. Şekil 3.39. (a)'da verilen mamografi görüntüsündeki bir tümör bölgesi yine aynı görüntüdeki bir başka bölge ile kapatılarak Şekil 3.39.(b)'deki oynanmış medikal görüntü elde edilmiştir. Oynanmış görüntü oluşturma aşamasında kapama işlemi sonucunda oluşan dış kenar izlerini gizlemek için ise yapıştırmadan sonra dış kenarlara sınır yumuşatma işlemi uygulanmıştır. Üç senaryo için oynanmış bölgelerdeki eşleşen anahtar nokta sayısı Şekil 3.39.(c)-3.39.(h)'da görüldüğü üzere sırasıyla 19, 26 ve 43 elde edilirken, tespit edilen anahtar nokta sayısı sırasıyla 886, 816 ve 2059 dur. Şekil 3.39.(h)'dan görüldüğü üzere üçüncü senaryo sonucunda daha fazla eşleşen nokta elde edilmiştir.

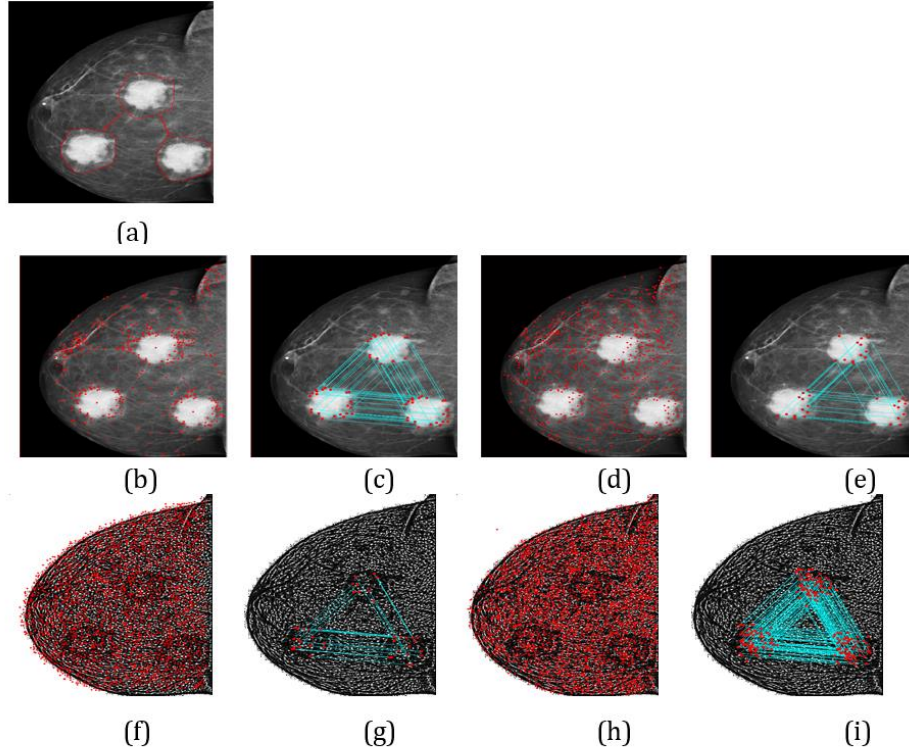


Şekil 3.39. (a) Medikal görüntü (b) Oynanmış medikal görüntü (c) (d) Senaryo 1 sonucu: SIFT tabanlı yöntem (Anahtar nokta: 886 Eşleşen Anahtar nokta: 19) (e) (f) Senaryo 2 sonucu: SURF tabanlı yöntem (Anahtar nokta: 816 Eşleşen anahtar nokta: 26) (g) (h) Senaryo 3 sonucu: LBPROT+SURF tabanlı yöntem (Anahtar nokta: 2059 Eşleşen anahtar nokta: 43) (i) (i) Önerilen yöntem sonucu (Anahtar nokta: 3791 Eşleşen anahtar nokta: 121)

Önerilen yöntem sonuçlarına bakıldığında ise oynanmış medikal görüntü üzerinde toplam 3791 nokta elde edilmiş ve bu noktaların Şekil 3.39.(i) ve Şekil 3.39.(j)'de görüldüğü üzere 121'i eşleşmiştir. Sonuçlardan da görüldüğü üzere önerilen yöntemde diğer senaryolara göre daha fazla eşleşen nokta elde edilmiştir.

Bir diğer atak olarak tıbbi görüntü üzerinde çoklu atak uygulanmış birden fazla sahte bölge oluşturulmuştur. Mamografi görüntüsünde Şekil 3.40.(a)'da görüldüğü üzere tümör bölgesi kopyalanarak aynı görüntü içerisinde farklı iki bölgeye yapıştırılmıştır. Bu görüntü için üç senaryo sonucunda elde edilen eşleşen anahtar nokta sayıları ve çıkarılan toplam anahtar nokta sayısı sırasıyla Şekil 3.40.(b)-3.40.(g)'de verildiği gibi (14, 16, 3) ve (470, 769, 2667)'dir. Şekil 3.40.(h)-(i)'de önerilen yöntemin sonuçlarına bakıldığında bu görüntü üzerinde toplam 4868 anahtar nokta çıkarmış ve bunların 57'sini eşleştirmiştir. Ayrıca

çoklu atak yapılsa dahi önerilen yöntemde anahtar nokta çıkarımından önce doku bilgisi çıkarımı yapıldığı için diğer senaryolara göre eşleşen anahtar nokta sayısı arttırılmıştır.

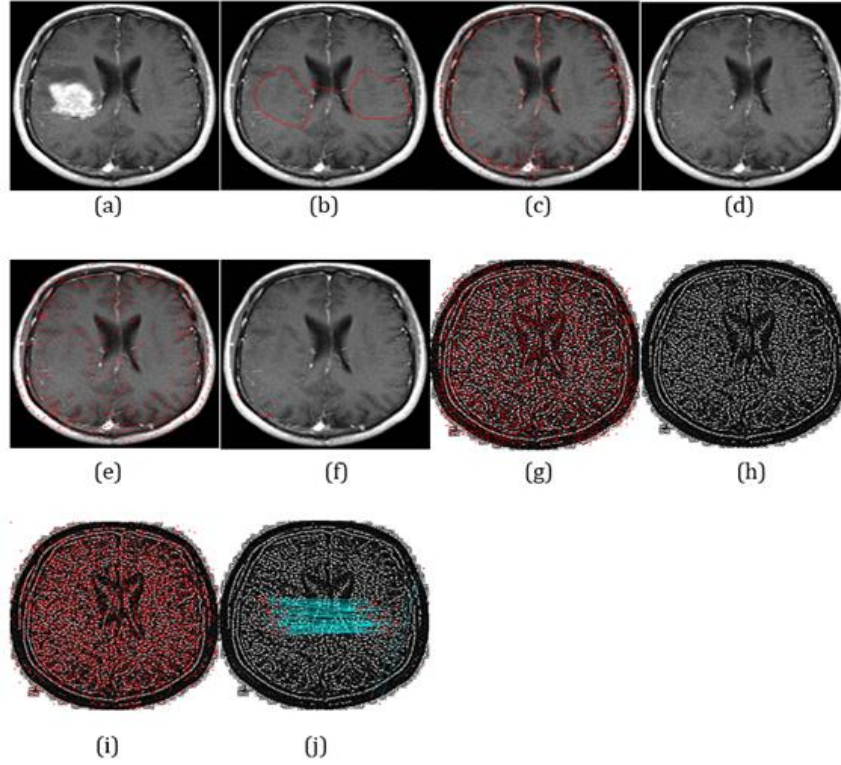


Şekil 3.40.(a) Oynanmış medikal görüntü (b) (c) Senaryo 1 sonucu: SIFT tabanlı yöntem(Anahtar nokta: 470 Eşleşen anahtar nokta: 14) (d) (e) Senaryo 2 sonucu: SURF tabanlı (Anahtar nokta: 769 Eşleşen anahtar nokta: 16) (f) (g) Senaryo 3 sonucu: LBPROT+SURF tabanlı (Anahtar nokta: 2667 Eşleşen anahtar nokta: 3) (h) (i) Önerilen Yöntem (Anahtar nokta: 4868 Eşleşen anahtar nokta: 57)

Sağlamlık Testi: Bu testte önerilen yöntemin dönme, ölçekleme, bulanıklaştırma ve gürültü ekleme ataklarına karşı dayanıklılığı test edilmiştir. Bu ataklar karşısında önerilen yöntem ve üç senaryodan elde edilen sonuçlar bu bölümde sırasıyla verilecektir.

Şekil 3.41.(a)'da verilen beyin MR test görüntüsü döndürme atağının uygulanması için kullanılmıştır. Tümör bölgesi Şekil 3.41.(b)'de görüldüğü üzere yine aynı görüntü içerisindeki bir başka bölge alınarak kapatılmıştır. Bununla birlikte bölge yapıştırılmadan önce saat yönünde 30 derece döndürülmüş ve etrafında çevresi ile uyumsuzluk oluşturmasını diye sınır yumuşatma uygulanmıştır. Bu görüntü için üç senaryonun eşleşen anahtar nokta sayısı ve toplam çıkarılan anahtar nokta sayıları sırasıyla (0, 4, 0) ve (531, 691, 2197)'dir. Önerilen yöntemde bu görüntü üzerinde 111 eşleşmiş nokta bulunurken diğer yöntemlerde 4 eşleşmiş noktadan fazla bulunamamıştır. Şekil 3.41.(c)-3.41.(j)'deki

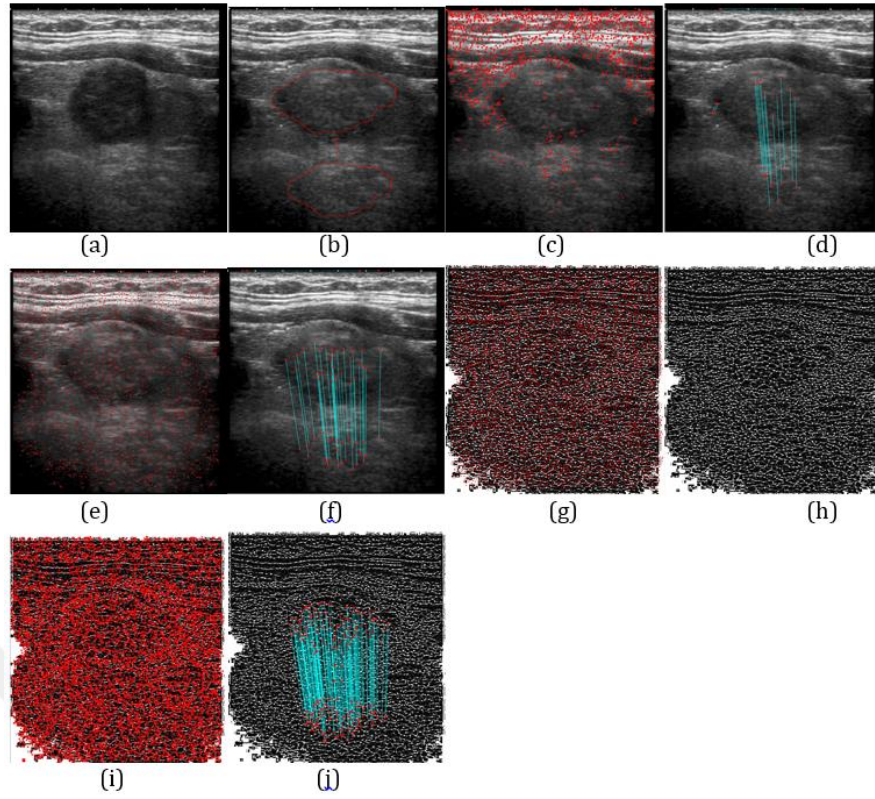
görsel sonuçlara bakıldığında sahte bölgeler üzerinde bulunan anahtar nokta yoğunlukları gösterilmiştir.



Şekil 3.41. (a) Medikal görüntü (b) Oynanmış medikal görüntü(c) (d) Senaryo 1 sonucu: SIFT tabanlı yöntem (Anahtar nokta: 531 Eşleşen anahtar nokta: 0) (e) (f) Senaryo 2 sonucu: SURF tabanlı (Anahtar nokta: 691 Eşleşen anahtar nokta: 4)(g) (h) Senaryo 3 sonucu: LBPROT+SURF tabanlı (Anahtar nokta: 2197 Eşleşen anahtar nokta: 0)(i) (j) Önerilen yöntem (Anahtar nokta: 3958 Eşleşen anahtar nokta: 111)

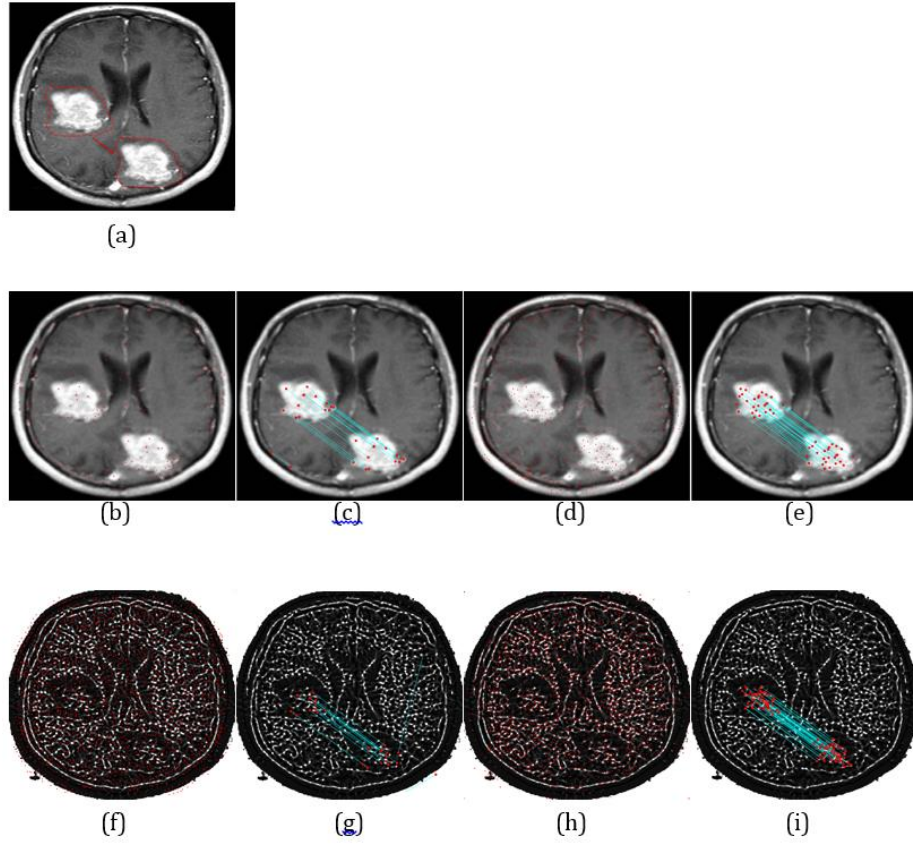
Önerilen yöntemin en büyük avantajı bu görsel örnek sonucunda görülmektedir. Sahtecilik işlemi olarak görüntüdeki bir bölge düz bir bölge ile kapadığı zaman diğer yöntemler sahte bölge üzerinde hiç anahtar nokta bulamamakta ya da sahte bölgeler dışındaki alanda birkaç anahtar nokta bulabilmektedir.

Bir başka test olarak kopyalanan bölge yapıştırılmadan önce ölçeklenecektir. Şekil 3.42.(a)'da verilen ultrason görüntüsünün bir bölümü 120% oranında ölçeklenerek aynı görüntü içerisindeki kistik bölgeyi kapamak için kullanılmıştır.



Şekil 3.42. (a) Medikal görüntü (b) Oynanmış medikal görüntü (c)(d) Senaryo 1 sonucu: SIT tabanlı yöntem (Anahtar nokta: 1559 Eşleşen anahtar nokta: 32) (e)(f) Senaryo 2 sonucu: SURF tabanlı (Anahtar nokta: 2325 Eşleşen anahtar nokta: 59) (g)(h) Senaryo 3 (Anahtar nokta: 5967 Eşleşen anahtar nokta: 2) (i) (j) Önerilen yöntem (Anahtar nokta: 10668 Eşleşen anahtar nokta: 260)

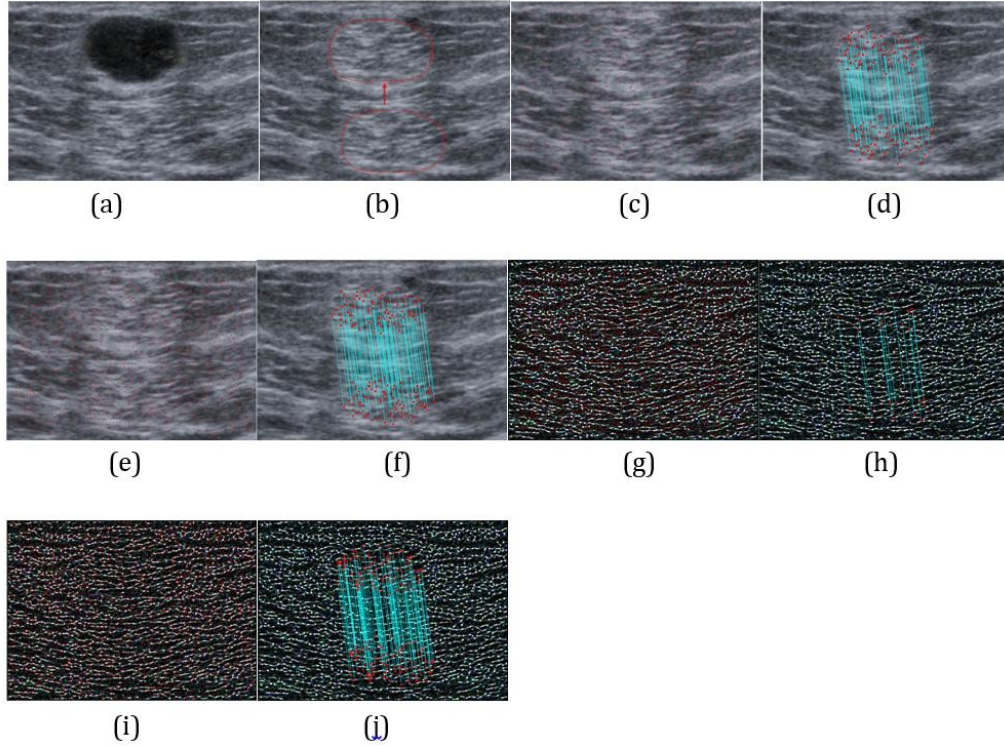
Şekil 3.42.(b)'deki kırmızı çizgiler sahte bölgeleri göstermektedir. Şekil 3.42.(c), 3.42.(e), 3.42.(g)'den görüldüğü üzere üç senaryo sonucunda sırasıyla 1559, 2325 ve 5967 anahtar nokta elde edilmiş ve Şekil 3.42.(d), 3.42.(f) ve 3.42.(h)'de ise bu anahtar noktaların 32, 59 ve 2'si eşleşmiştir. Önerilen yöntem sonucunda ise bu görüntü üzerinde 10668 anahtar nokta elde edilmiş ve bu noktaların Şekil 3.42.(i) ve 3.42.(j)'de görüldüğü üzere 260'ı eşleşmiştir. SURF öncesinde LBPROT operatörü kullanılan yöntemde 5967 nokta bulmasına rağmen bu test sonucunda başarısız olmuş çünkü sadece bir çiftini noktaların eşleştirebilmiş, bu çiftte sahte bölge dışındaki bir eşleşmedir. İkinci senaryo olan sadece SURF kullanılan yöntem ise diğer iki senaryoya göre daha iyi sonuç verirken, önerilen yöntemde bu yöntemin dört katı anahtar nokta eşleşmesi elde edilmiştir.



Şekil 3.43. (a) Oynanmış medikal görüntü (b) (c) Senaryo 1 sonucu: SIFT tabanlı yöntem (Anahtar nokta: 267 Eşleşen anahtar nokta: 27) (d) (e) Senaryo 2 sonucu: SURF tabanlı (Anahtar nokta: 583 Eşleşen anahtar nokta: 50) (f) (g) Senaryo 3 sonucu: LBPROT+SURF tabanlı (Anahtar nokta: 1864 Eşleşen anahtar nokta: 26) (h) (i) Önerilen yöntem (Anahtar nokta: 2485 Eşleşen anahtar nokta: 84)

Önerilen yöntemin bulanıklaştırmaya olan dayanıklılığını ölçmek için ise oynanmış medikal görüntülere bulanıklaştırma uygulanmıştır. Şekil 3.43.(a) oynanmış medikal görüntünün $w=5$ ve $\sigma=3$ ile birlikte Gauss bulanıklaştırma uygulanmış halini göstermektedir. Üç senaryonun bulanık medikal görüntüde elde ettiği toplam anahtar nokta sayıları sırasıyla 267, 583 ve 1864 olarak sırasıyla Şekil 3.43.(b), 3.43.(d) ve 3.43.(f)'de verilirken eşleşen anahtar nokta sayıları sırasıyla 3.43.(c), 3.43.(e) ve 3.43.(g)'de sırasıyla 27, 50 ve 26 olarak verilmiştir. Önerilen yöntem ise bu bulanıklaştırılmış medikal görüntüden 2485 anahtar nokta çıkarırken Şekil 3.43.(i)'de görüldüğü gibi sahte bölge üzerinde bunların 84'ünü eşleştirmiştir. Eşleşen anahtar nokta sayılarına göre kıyaslama yapıldığında sadece SURF kullanan ikinci senaryo diğer senaryolara göre daha iyi sonuç vermiştir. Ancak önerilen yöntem bu senaryoların hepsinden daha iyi sonuç vermiştir.

Son test olarak da yöntemin gürültüye olan dayanıklılığını ölçmek için oynanmış medikal görüntüye beyaz Gauss gürültüsü eklenmiştir. Şekil 3.44.(a)'daki orijinal medikal görüntü değiştirilerek değiştirilme izlerini kapatmak için oynanmış görüntüye 30 dB'lik gürültü eklenerek Şekil 3.44.(b) elde edilmiştir.

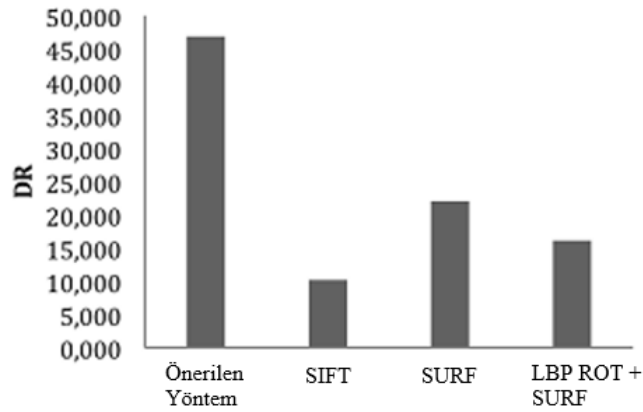


Şekil 3.44.(a)Medikal görüntü (b) Oynanmış medikal görüntü (c) (d) Senaryo 1 sonucu: SIFT tabanlı yöntem (Anahtar nokta: 784 Eşleşen anahtar nokta: 130) (e) (f) Senaryo 2 sonucu: SURF tabanlı (Anahtar nokta: 1469 Eşleşen anahtar nokta: 166) (g) (h) Senaryo 3 sonucu: LBPROT+SURF tabanlı (Anahtar nokta: 3043 Eşleşen anahtar nokta: 20) (i) (j) Önerilen yöntem (Anahtar nokta: 4730 Eşleşen anahtar nokta: 194)

Önerilen yöntem gürültü eklenmiş görüntü için 4730 anahtar nokta çıkarmış Şekil 3.44.(i) ve 3.44.(j)'de görüldüğü üzere bu noktaların 194'ünü eşleştirmiştir. Şekil 3.44.(d), 3.44.(f) ve 3.44.(h)'deki görsel sonuçlardan görüldüğü üzere daha az anahtar nokta bulmuş üçüncü senaryo olan LBPROT ile SURF kullanan yöntem eşleşen anahtar nokta sayısı açısından en kötü sonucu vermiştir. Bu senaryolar arasından ikinci senaryo en iyi sonucu vermekle beraber önerilen yöntem bu yöntem sahte bölgeler üzerinde 28 fazla anahtar nokta bulmuştur.

Sonuç olarak üç senaryoyu sağlık açısından değerlendirdiğimizde en kötü sonucu üçüncü senaryo verirken en iyi sonucu sadece SURF’u kullanan ikinci senaryo vermiştir. Bununla birlikte tüm testlerde özellikle dönme ve ölçekleme ataklarında, sahte bölgeler üzerinde diğer üç senaryo ile karşılaştırıldığında önerilen yöntem daha fazla eşleşen anahtar nokta sayısı bulmuştur. Bu sonuçlardan hareketle LBPROT ile doku bilgisi çıkarımı ile birlikte SIFT ile anahtar nokta çıkarım ve eşleştirmesi medikal görüntülerde oynanmış bölge tespitini iyileştirmektedir.

Karşılaştırma Testleri: Bu testte önerilen yöntem bir önceki bölümde verilen üç senaryo ile karşılaştırılacaktır. Testlerde değiştirilmiş medikal görüntü veri tabanı kullanılarak önerilen yöntem ve üç senaryo için ortalama tespit oranları DR hesaplanmıştır. Elde edilen yüksek DR’ler sahte bölge üzerinde daha fazla eşleşmiş noktaya karşılık gelerek tespitin güvenilirliğini ortaya koymaktadır.

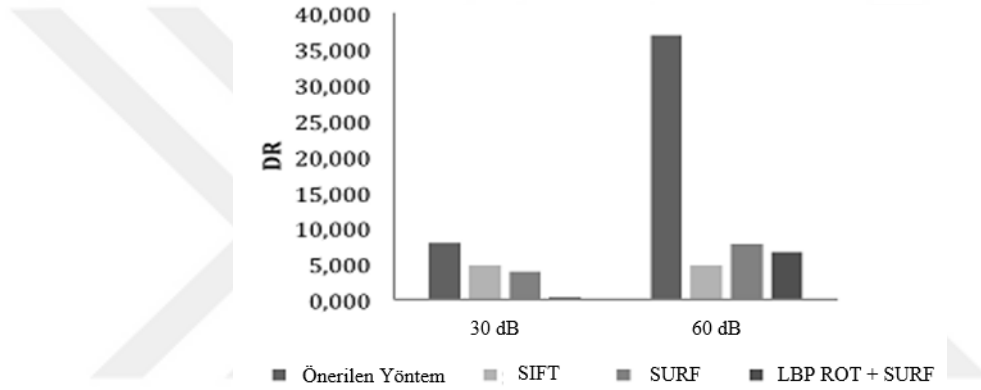


Şekil 3.45. Ataksız tıbbi görüntüler için ortalama DR değerleri

İlk testte ataksız oynanmış medikal görüntüler alınarak önerilen yöntem ve diğerlerinin performansına bakılmıştır. Yöntemlerin 58 test görüntüsü için ortalama DR değerleri hesaplanmış ve Şekil 3.45.’te gösterilmiştir. Önerilen yöntemden 46.8 DR, SURF tabanlı, LBPROT + SURF ve SIFT tabanlı senaryolardan ise sırasıyla yaklaşık olarak 22.06, 16.03 ve 10.11 DR elde edilmiştir. Sonuçlardan görüldüğü üzere önerilen yöntem diğer senaryolarla kıyaslandığında en yüksek DR değerine sahip olarak sahte bölgeler üzerinde en fazla anahtar nokta tespit etmiştir.

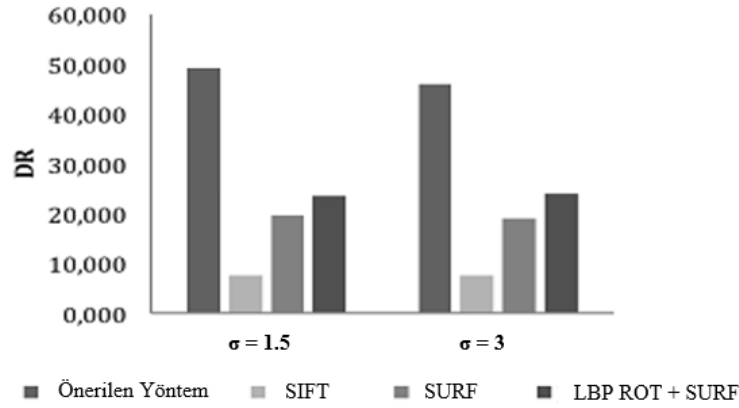
Bir başka test olarak da atağa uğramış ve sahte bölgeleri döndürülmüş ve ölçeklenmiş medikal görüntüler alınarak önerilen yöntem ve diğer senaryoların performans kıyaslaması yapılmıştır. Elde edilen sonuçlar aşağıda verilecektir.

İlk olarak veri tabanındaki oynanmış medikal görüntülere AWGN uygulanmıştır. Oynanmış medikal görüntülerde değiştirme izlerini gizlemek için görüntülere 30 dB ve 60 dB’lik gürültü eklenmiştir. Şekil 3.46.’da önerilen yöntem ve diğer senaryoların ortalama DR değerleri verilmiştir. Önerilen yöntem bar grafiğinde görüldüğü üzere en yüksek DR değerine sahiptir. Değiştirilmiş görüntüye 30 dB gürültü eklense dahi önerilen yöntemin ortalama DR değeri yaklaşık olarak 7.8’dir. SIFT tabanlı yöntem önerilen yöntemden sonra 4.8 DR değeri ile en iyi sonuç vermektedir. Bununla birlikte 60 dB’lik gürültü eklendiğinde 7.7 DR değeri ile SURF tabanlı yöntem önerilen yöntemin 36.86 DR değerinden sonra en iyi ikinci değeri vermektedir.



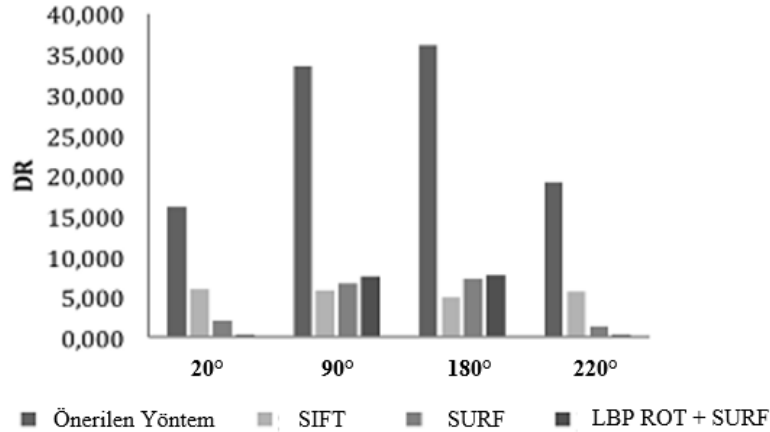
Şekil 3.46. Gürültü eklenmiş medikal görüntüler için ortalama DR değerleri

Oynanmış medikal görüntülerde değiştirilmiş bölge sınırlarındaki izleri kaldırmak için bir başka atak olarak bulanıklaştırma işlemi uygulanmıştır. Oynanmış medikal görüntüler pencere boyutu 5 ve standart sapma $\sigma=1.5$ ve 3 olacak şekilde bulanıklaştırılmıştır. Önerilen yöntem diğer üç senaryo ile karşılaştırıldığında en yüksek DR değerine sahiptir. Üçüncü senaryo $\sigma=1.5$ ve 3 standart sapmalarında DR’si 23 iken, önerilen yöntemin Şekil 3.47.’de görüldüğü üzere bu standart sapmalarda aynı bulanık görüntüler için elde edilen DR değerleri sırasıyla 49.29 ve 46.05’dur. Diğer iki senaryo (SIFT tabanlı ve SURF tabanlı) sırasıyla yaklaşık 7 ve 19 değeri vererek önerilen yöntemle kıyaslandığında en kötü sonucu vermişlerdir.



Şekil 3.47. Bulanıklaştırma uygulanmış medikal görüntüler için ortalama DR değerleri

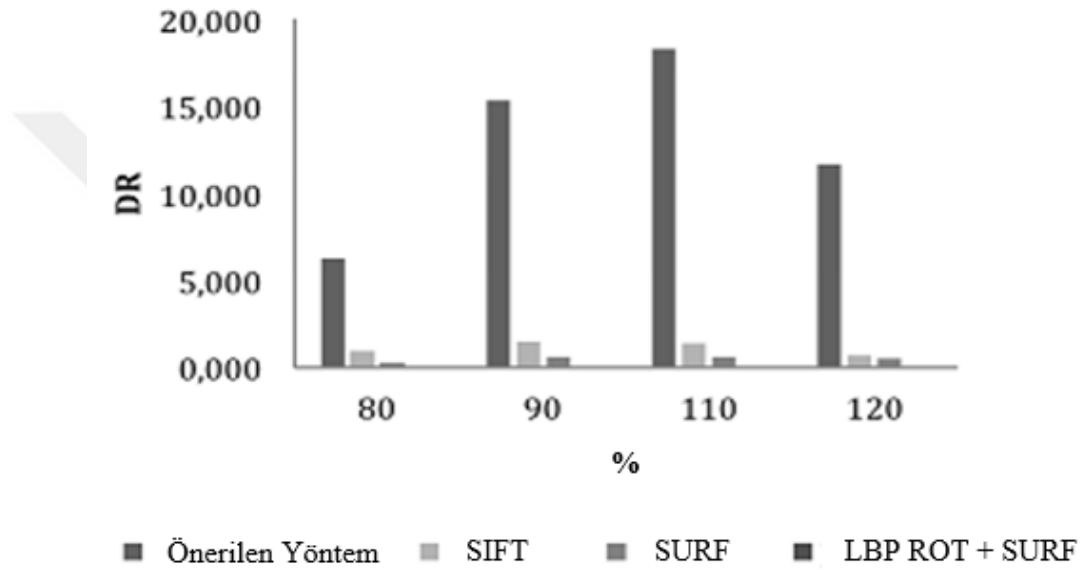
Bir başka test olarak 48 oynanmış tıbbi görüntü oluşturulurken sahte bölgeler yapıştırılmadan önce döndürülmüştür. Şekil 3.48’de 20° , 90° , 180° ve 220° döndürme sonucundaki ortalama DR değerlerini göstermektedir. Yöntemler arasında SURF tabanlı iki yöntem 90° ve 180° döndürme sonucunda SIFT tabanlı yöntemle göre daha iyi sonuç vermektedir. Bununla birlikte SIFT tabanlı yöntem yani birinci senaryo 20° ve 220° döndürme sonucunda SURF tabanlılara göre daha başarılıdır.



Şekil 3.48. Döndürme sonucunda medikal görüntüler için elde edilen DR değerleri

Önerilen yöntem ise tüm döndürme sonuçlarında diğer yöntemlere göre daha yüksek DR değerine sahiptir. 20° , 90° , 180° ve 220° döndürme sonucunda önerilen yöntemin DR değerleri sırasıyla 16.06, 33.48, 36.15 ve 19.07’dir. 90° ve 180° sonucunda elde edilen DR değerleri 20° ve 220° sonucundan daha yüksektir.

Son olarak önerilen yöntemin ölçeklemeye karşı dayanıklılığını test etmek için 59 test görüntüsünün oluşturulması aşamasında kopyalanan bölge yapıştırılmadan önce yüzde 80, 90, 110 ve 120 oranında ölçeklenmiştir. Şekil 3.49.'da bu 59 test görüntüsü için yöntemlerden elde edilen ortalama DR değerlerini göstermektedir. Önerilen yöntem %80 ölçekleme sonucunda 6.2 DR değerine sahip iken diğer yöntemler yaklaşık olarak sırasıyla 0.9, 0.2 ve 0 DR'dir. Şekilden de görüldüğü üzere tüm ölçeklemelerde önerilen yöntemden çok daha yüksek DR'ler elde edilmiştir.



Şekil 3.49. Ölçekleme sonucunda medikal görüntüler için elde edilen DR değerleri

Sahtecilik izlerini gizlemek adına yapılan çeşitli ataklar sonucunda önerilen yöntem ve üç senaryonun ortalama DR değerleri verilmiştir. Sonuçlardan da görüldüğü üzere önerilen yöntem diğer yöntemlere göre daha yüksek DR değeri vermiştir.

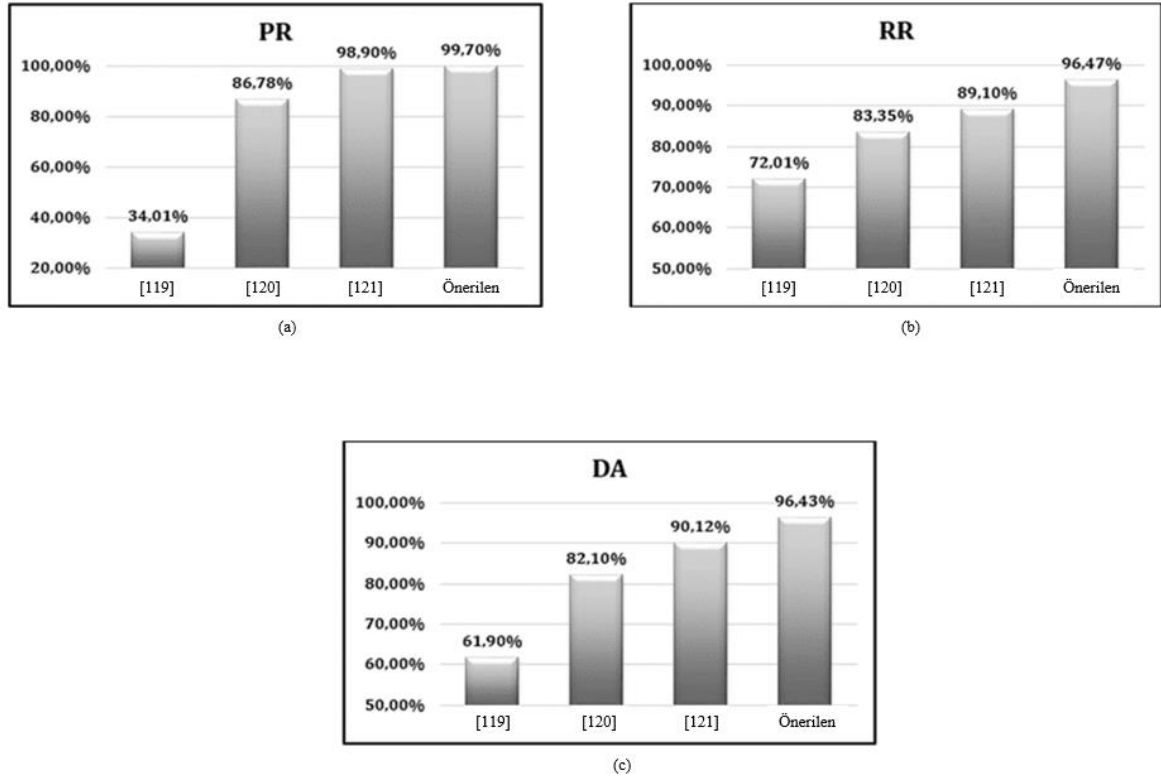
Çalışmada oynanmış medikal görüntülerin tespiti için yeni bir pasif görüntü doğrulama yöntemi önerilmiştir. Bu şekilde aktif doğrulama yöntemlerindeki görüntü alındıktan sonra bir damga gömme ihtiyacı ortadan kaldırılmıştır.

3.10. LDB'ye Dayalı Çerçeve Tekrarlama Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar

Çalışmada literatürde video sahteciliklerinin tespiti alanında ilk kez LDB kullanan yeni bir Çerçeve Tekrarlama sahteciliği tespit yöntemi önerilmiştir. Yöntemde sahteciliğin

uzaklığını tahmin etmek ve tekrarlanmış çerçevelerin tam yerini belirlemek için benzer özellik vektörlerine sahip benzer çerçeveler arasındaki uzaklık kullanılmaktadır. Benzer çerçeveler arasındaki PSNR, bu uzaklıkları üç sınıfa ayırtırmak için kullanılmakta ve kural tabanlı mekanizma üye olduğu sınıfa göre sahte çerçeveleri belirlemektedir. Önerilen yöntem diğer çalışmalara göre daha düşük çalışma zamanına ve daha yüksek doğruluk oranına sahiptir. Yönteme ilişkin detaylar yapılan çalışmalar kısmında verilmektedir. Önerilen yöntemin etkinliğini gösterebilmek ve literatürdeki benzer yöntemlerle kıyaslamasını gerçekleştirebilmek amacıyla yapılan testler ve elde edilen deneysel sonuçlar aşağıda verilmektedir.

Yapılan testlerde kullanılan eşik değerleri $t=5$, $t_h=2$, $w=10$, $t_1=30$, $t_2=34$, $t_3=37$ olarak alınmıştır.



Şekil 3.50. Önerilen yöntemin diğer çalışmalarla (a) PR (b) RR (c) DA değerlerine göre karşılaştırılması

Önerilen yöntemin etkinliğini gösterme amacıyla yöntem literatürdeki benzer çalışmaları [119-121] kıyaslanmıştır. Şekil 3.50.(a) ve 3.50.(b) önerilen yöntemin sırasıyla ortalama PR ve RR değerlerini göstermektedir. Şekilden de görüldüğü üzere yöntem benzer çalışmalara [119-121] göre daha yüksek PR ve RR değerlerine sahiptir.

Şekil 3.50.(c)'de verilen DA değerlerine bakıldığında ise önerilen yöntem %96.43 gibi bir değere sahipken benzer çalışmaların DA değerleri bu değerin altında kalmaktadır.

Önerilen yöntem literatürdeki diğer yöntemlerle algoritmanın koşum zamanı açısından da kıyaslanmıştır. Tablo 3.9.'da önerilen yöntem ve benzer çalışmaların çerçeve başına düşen çalışma zamanı saniye cinsinden verilmiştir. Tablodan da görüldüğü üzere önerilen yöntem Çerçeve Tekrarlama sahteciliği tespitini diğer çalışmalara göre daha hızlı gerçekleştirmektedir.

Tablo 3.9. Çalışma zamanlarının kıyaslanması

Yöntem	Çalışma zamanı(s/çerçeve)
Wang vd. [119]	294.67
Lin vd.[120]	140.6
Yang vd. [121]	0,38
Önerilen Yöntem	0,02

Elde edilen deneysel sonuçlardan da görüldüğü üzere önerilen yöntem Çerçeve Tekrarlama Sahteciliğinin tespitini diğer yöntemlere göre daha yüksek doğruluk ve daha düşük çalışma zamanı ile tespit etmektedir.

3.11. İkili Özelliklere Dayalı Çerçeve Yansıma\Tekrarlama Sahteciliği Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar

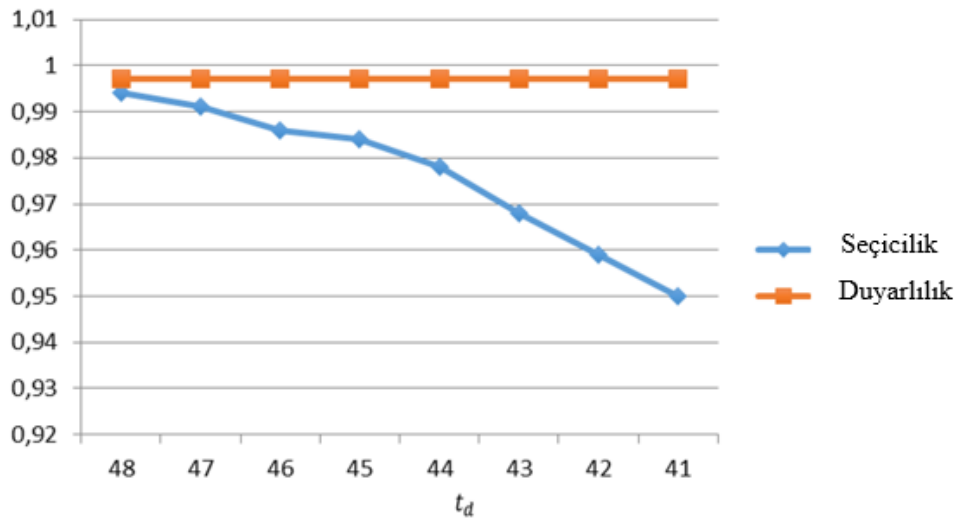
Çerçeve Tekrarlama sahteciliği yapılırken tekrarı yapılacak olan çerçeve grubu, eklenmeden önce eğer ayna yansıması alınarak eklenmiş ise literatürde Çerçeve Tekrarlama sahteciliği tespiti için önerilen yöntemler benzer algoritmik yapıya sahip oldukları için bu sahteciliği tespit edememektedir. Önerilen çalışmada yeni bir Çerçeve Tekrarlama/Yansıma sahteciliği tespit yöntemi geliştirilmiştir. Yöntemde çerçevelerden ikili özellikler çıkarılmakta ve bu özellikler arasındaki benzerlik belirlenmektedir. PSNR kullanılarak çok sayıda aday elimine edilip, sahte çerçevelerin tespiti iyileştirilmektedir. Önerilen yöntem benzer çalışmalarla kıyaslandığında daha iyi çalışma zamanına ve tespit doğruluğuna sahiptir. Yönteme ilişkin detaylar yapılan çalışmalar kısmında verilmektedir. Önerilen yöntemin etkinliğini gösterebilmek ve literatürdeki benzer yöntemlerle kıyaslamasını gerçekleştirebilmek amacıyla yapılan testler ve elde edilen deneysel sonuçlar aşağıda verilmektedir.

Algoritmada test aşamasında $t_d=48$, $t_h=10$ ve $e=2$ olarak alınmıştır. Deneyler esnasında t_d değeri 48 olarak seçilmiştir. Yöntemde veri tabanındaki videolar [41-48] aralığındaki t_d değerleriyle test edilmiş ve Şekil 3.51.'de görüldüğü gibi Duyarlılık ve Seçicilik eğrileri oluşturulmuştur. Eğrilerden de görüldüğü üzere eşik değeri 48 seçildiğinde FP ve FN değerleri en iyi değerini almaktadır.

Yöntemin performansını gösterebilmek için bu testte 30 sahte video içeren Çerçeve Yansıma sahteciliği için oluşturulmuş veri tabanı kullanılmıştır.

İlk deney olarak da önerilen yöntemin PR, RR ve DA değerleri hesaplanmıştır. Şekil 3.52.(a), 3.52.(b) ve 3.52.(c) üç ayrı sahte test videosu kümesi için önerilen yöntemin PR, RR ve DA değerlerini göstermektedir.

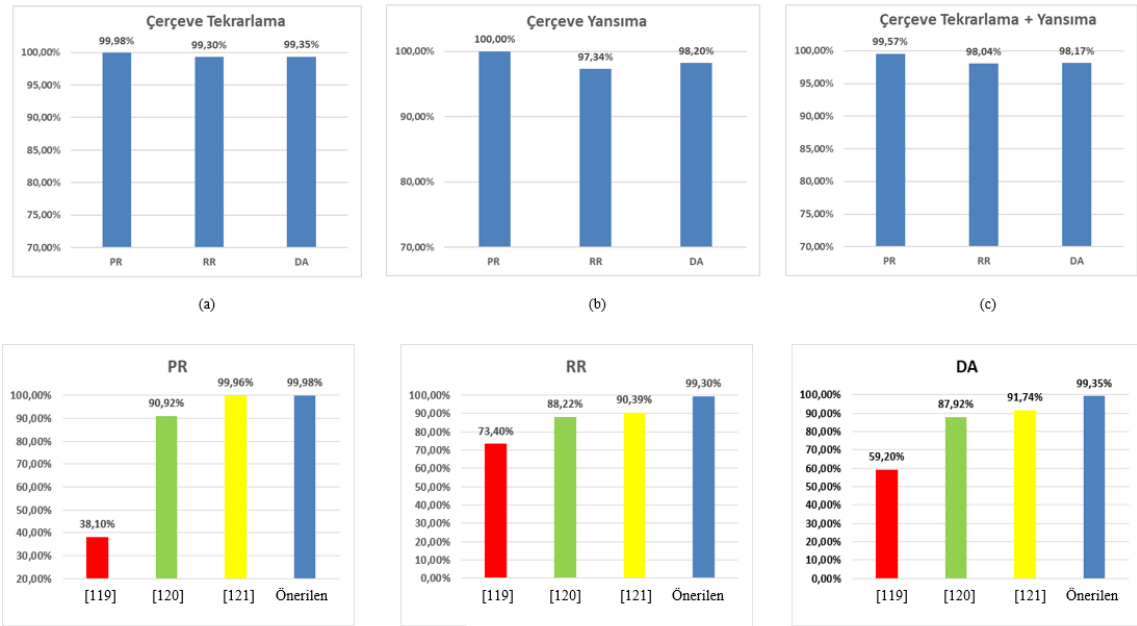
Birinci sahte video kümesi Tablo 3.2.'deki ilk on videoyu içermekle beraber bu videolar için PR, RR ve DA sonucu Şekil 3.52.(a)'da verilmektedir. Şekil 3.52.(b) ve 3.52.(c)'de ise sırasıyla önerilen yöntemin ikinci test video kümesi olan (video 11 ile video 20 arasındaki videolar) sade çerçeve yansıma ve üçüncü test video kümesi olan (video 21 ile video 30 arasındaki videolar) hem çerçeve tekrarlama hem de çerçeve yansıma atakları yapılmış test videoları için PR, RR ve DA değerlerini göstermektedir.



Şekil 3.51. Duyarlılık ve Seçicilik eğrilerine t_d değerinin etkisi

Sonuçlardan da görüldüğü üzere Şekil 3.52'de gösterildiği gibi videolarda hem çerçeve tekrarlama hem de çerçeve yansıma atağı birlikte yapılmış olsa dahi önerilen yöntem sahte çerçeve grubunu yüksek PR, RR ve DA değerleriyle tespit edebilmektedir.

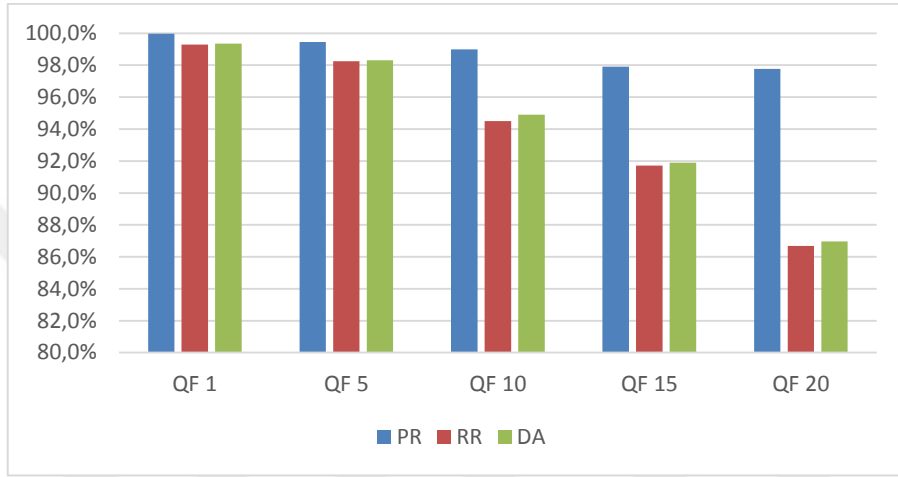
Bir başka deney olarak da önerilen yöntemin PR, RR ve DA değerleri literatürdeki diğer çalışmalarla kıyaslanmıştır. Bu deney sonuçları Şekil 3.52.(d), 3.52.(e) ve 3.52.(f)'de verilmiştir. Bununla birlikte literatürdeki çalışmalar [119-121] çerçeve yansıma sahteciliğini tespit edemeyeceği için deneyde kullanılan test videoları çerçeve yansıma atağı içermeyen videolardan alınmıştır. Şekil 3.52.(e) yöntemin RR değerinin [121] çalışmasından yaklaşık olarak %9 daha yüksek olduğunu göstermektedir. [119, 120] çalışmalarının RR değerleri ise sırasıyla %73 ve %88'dir. DA değerine bakıldığında ise Şekil 3.52(f) 'de görüldüğü üzere önerilen yöntemden [121] çalışmasına göre daha iyi bir sonuç elde edilmiştir. Yaklaşık olarak önerilen yöntemin DA'sı ile [121] çalışmasının DA'sı arasındaki fark %8'dir. Şekil 3.52.(d) ise [121] ile önerilen yöntemden elde edilen PR değerlerinin neredeyse aynı olduğunu göstermektedir.



Şekil 3.52. (a)(b)(c) Çerçeve Tekrarlama, Çerçeve Yansıma ve hem Çerçeve Yansıma hemde Çerçeve Tekrarlama sahteciliklerinde yöntemin performans değerlendirmesi(d)(e)(f) Yöntemin diğer çalışmalarla kıyaslanması

Diğer bir deneyde de iki kere sıkıştırma sonucunda önerilen yöntemin etkinliği test edilmiştir. Bunun için Tablo 3.2'de verilen sahte videolar sahtecilik işleminden sonra farklı kalite faktörleriyle tekrar sıkıştırılmıştır. Sahte videoları MPEG-4 enkoder ile kodlamak için QF değerleri 1 (neredeyse kayıpsız), 5, 10, 15 ve 20 (önemli ölçüde sıkıştırılmış) olarak alınmıştır. Şekil 3.53. farklı kalite faktörlerinin sıkıştırılmış test

videoları için önerilen yöntemin DA, PR ve RR değerlerini göstermektedir. Şekilden de görüldüğü üzere QF=1 iken önerilen yöntemin DA değeri yaklaşık %99.3 iken kalite faktörü QF=20 olduğunda bu değer %86.97 değerine gerilemektedir. Yine PR ve RR değerleri sırasıyla QF=1’de %99.98 ve %99.3 olarak elde edilirken QF=20 olduğunda %97.76 ve %86.69 ‘a düşmektedir. Bu deneyden görüldüğü üzere sıkıştırma oranı arttıkça DA ve RR değerleri azalmakta buna karşın PR değeri fazla etkilenmemektedir.



Şekil 3.53. Sahtecilik işlemi sonrasında uygulanan yeniden sıkıştırma işlemi sonucunda önerilen yöntemin tespit sonuçları

Son deney olarak ise önerilen yöntem çeşitli ataklara dayanıklılık açısından ve çalışma zamanı, PR, RR ve DA değerlerine göre diğer çalışmalarla kıyaslanmıştır. Tablo 3.10’den da görüldüğü üzere önerilen diğer yöntemlere göre sahtecilik tespitini daha hızlı gerçekleştirmektedir. Önerilen yöntemin tek bir çerçeveyi işleme zamanı 0.01 s iken, diğer yöntemler [119-121] yaklaşık olarak sırasıyla 294 s, 140 s ve 0.58 s’de tek bir çerçeveyi işlemektedir. Önerilen yöntemin diğer yöntemlere göre hızındaki bu fark karmaşık olmayan bir algoritmaya sahip olmasından ileri gelmektedir. Aynı zamanda önerilen yöntem sahtecilik tespitini daha iyi DA, PR ve RR değerleriyle gerçekleştirmektedir. Tablo 3.10’den da görüldüğü üzere DA değeri önerilen yöntemde 0.99 iken diğer yöntemler de sırasıyla 0.59, 0.87 ve 0.91 olarak elde edilmiştir. Yine önerilen yöntemin RR değeri diğer yöntemlerden daha yüksek iken PR değeri [121] çalışması ile yaklaşık aynı olarak elde edilmiştir. Tablodaki sonuçların tamamına bakıldığında önerilen yöntemin daha iyi tespit performansına sahip olduğu görülmektedir.

Yöntem Çerçeve Yansıma sahteciliğini tespit edebilmesi yanında Çerçeve ekleme (FI), Çerçeve tekrarlama (FD) ve bu sahteciliklerin karışımı da test edebilmektedir. FI sahteciliğinde videodaki bir çerçeve kopyalanıp yine aynı videoda başka bir yere eklenmektedir. Bu atakların karışımı FM+FD ve FM+FI şeklinde olabilmektedir. Örneğin sahte video içerisindeki sahte video gruplarının bir kısmı çerçeve yansıması alınarak tekrarlanmış bir kısmı ise sadece tekrarlanmış olabilmektedir (FM+FD). Tablo 3.10.'dan da görüldüğü üzere önerilen yöntem bu atakları tespit edebilirken diğer yöntemler tespit edememektedir.

Tablo 3.10. Önerilen yöntemin diğer çalışmalarla kıyaslanması

Yöntem	FM*	FD*	FI*	FM+FD	FM+FI	Çalışma zamanı(s/çerçeve)	DA	PR	RR
Wang vd. [119]	×	✓	✓	×	×	294.67	0.59	0.38	0.73
Lin vd. [120]	×	✓	✓	×	×	140.6	0.87	0.9	0.88
Yang vd. [121]	×	✓	✓	×	×	0.58	0.91	0.99	0.90
Önerilen yöntem	✓	✓	✓	✓	✓	0.01	0.99	0.99	0.99

* FM: Çerçeve Yansıma, FD: Çerçeve Tekrarlama, FI: Çerçeve Ekleme

Bu bölümde verilen deneysel sonuçlardan da görüldüğü üzere literatürdeki benzer çalışmalar Çerçeve Yansıma sahteciliğini tespit edemezken önerilen yöntem Çerçeve Tekrarlama ve Çerçeve Yansıma sahteciliklerini tespit edebilmektedir. Diğer yöntemlerle kıyaslandığında önerilen yöntem daha iyi PR, RR ve DA değerlerine sahip olup, daha kısa süreli çalışma zamanına sahiptir.

3.12. BoW Modeline Dayalı Çerçeve Tekrarlama Sahteciliğinin Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar

Çalışmada BoW modeline dayalı yeni bir sahtecilik tespit yöntemi önerilmiştir. BoW, görsel sözcükler üretmek ve video çerçevelerinden SIFT ile çıkarılmış anahtar noktalarından bir sözlük inşa edebilmek için literatürde çerçeve tekrarlama sahteciliği tespiti alanında ilk kez kullanılmıştır. Yöntem yüksek başarı oranı ve düşük çalışma zamanına sahiptir. Yönteme ilişkin detaylar yapılan çalışmalar kısmında verilmektedir. Önerilen yöntemin etkinliğini gösterebilmek ve literatürdeki benzer yöntemlerle

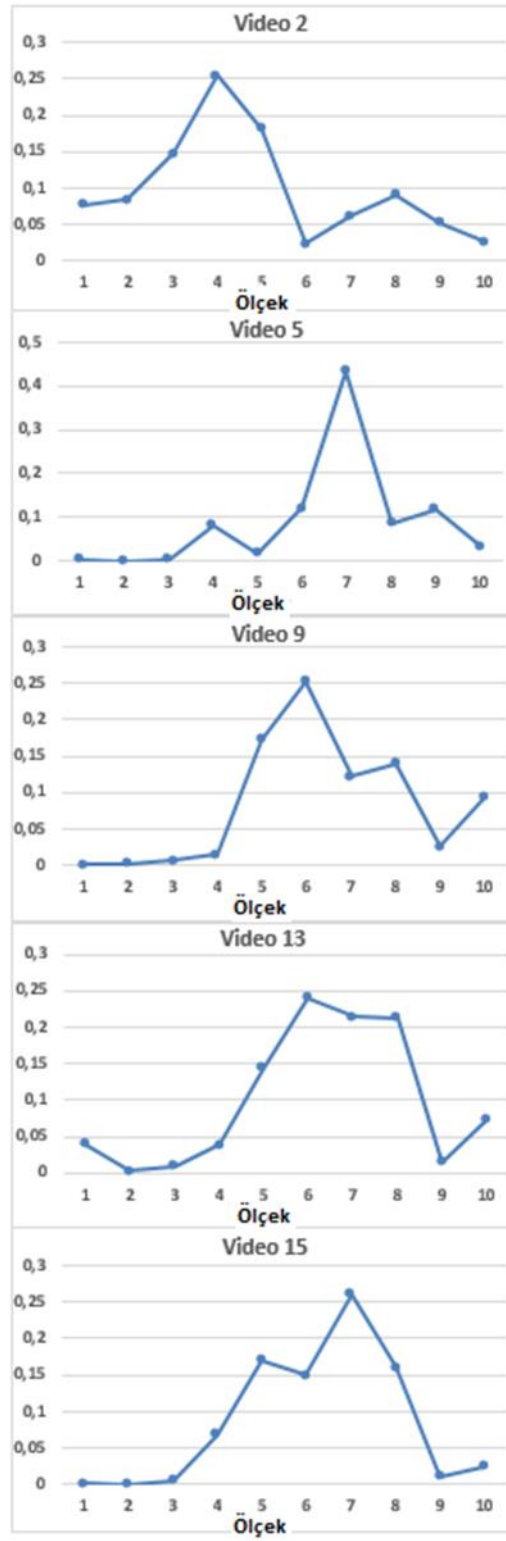
kıyaslamasını gerçekleştirebilmek amacıyla yapılan testler ve elde edilen deneysel sonuçlar aşağıda verilmektedir.

Algoritmanın içerisinde yer alan çeşitli değişkenler ve değer atamaları şu şekildedir: $th=3$, $w=30$, $d=2$, $g=5$, $l=100$. Çalışmada yapılan her bir değerlendirme alt bölüm olarak verilecektir.

Ölçek Seçim Algoritmasının Etkinlik Değerlendirmesi: Öncelikle ölçek seçim algoritmasının etkinliği değerlendirilmiştir. Yöntem çerçevelerden anahtar noktalarını elde etmekte, her anahtar noktası için bir tanımlayıcı oluşturmakta, tanımlayıcıları kullanarak görsel sözcükleri oluşturmaktadır. Görsel sözcükler cinsinden ifade edilen çerçevelerin benzerliği aynı zamanda PSNR testi kullanılarak da garantilenmektedir. Video içerisinde bir çerçeve başka bir yerde tekrarlandığı zaman, her iki çerçeveden üretilen özellik vektörleri benzerlik göstermektedir. Fakat çerçeveler aynı olsa bile elde edilen özellik vektörleri video sıkıştırma etkilerinden dolayı aynı olmayacaktır. Bu nedenle çalışmada belirli bir ölçek değeri ve üzerindeki anahtar noktalarının görsel sözcüklerin oluşturulmasında kullanımı önerilmiştir.

Testlerde kullanılmak amacıyla rasgele seçilen beş test videosu (Video 2, Video 5, Video 9, Video 13 ve Video 15) Virtual Dub yazılımı aracılığıyla, Ölçek Seçim algoritması için modifiye edilmiştir. Ölçek seçim algoritması ilgili videolar üzerinde uygulanmış ve normalize edilmiş ölçek histogramları, SC, elde edilmiştir. Şekil 3.54'te bu test videoları için histogramlar sırasıyla verilmiştir. Verilmiş olan histogramlardaki tepe değerlerine karşılık düşen bin değerleri, ilgili videolar için seçilecek olan ölçek değerlerini temsil etmektedir.

Önerilen yöntem, seçilen ölçek değerlerini kullanarak ilgili videolardan anahtar noktası seçimini gerçekleştirir. Örneğin, Video 5'teki çerçevelerden anahtar noktası çıkarımı esnasında 7 ve üzerindeki ölçek değerine sahip anahtar noktaları kullanılacaktır. Bu aşamada seçilen değerinin etkinliğini göstermek için bir ek deney daha yapılmıştır. Seçilen ölçek değeri ve ikinci en iyi ölçek değeri kullanılarak Video 5 üzerinde tanılama işlemi gerçekleştirilmiştir. Şekil 3.55.'te elde edilen tanılama performansını göstermektedir. PR, RR ve DA değerleri açısından değerlendirildiğinde Video 5 için seçilen değerinin optimum olduğu gözlemlenmektedir.



Video 2→Ölçek 4

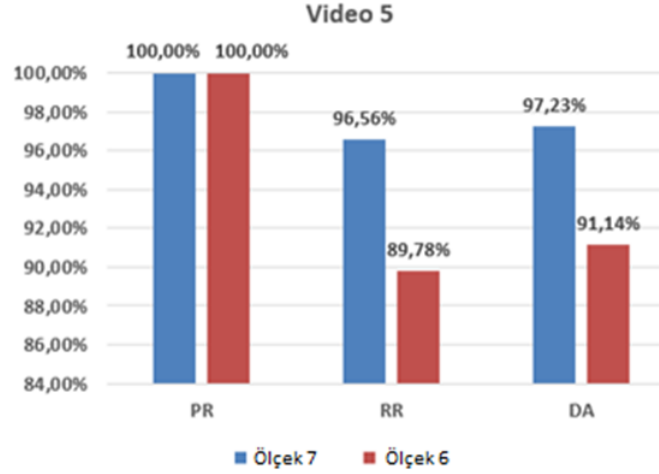
Video 5→Ölçek 7

Video 9→Ölçek 6

Video 13→Ölçek 6

Video 15→Ölçek 7

Şekil 3.54. Test videoları için elde edilen histogramlar ve belirlenen ölçek değerleri



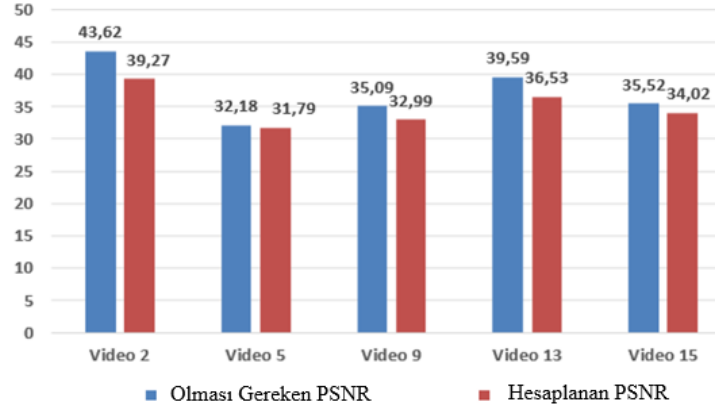
Şekil 3.55. Seçilen video için tanılama performansına ölçek değerinin etkisi

PSNR Seçim Algoritmasının Etkinlik Testi: PSNR seçim algoritmasının tanılama performansı üzerindeki etkisi çalışmada irdelenmiştir. Bu nedenle de olması gereken PSNR değeri ve seçilen PSNR değeri arasında bir kıyaslamaya gidilecektir.

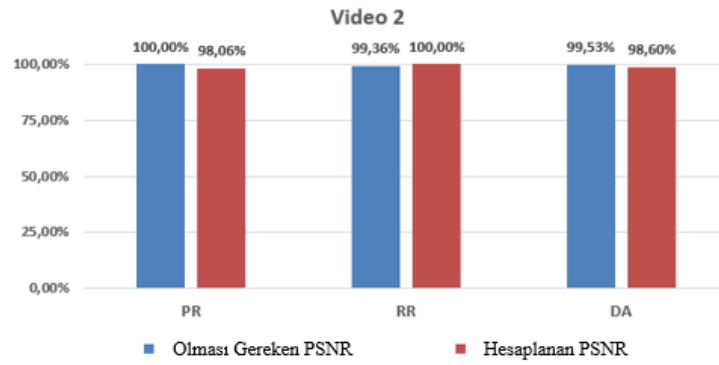
Video 2’de 100 ve 160. çerçeveler arası 315. çerçeveden sonra yapıştırılmıştır. Böylece 316 numaralı çerçeve 100 numaralı çerçevenin tekrarıdır. 100 ve 316 numaralı çerçeveler arasındaki hesaplanan PSNR değerinin, eğer herhangi bir video sıkıştırması olmasaydı, sonsuz olması beklenirdi. Fakat sahtecilik işleminin ardından videonun yeniden aynı codec özellikleri ile kaydedilmesinden dolayı PSNR değeri farklı çıkacaktır. Test amacıyla Video 2’deki 100-316, 101-317, ... çerçeveleri arasında PSNR değerleri hesaplanmış ve minimum PSNR değeri olması gereken PSNR değeri olarak seçilmiştir. Benzer şekilde diğer test videoları için de olması gereken PSNR değerleri hesaplanmıştır. Çalışmada önerilmiş olan PSNR seçim algoritması uygulanarak aynı test videoları üzerinde hesaplanan PSNR değerleri de elde edilmiştir. Şekil 3.56.’da olması gereken ve hesaplanan PSNR değerleri arasındaki mutlak fark verilmiştir. Şekil 3.56.’da gözlemlenebileceği gibi olması gereken ve hesaplanan PSNR değerleri arasındaki fark 4’ü geçmemekte ve [0-4] aralığında değişmektedir. PSNR seçim algoritmasının olması gereken değere yakınsadığı elde edilen deneysel sonuca göre söylenebilir.

Hesaplanan ve olması gereken PSNR değerleri arasındaki farkın tanılama performansı üzerindeki etkisi de aynı zamanda ölçülmüştür. Video 2 ve Video 5 test amacı ile seçilmiştir. Çünkü bu iki video için elde edilen farklar minimum ve maksimum değerleri içermektedir. Şekil 3.57.(a) ve 3.57.(b) her iki video için elde edilen PR, RR ve

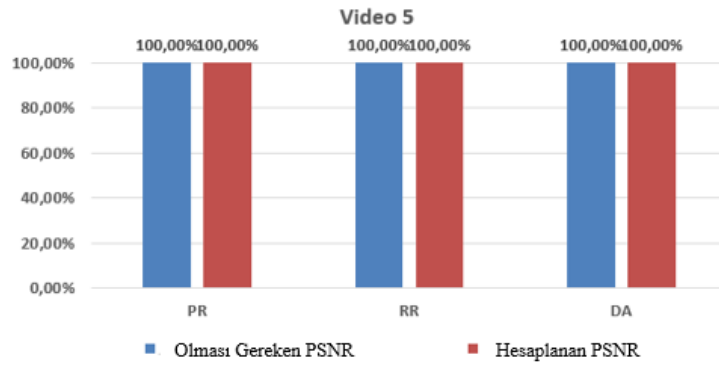
DA değerlerini göstermektedir. Şekilden de gözlemlenebileceği gibi, hesaplanan ve olması gereken PSNR değerleri arasındaki fark, tanılama performansını önemli derecede etkilememektedir.



Şekil 3.56. Seçilen videolar için olması gereken ve hesaplanan PSNR değerleri



(a)



(b)

Şekil 3.57. (a)Video 2 (b) Video 5 için tanılama performansı sonuçları

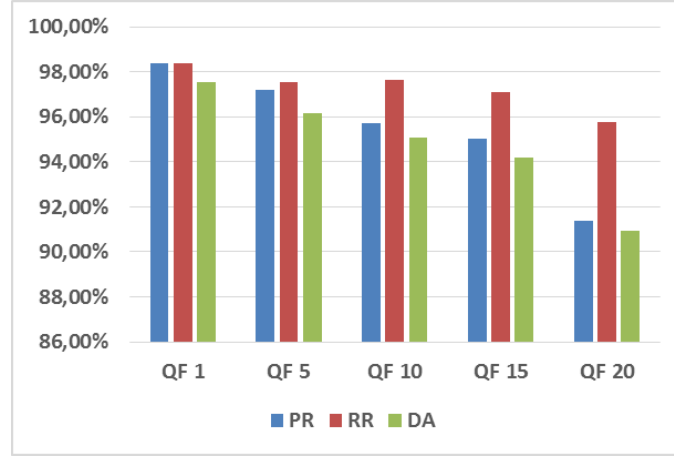
Tanımlayıcı Çıkarma Yönteminin Etkinliği: Önerilen yöntem çerçevelerden SIFT yöntemini kullanarak anahtar noktası çıkarmakta ve bu anahtar noktalar için 1×128 boyutlu tanımlayıcı elde edilmektedir. Yöntem ayrıca SIFT'in orijinal tanımlayıcısına ölçek ve yönlenme bilgisi olarak 2 boyut daha ekleyerek 1×130 boyutunda bir tanımlayıcı oluşturmaktadır. Çalışmada tanımlayıcıya eklenen ölçek ve yönlenme bilgisinin önerilen yöntemin performansına olan etkisi incelenmiştir. Önerilen tanımlayıcı çıkarma algoritmasının üstünlüğünü göstermek için aday listesi C deki çift sayısı incelenmiştir.

Önerilen tekniği sadece SIFT tanımlayıcısı ile karşılaştırmak için, yapılan çalışmalarda verilen görsel sözcük üretimi ve özellik vektörü oluşturma adımları SIFT tanımlayıcı ve önerilen tanımlayıcı çıkarma tekniği ile ayrı ayrı gerçekleştirilmiştir. Tablo 3.11.'de bütün test videoları için gerçekleştirilen bu iki yaklaşım sonucunda elde edilen aday listesi C'nin boyutları verilmektedir. Sonuçlar, önerilen yöntemin SIFT'e kıyasla daha elimine edilmiş bir aday listesi verdiğini ve algoritmanın geri kalan kısmı için daha az zamana ihtiyaç duyduğunu göstermektedir.

Tablo 3.11. Önerilen tanımlayıcı çıkarma algoritmasının etkinlik testi

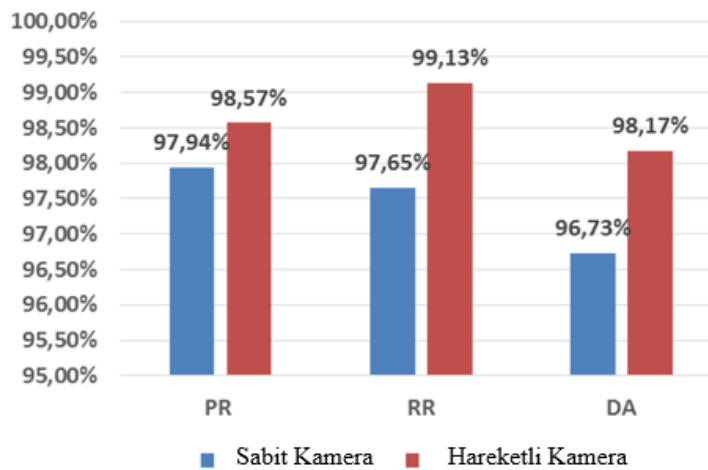
Aday Çiftlerin Sayısı		
Test video	SIFT	Önerilen
Video 1	641	321
Video 2	206	176
Video 3	63	41
Video 4	663	549
Video 5	53	40
Video 6	71	56
Video 7	1256	1163
Video 8	2658	2094
Video 9	143	113
Video 10	5681	3129
Video 11	1245	232
Video 12	813	722
Video 13	176	152
Video 14	2159	2049
Video 15	42	39
Video 16	83	60
Video 17	132	111
Video 18	67	40
Video 19	284	245
Video 20	57	37
Video 21	51	47
Video 22	33	30
Video 23	1935	1307
Video 24	46	39
Video 25	65	40
Video 26	214	198
Video 27	51	51
Video 28	960	207
Video 29	108	58
Video 30	64	54
Video 31	85	80

Önerilen Yöntemin Dayanıklılık Testi: Çalışmada önerilen yöntemin sıkıştırma ve video içeriğine karşı dayanıklılığı test edilecektir. İlk deney olarak test videoları sahtecilik işlemi yapıldıktan sonra farklı kalite faktörlerinde sıkıştırılmıştır. Test aşamasında farklı kalite faktörleri değerleri (1, 5, 10, 15 ve 20) kullanılmakla birlikte QF=1 en iyi kalite faktörüne karşılık gelmektedir. Yöntemin sıkıştırmaya karşı dayanıklılığı göstermek için DA, PR ve RR değerleri hesaplanmıştır. Şekil 3.58.'de bu kalite faktörleri için elde edilen DA, PR ve RR değerleri görülmektedir. Şekilden de görüldüğü üzere önerilen yöntem en kötü senaryo olan kalite faktörü QF=20 iken bile %91.4 PR, %95.75 RR ve %90.96 DA verirken en iyi kalite faktöründe %98.37 PR, %98.38 RR ve %97.54 DA değeri vermektedir. Sonuçlardan da görüldüğü üzere önerilen yöntem yüksek sıkıştırma oranlarına dahi oldukça dayanıklıdır.



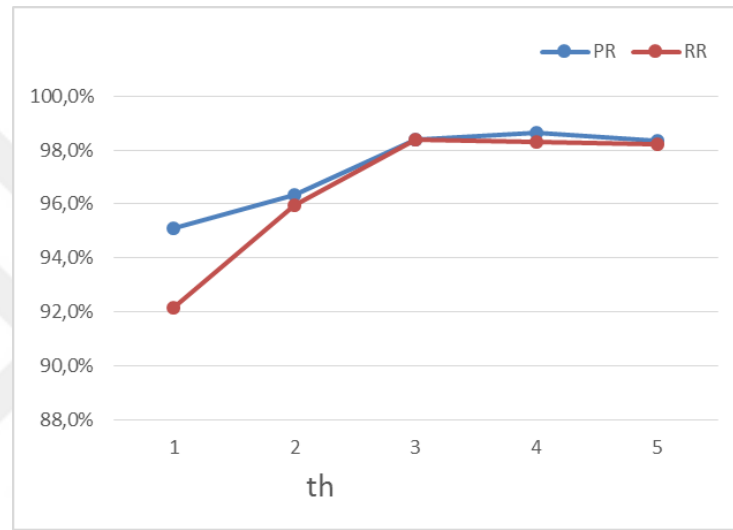
Şekil 3.58. Önerilen yöntemden farklı kalite faktörlerinde elde edilen PR, RR ve DA değerleri

Bir başka deneyde de video içeriğinin önerilen yöntemin performansına olan etkisi test edilmiştir. Tablo 3.1’de verilen test videoları iki kategoriye ayrılmaktadır. İlk kategorideki videolar sabit kamera ile çekilirken, hareketli kameralar diğer videoların çekiminde kullanılmıştır. Şekil 3.59.’da iki kategorideki videolar için ayrı ayrı hesaplanan DA, PR ve RR değerleri verilmektedir. Yöntem, hareketli kameralarla çekilen videolar için daha yüksek DA, PR ve RR vermekte çünkü bu kategorideki videolar sabit kamera ile çekilen videolara kıyasla daha az sıkıştırılmaktadır. Diğer yandan hareketli kameradan çekilen videolardaki çerçevelerin içeriği daha az benzerdir ve dolayısıyla sabit video içeriğiyle karşılaştırıldığında yapılan sahtecilik işlemi daha kolay tespit edilebilmektedir.



Şekil 3.59.Video içeriğine karşı önerilen yöntemin dayanıklılığı

Karşılaştırma Testleri: Çalışmada önerilen yöntemin performansını değerlendirebilmek için karşılaştırma testleri yapılmaktadır. Bununla birlikte, özellik vektörlerinin benzerliğini belirleyen algoritma tarafından kullanılan eşik değeri th , yöntemin sahtecilik tespiti aşamasında oldukça önemlidir. Bu sebeple yöntemin karşılaştırma testlerinden önce en iyi eşik değerini seçebilmek adına bir deney gerçekleştirilmiştir. Eşik değeri th 'nin değeri [1-5] aralığında değişmekte Tablo 3.1'de verilen test videoları için PR ve RR değerleri elde edilmektedir.



Şekil 3.60. th değerinin seçilmesi

Şekil 3.60.'dan da görüldüğü üzere th 'nin en iyi PR ve RR değeri verdiği değer 3 olarak gözükmektedir. Bu sebeple yapılan karşılaştırma testlerinde th değeri 3 olarak alınmıştır. Deneylerde özellik vektörü kıyaslamalarının zaman almaması için daha yüksek eşik değeri alınmamıştır.

Önerilen çalışma literatürdeki benzer çalışmalar ile iki kriter göz önüne alınarak kıyaslanmıştır: İcra zamanı ve tanılama performansı açısından kıyaslanacaktır. Tablo 3.1'de verilmiş olan 31 adet test videosu ortalama çalışma zamanı ve ortalama DA, PR ve RR değerlerini elde etmek için kullanılacaktır. Literatürdeki benzer yöntemlerin de ilgili videolar üzerindeki performans değerlendirmesi sunulacaktır.

Çerçeve başına düşen ortalama çalışma zamanı önerilen yöntemin benzer yöntemlerle [119-126] kıyaslanması için kullanılmıştır. Tablo 3.12'de, Tablo 3.1'de verilen test videoları ile yöntemlerin ortalama çalışma zamanı açısından kıyaslaması

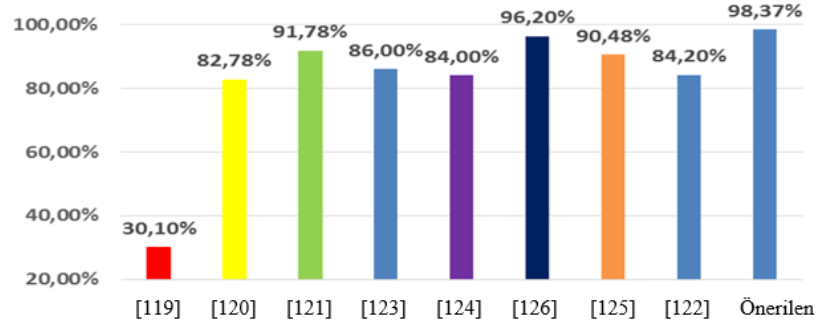
verilmektedir. Tablodan da gözlemlenebileceği gibi önerilen yöntem, benzer yöntemlere göre daha hızlı çalışmaktadır. Diğer yöntemlerin [119-126] ortalama çalışma zamanları yaklaşık olarak sırasıyla 294, 140, 0.41, 0.42, 0.35, 0.8, 0.7 ve 0.6 iken önerilen yöntemin çalışma zamanı 0.2 saniyedir. Böyle bir iyileştirme günümüzdeki video uzunlukları hesaba katıldığında önemli bir iyileştirme olarak göze çarpmaktadır.

Tablo 3.12. Önerilen yöntemin benzer çalışmalarla çalışma zamanı açısından kıyaslanması

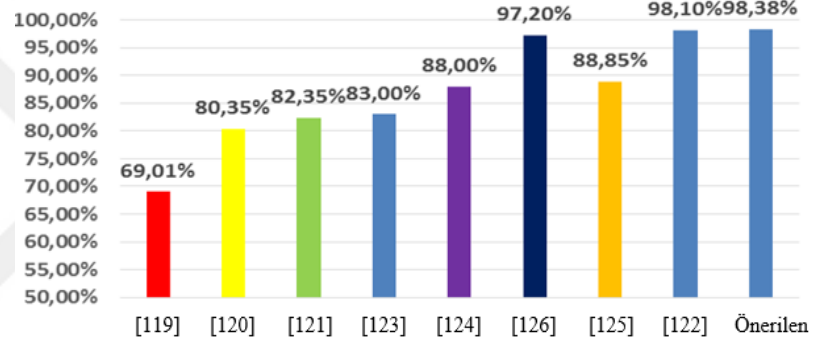
Yöntem	Çalışma zamanı(s/çerçeve)
Wang vd. [119]	294.67
Lin vd. [120]	140.6
Yang vd. [121]	0.41
Zhang vd. [123]	0.42
Zheng vd. [124]	0.35
Liu vd. [126]	0.8
Aghamaleki vd. [125]	0.72
Singh vd. [122]	0.62
Önerilen yöntem	0.2

Son deneyde ise önerilen yöntem benzer çalışmalar ile DA, PR ve RR açısından kıyaslanmıştır. Şekil 3.61.(a) yöntemin PR değerlerini listeler. Önerilen yöntem yaklaşık %98 PR değerlerine sahip iken performans açısından en yakın yöntem olan [126] %96 PR değerlerine sahiptir. Şekil 3.61.(b)'de ise yöntemlerin RR değerleri açısından kıyaslaması verilmiştir. Literatürdeki yöntemler [119-126] ve önerilen yöntemin RR değerleri sırasıyla %69, %80, %82, %83, %88, %97, %88, %98 ve %98 şeklindedir. Önerilen yöntem, Şekil 3.61.(b)'den de gözlemlenebileceği gibi RR oranları açısından da daha iyi sonuçlar üretmektedir. DA değerleri göz önüne alındığında sekiz yöntem ve önerilen yöntem sırasıyla %59, %79, %83, %84, %86, %94, %90, %96 ve %97 değerlerine sahiptir ve Şekil 3.61.(c)'de değerler gözlemlenmektedir. Şekilden de gözlemlendiği gibi yöntem daha yüksek doğrulama yeteneğine sahiptir.

Tüm sonuçlar değerlendirildiğinde çalışmada önerilmiş olan yöntem daha iyi çalışma zamanına ve daha yüksek tanılama yeteneğine sahiptir. Aynı zamanda literatürdeki çalışmalardan farklı olarak test videoları sıkıştırılmış formatta hazırlanmıştır. Sıkıştırılmış formatlarda dahi çalışmanın tanılama performansının yüksek olması, yöntemin başarısını ortaya koymaktadır.

DUYARLILIK

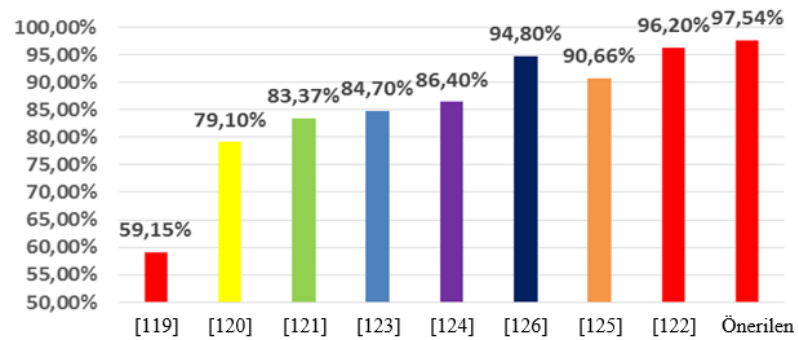
(a)

SEÇİCİLİK

(b)

Şekil 3.61. Önerilen yöntemin literatürdeki benzer çalışmalarla (a) PR(b)RR(c)DA açısından kıyaslanması

Şekil 3.61.'in devamı

Doğrulama Oranı

(c)

3.13. Videolarda Bölge Tabanlı Sahtecilik Tespitinin Değerlendirilmesi ve Elde Edilen Deneysel Sonuçlar

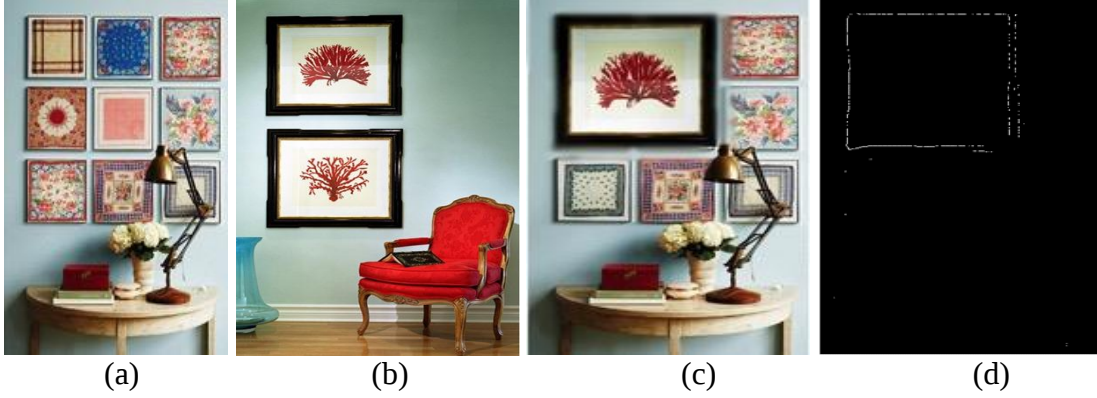
Önerilen yöntem videodaki hem kopyala-yapıştır hem de video birleştirme sahteciliğini tespit edebilmektedir. Aynı zamanda yöntem sahtecilik tespiti izlerinden faydalandığı için ayna simetrik kopyala-yapıştır sahteciliğine de dayanıklıdır. Yönteme ilişkin detaylar yapılan çalışmalar kısmında verilmektedir. Önerilen yöntemin etkinliğini gösterebilmek ve literatürdeki benzer yöntemlerle kıyaslamasını gerçekleştirebilmek amacıyla yapılan testler ve elde edilen deneysel sonuçlar aşağıda verilmektedir.



Şekil 3.62.(a) Merkezi orijinal arka planı bulanık görüntü (b) Bulanık kenarları etiketlemiş sonuç görüntüsü

Detayları yapılan çalışmalar kısmında verilen yöntemin, bulanık kenarları tespit edebilme oranını gösterebilmek adına Şekil 3.62.'deki örnek verilmiştir. Şekil 3.62.(a)'daki merkezi orijinal arka planı bulanık görüntü için elde edilen bulanık kenarları etiketlemiş sonuç görüntüsü Şekil 3.62.(b)'de verilmiştir.

Bir diğer deneysel sonuçta da bölgesel sahtecilik yapılan test görüntüsü üzerinde yöntem uygulanmış ve bulanık kenar tespit sonuçları Şekil 3.63.'te verilmiştir. Şekil 3.63.(a) ve 3.63.(b) orijinal resimleri gösterirken, görüntü birleştirme sonucu oluşturulan sahte görüntü Şekil 3.63.(c)'de verilmiştir. Yöntemin tespit sonucu ise Şekil 3.63.(d)'de gösterilmektedir. Şekil 3.63.(c)'deki görüntüyü oluşturma amacı ile sahte bölgenin etrafına 5×5 çerçeve $\sigma = 5$ Gauss fonksiyonu ile bulanıklaştırma işlemi uygulanmıştır.



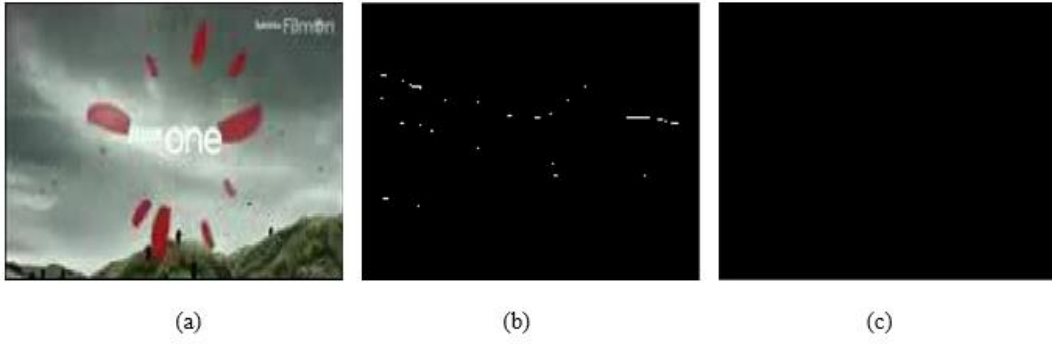
Şekil 3.63. Görüntü birleştirme sahteciliğine bir örnek (a) Test görüntüsü (b) Sahte bölgenin alındığı görüntü (c) Sahte görüntü (d) Tespit sonucu

Şekilden de görüldüğü gibi bulanık kenarlar işaretlenmekte ve bulanık kenarların Şekil 3.63(d)'de görüldüğü gibi beyaz ile temsil edildiği B görüntüsü oluşturulmaktadır. Bir sonraki bölümde, bulanık kenarların olduğu bölgeleri gösteren B görüntüsü üzerinde son işlemler gerçekleştirilecektir.

İşaretlenmiş Bulanık Kenarların Kullanımı İle Sahteciliğin Tespiti: Bulanık kenarları temsil eden B görüntüsü üzerinde sonradan işlemler uygulanacaktır. Bu amaçla B üzerindeki her bir beyaz piksel merkez kabul edilerek 3x3'lük komşuluğundaki kenar sayısının 3'ü geçmesi durumunda ilgili piksel, bulanık olarak etiketlendirilecektir. Bahsedilen işlemin B için uygulanış şekli ifade (3.4)'te verilmiştir.

$$B'(i, j) = \sum_{p=-1}^1 \sum_{q=-1}^1 B(i + p, j + q) \quad (3.4)$$

Son işlem sonrası üretilen B'ikili görüntüsünde beyaz piksel bulunması, referans çerçevenin bulanık kenar içerdiğini göstermektedir. Son işlemin performansına ait iki farklı deney gerçekleştirilmiştir. İlk deneyde, Şekil 3.64.(a)'da verilmiş olan orijinal referans çerçevede (sahtecilik içermeyen) bulanık kenarlar tespit edilmiştir. Bu işlem sonucu üretilen B görüntüsü de Şekil 3.64.(b)'de verilmiştir.

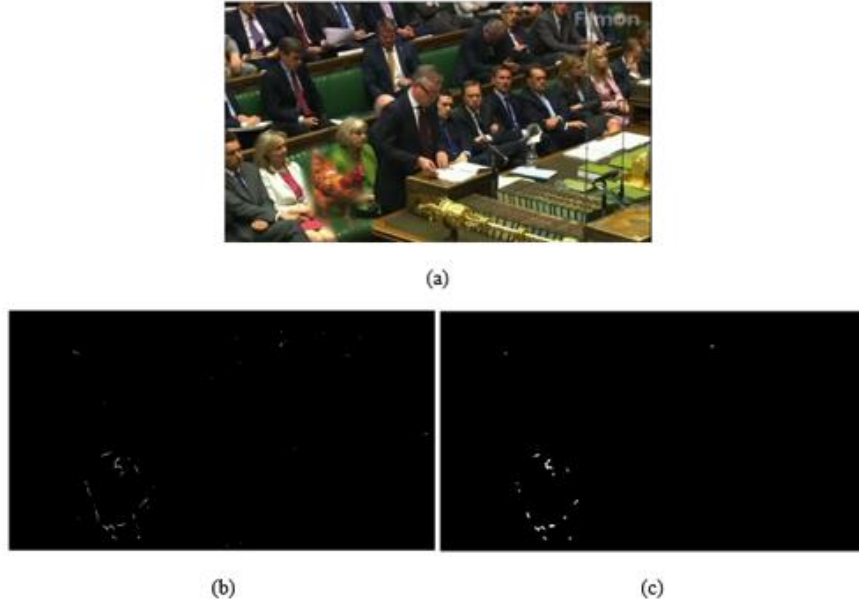


Şekil 3.64. (a) Referans çerçeve (b) Bulanık kenar tespit sonucu (c) Son işlem sonrası kenar tespit sonucu

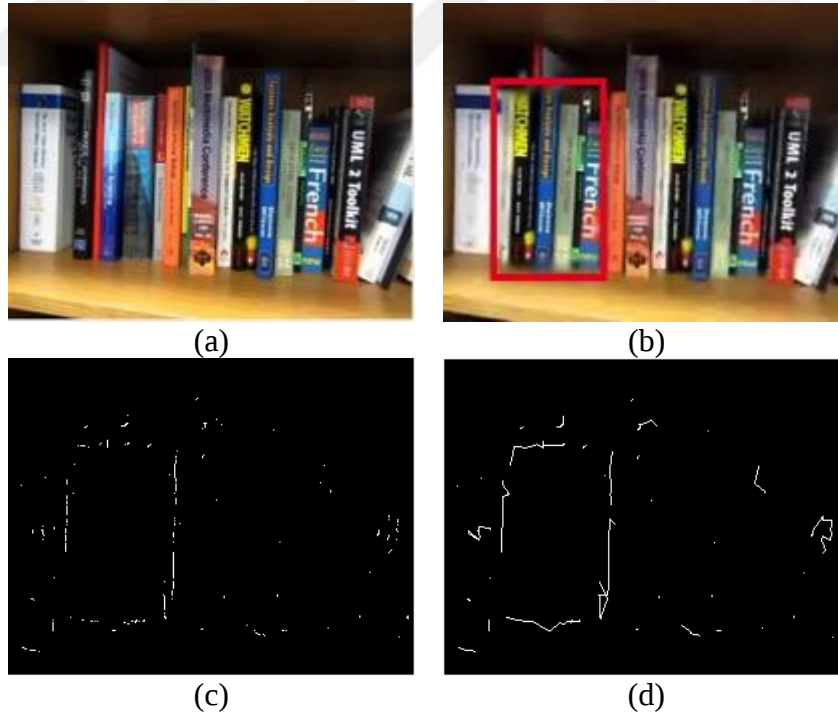
Şekilden de gözlemlenebileceği gibi bazı pikseller bulanık kenar olarak sınıflandırılmıştır. Böyle bir hatalı sınıflandırmaya videonun düşük çözünürlüğü olmasının yol açtığı düşünülmüştür. Son işlemin uygulanması sonucu üretilen Şekil 3.64.(c)'de ki B'görüntüsünde ise hatalı etiketlendirmelerin elimine olduğu görülmektedir. Son işlem sonrası ilgili çerçeve, sistem tarafından orjinal olarak etiketlenmektedir.

Bir diğer deneysel sonuçta ise, Şekil 3.65.(a)'da verilen sahte referans çerçeve üzerinde yöntem uygulanmış ve Şekil 3.65.(b)'deki sonuç elde edilmiştir. Son işlem ardından üretilen görüntü ise Şekil 3.65.(c)'de verilmiştir. Deneysel sonuçtan da gözlemlenebileceği gibi son işlem operasyonu tespit edilen kenarları kuvvetlendirmiş ve elde edilen beyaz pikseller ilgili çerçevenin sahte olarak sınıflandırılmasını sağlamıştır. Referans çerçevenin sahte olarak işaretlemesi de ilgili videonun sahte olduğunu göstermektedir.

Bölgesel sahteciliğin tespiti için önerilen yöntem bulanık kenarları şekillerde gözlemlendiği gibi noktasal olarak işaretlemektedir. Problemli bölgenin kullanıcı tarafından daha net olarak gözlemlenebilmesi amacı ile çalışmada her beyaz piksel ile ona en yakın olan beyaz piksel arasında doğru çizme algoritması kodlanmıştır. Böylelikle sahte olan bölgenin doğrular ile işaretlenebilmesi mümkün kılınacaktır. En yakın komşuların doğru ile birleştirilmesine ilişkin elde edilen çeşitli görsel sonuçlar Şekil 3.66'da verilmiştir.



Şekil 3.65. (a) Sahte referans çerçeve (b) Bulanık kenar tespit sonucu (c) Son işlem sonrası kenar tespit sonucu



Şekil 3.66. En yakın noktaların birleştirilmesine ilişkin görsel sonuçlar

Şekil 3.66.(c)'deki noktasal işaretleme sonuçlarında her ne kadar birleştirmenin olduğu bölge gözlemlenebilse de, Şekil 3.66.(d)'de ki en yakın noktaların doğrular ile

birleştirilmesi sonucu elde edilen işaretlenmiş görüntü daha kullanıcı dostu olarak gözükmetedir. Bu nedenle, bundan sonraki görsel sonuçlar en yakın noktaların doğru ile birleştirilmesi sonucu elde edilen görüntüler üzerinden verilecektir.

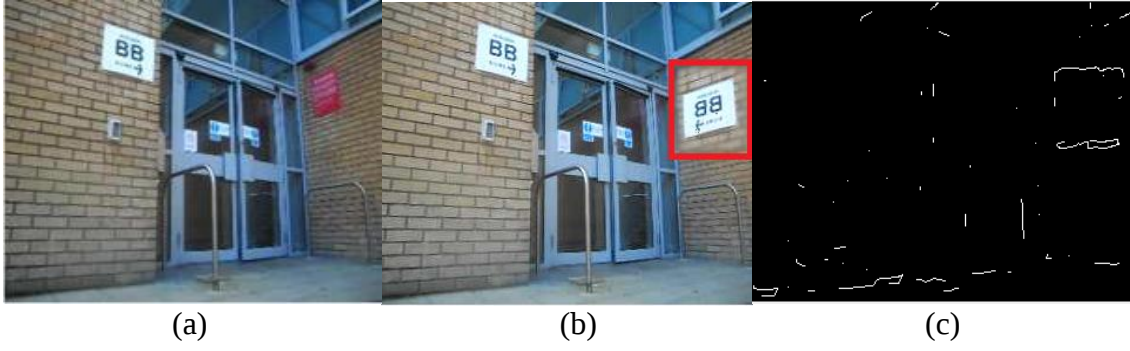
Bir diğer bir deneysel sonuçta , yöntemin, detayları Tablo 3.3'te verilen test veritabanı üzerindeki sınıflama başarısı ölçülecektir. Bölgesel sahteciliğin tespiti için önerilen yöntemin başarımını ölçmek amacı ile yapılan test esnasında 20 orijinal, 20 sahte video kullanılarak yöntemin videoları sahte/sahte değil olarak işaretlemesine göre sınıflama başarısı ölçülecektir. DA ölçütü kullanılarak önerilen yöntemin performansı değerlendirilmiştir. TP, sahte videonun sahte olarak işaretlenmesini temsil ederken, FP ise sahte videonun orijinal olarak etiketlenmesini ifade eder. Benzer şekilde TN ve FN sırası ile, orijinal videonun orijinal olarak ve orijinal videonun sahte olarak işaretlenmesine karşılık düşmektedir. Tablo 3.13'te üretilen sonuçlarda da gözlemlenebileceği gibi DA değerleri sırası ile 70, 75, 77.5, 80 ve 92.5 olarak bulunmuştur. Test veritabanı üzerinde elde edilen sonuçlarda yöntemin sınıflama başarısının literatürdeki benzer yöntemlere kıyasla daha yüksek sınıflama başarısı gösterdiği gözlemlenmektedir.

Kıyaslamalarda aynı zamanda yöntemler iki farklı metrik açısından da değerlendirmeye alınmıştır. Bunlardan ilki kopyala-yapıştır ile kopyalanan bölgenin ayna simetriği alınarak sahte videonun oluşturulması durumunda dayanıklılığıdır.

Tablo 3.13. Test veri tabanı üzerinde elde edilen sonuçlar

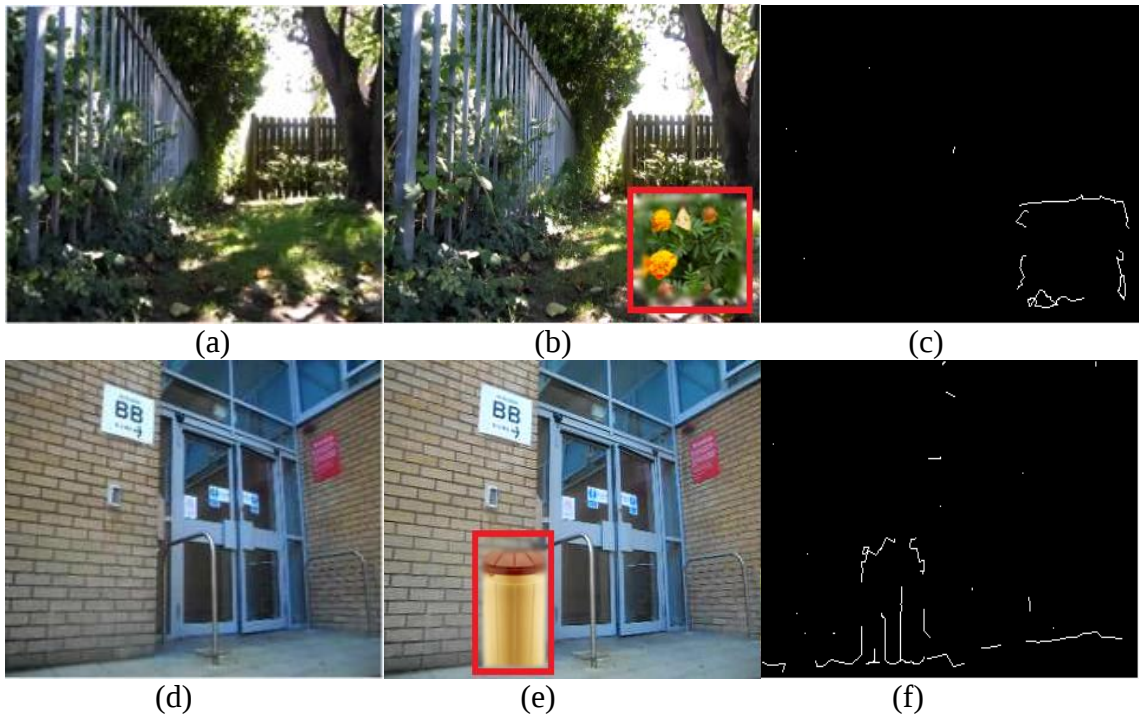
	DA	Ayna simetrik Kopyala-yapıştır dayanıklılık	Video Birleştirmeye Dayanıklılık
Wang ve Farid [119]	70	Hayır	Hayır
Subramanyam [135]	75	Hayır	Hayır
Pandey [137]	77,5	Hayır	Hayır
Su ve Li [181]	80	Evet	Hayır
Önerilen Yöntem	92.5	Evet	Evet

Şekil 3.67'de yöntemin ayna dönüşümü durumundaki dayanıklılığını gösterilmiştir. Şekil 3.67.(a)'da test videosundaki bir çerçeve gösterilirken, 3.67.(b)'de ayna simetri atağı uygulanarak oluşturulmuş sahte çerçeve verilmiştir. Önerilen yöntemin uygulanması sonucu elde edilen 3.67.(c) görüntüsünde ise kopyalanıp ayna simetriği alınarak yapıştırılmış bölgenin tespit edilebildiği gösterilmiştir.



Şekil 3.67. Ayna Simetrik Bölgesel Tekrarlama Atağı

Tablo 3.12’de vurgulandığı gibi önerilen yöntem, literatürdeki benzer çalışmalardan farklı olarak Ayna simetrik bölge tekrarlar ataklarını tespit edebilmektedir. Yine aynı tabloda çalışmanın, video birleştirme ataklarına karşı da dayanıklı olduğu gösterilmektedir. Rapor edilen durumu destekleyebilmek amacı ile gerçekleştirilen bir diğer test senaryosunda ise Şekil 3.68.(a)’daki orjinal çerçeve 3.68.(b)’de görüldüğü gibi modifiye edilmiştir. Yöntem, Şekil 3.68.(c)’de görüldüğü gibi video birleştirmenin uygulandığı kısmı işaretleyebilme yeteneğine sahiptir. Şekil 3.68.(d)’de verilen farklı bir test çerçevesine atak uygulanması sonucu elde edilen sahte çerçeve ise 3.68.(e)’de verilmiştir. Yöntemin tespit sonucundan da gözlemlenebileceği gibi birleştirmenin yapıldığı bölge tespit edilebilmiştir.



Şekil 3.68. Video Birleştirme atağına karşı dayanıklılık testi



4. SONUÇLAR

Kameralar, fotoğraf makineleri ve hatta cep telefonlarıyla bile görüntü ve video kayıtları son yıllardaki sensör ve depolama teknolojilerindeki hızlı gelişme sayesinde her an ve her yerde alınabilmektedir. Bununla birlikte görüntü/video düzenleme araçlarının kolay kullanılabilir olması, tecrübesiz kullanıcıları bile videolar üzerinde işlem yapabilir hale getirmektedir. Yapılan işlemler kimi zaman görüntü/video kalitesini artırma amacıyla iyi niyetli olurken, kimi zamanda sahte görüntü/video oluşturma amacıyla kötü niyetli olmaktadır. Bu açıdan düşünüldüğünde görüntü/video içeriğinin adli ortamlarda delil olarak kullanılması durumunda ilk alındığı andan itibaren değişmediğinin doğrulanması büyük önem arz etmektedir.

Son yıllarda görüntü/videoların doğruluğunu ve değiştirilmediğini ispatlamak için çeşitli yöntemler araştırmacılar tarafından geliştirilmektedir. Literatürde görüntü/video sahteciliği tespit yöntemleri aktif ve pasif doğrulama yöntemleri olmak üzere ikiye ayrılmaktadır. Aktif doğrulama yöntemlerinden sayısal damgalama ve sayısal imzada görüntü/video içerisine saklanan özel bilgi aracılığıyla, görüntünün bütünlüğünün korunduğu doğrulanabilmektedir. Görüntü/videonun değiştirilmesi durumunda, yerleştirilen bilgi doğru bir şekilde yapılandırılmayacaktır. Fakat genel amaçlı kullanılan birçok görüntü/video yakalama aygıtı, görüntü/video alma ve saklama sırasında, görüntü/videonun bütünlüğünü koruyacak bir bilgiyi yerleştirecek yetenekte değildir. Sayısal ortamlarda adli inceleme yöntemleri olarak da bilinen pasif doğrulama yöntemlerinde ise görüntü/video ile ilgili herhangi bir ön bilgi bulunmadığı ve önceden görüntünün bütünlüğünü korumak üzere hiçbir önlemin alınmadığı durumlarda dahi görüntü/video doğrulaması yapılabilmektedir. Pasif yöntemler görüntü/videodan çıkarılan özelliklere göre doğrulamayı gerçekleştirmektedir. Bu durumda pasif doğrulama yöntemlerini pek çok durum için kabul edilebilir tek çözüm yolu haline getirmektedir.

Literatürde pasif doğrulama yöntemlerini kullanarak görüntü/video sahteciliklerinin tespitini gerçekleştiren yöntemlerde tanılama performansının yükseltilmesi, hesaplama karmaşıklığının azaltılıp, işlem zamanının düşürülmesi, kullanılan eşik değerlerinin görüntü/video içerikleri ve boyutundan bağımsız bir şekilde dinamik olarak belirlenmesi, özellik çıkarım yönteminin görüntü için ara ve son işlem operasyonlarına dayanıklı olması, video için ise sıkıştırma türünden bağımsız ve sıkıştırmaya dayanıklı olması, düz

geçişsiz dokularda da yapılan sahteciliğin tespit edilebilmesi araştırmacılar tarafından iyileştirilmesi hedeflenen unsurlardır. Tez süresince hedefleri iyileştirmek doğrultusunda önerilen sahtecilik tespit yöntemleri ve elde edilen sonuçlar kısaca aşağıdaki şekilde özetlenebilir.

1. Görüntü bloklarından özellik çıkarımında literatürde ilk kez Renk Momentleri ve Renk düzeni tanımlayıcısı yöntemlerini kullanan çalışma AWGN, Gauss bulanıklaştırmaya ve JPEG sıkıştırmaya karşı yüksek dayanıklılık göstermekte ve literatürdeki diğer AKD tabanlı çalışmalara [12, 33, 72] göre daha yüksek doğruluk oranı ve daha düşük yanlış negatif değeri vermektedir. Ayrıca blokların özellik çıkarımından önce sınıflandırılması ve sınıfların iş parçacıklarına dağıtılması sonucu işlem zamanında iyileştirilmeye gidilmiştir.

2. Literatürdeki görüntü sahteciliğinin tespiti için önerilmiş olan yöntemler görüntü bloklarının benzerlik aşamasında sabit bir eşik değeri kullanmaktadır. Tez kapsamında önerilen çalışmada Benford'un genelleştirilmiş yasası kullanılarak test görüntüsünün sıkıştırma geçmişi belirlenmekte ve AKD faz ile özellik çıkarılarak özellik vektörlerinin değer aralığı sınırlanmaktadır. Bunun sonucunda yöntem iki vektörün kıyaslamasında eşit eleman sayılarını değerlendirmekte ve test görüntüsünün sıkıştırma geçmişini kullanarak otomatik eşik değeri belirlemektedir. Son işlem operasyonlarından JPEG sıkıştırmaya, Gauss bulanıklaştırma ve beyaz Gauss gürültüsüne oldukça dayanıklı olan kopyala-yapıştır sahteciliği tespit yöntemi literatürdeki benzer çalışmalara [12, 14, 33, 35, 70, 72] kıyasla daha yüksek doğruluk oranı ve daha düşük yanlış negatif değeri vermektedir.

3. Literatürde ilk olarak kopyala-yapıştır sahteciliği tespitinde LBP ile AKD yöntemi birleştiren çalışma literatürdeki AKD tabanlı çalışmalarla [12, 72] kıyaslandığında çeşitli ataklar altında dahi kopyalanıp yapıştırılan bölgeleri daha yüksek doğruluk oranı ve daha düşük yanlış negatif değerleri ile tespit etmektedir. Önerilen yöntemle sahtecilik tespitinde AKD katsayılarının işaret bilgisinin yeterli olduğu ortaya konulmaktadır.

4. Görüntü bloklarından özellik çıkarımında literatürde ilk kez kopyala-yapıştır sahteciliği tespit alanında LPQ yöntemi kullanılan çalışmada bulanıklaştırma işleminden bağımsız yeni bir kopyala-yapıştır sahteciliği tespit yöntemi geliştirilmiştir. Literatürdeki benzer çalışmalara [14, 72] kıyasla önerilen yöntem daha yüksek doğrulama oranı ve daha düşük yanlış negatif değeriyle tekrarlanan bölgeleri tespit etmektedir.

5. Literatürde kopyala-yapıştır sahteciliğinin tespiti için önerilen anahtar nokta tabanlı yöntemlerin çoğunda SIFT ve SURF yöntemleri kullanılmakta ve gri seviye

görüntü üzerinde anahtar nokta çıkarımı gerçekleştirilmektedir. Tez kapsamında önerilen çalışmada görüntüden anahtar nokta çıkarımında literatürde ilk olarak renkli görüntü alınarak renkli SIFT algoritmasının kullanılması gerçekleştirilmiştir. Çalışmada renkli SIFT'in SIFT'e göre daha etkin sahtecilik tespiti yapabildiği test sonuçları ile birlikte rapor edilmiştir.

6. Literatürde kopyala-yapıştır sahteciliğinin tespiti için önerilmiş anahtar nokta ve blok tabanlı çalışmalar düz alanda yapılmış sahtecilikleri tespit edememektedir. Çünkü anahtar nokta çıkarım algoritmaları görüntüdeki düz alanlarda anahtar nokta çıkaramamaktadır. Bu nedenle tez kapsamında gerçekleştirilen çalışmada kopyala-yapıştır sahteciliği tespiti için yeni bir anahtar nokta tabanlı görüntü doğrulama yöntemi önerilmiştir. Önerilen yöntem anahtar nokta seçimine dayalı olup SIFT uygulanmadan önce LPQ uygulayarak doku bilgisini çıkarmaktadır. LPQ görüntünün görünüşte düz bölgelerinden doku bilgisi çıkarmaktadır. Bu sayede anahtar nokta çıkarım algoritmalarının düz bölgelerden de anahtar nokta çıkarabilmesi sağlanmıştır. Önerilen yöntemle blok tabanlı ve anahtar nokta tabanlı yöntemlerin düz bölgelerde gerçekleştirilen sahteciliklerdeki eksikliği giderilmiştir.

7. Sahtecilik yapılan görüntünün medikal görüntü olması durumu da çalışma kapsamında değerlendirilmiştir. Çalışmada oynanmış medikal görüntülerin tespiti için yeni bir pasif görüntü doğrulama yöntemi önerilmesiyle aktif doğrulama yöntemlerindeki görüntü alındıktan sonra bir damga gömme ihtiyacı ortadan kaldırılmıştır. Medikal görüntülerde çoklukla mevcut düz görüntülerdeki sahteciliği anahtar nokta tabanlı yöntemler tespit edemediğinden, bu teknikler görüntü dokusu gibi yapısal bilginin kullanımı gerektirmektedir. Önerilen yöntem anahtar nokta seçimine dayalı olup SIFT'den önce doku bilgisini vurgulamak için LBPROT operatörünü kullanmaktadır. LBPROT medikal görüntülerdeki özellikle düz bölgelerden doku bilgisi çıkarmaktadır. Kopyala-yapıştır yapılmış bölge ara işlem operasyonlarına (dönme, ölçekleme) uğrasa dahi yüksek tanılama performansı ile tespit edilmektedir. Önerilen yöntem ile literatürdeki anahtar nokta tabanlı pasif doğrulama mekanizmalarının en önemli eksikliği rapor edilmiş ve giderilmiştir.

8. Videodaki çerçeve tekrarlama sahteciliği tespiti için literatürde ilk kez çerçevelerden LDB ile özellik çıkaran çalışmada sahtecilik tespiti diğer yöntemlere [119-121] göre daha yüksek doğruluk ve daha düşük çalışma zamanı ile gerçekleştirilmiştir. Yöntemde sahteciliğin uzaklığını tahmin etmek ve tekrarlanmış çerçevelerin tam yerini

belirlemek için benzer özellik vektörlerine sahip benzer çerçeveler arasındaki uzaklık kullanılmaktadır. Benzer çerçeveler arasındaki PSNR, bu uzaklıkları üç sınıfa ayıştırmak için kullanılmakta ve kural tabanlı mekanizma üye olduğu sınıfa göre sahte çerçeveleri belirlemektedir.

9. Videoda çerçeve tekrarlama sahteciliği yapılırken tekrarı yapılacak olan çerçeve grubu, eklenmeden önce eğer ayna yansıması alınarak eklenmiş ise literatürde çerçeve tekrarlama sahteciliği tespiti için önerilen yöntemler benzer algoritmik yapıya sahip oldukları için bu sahteciliği tespit edememektedir. Tez kapsamında önerilen çalışmada yeni bir çerçeve tekrarlama/yansıma sahteciliği tespit yöntemi geliştirilmiştir. Yöntemde çerçevelerden ikili özellikler çıkarılarak özellikler arasındaki benzerlik belirlenmektedir. Sonrasında PSNR kullanılarak çok sayıda aday elimine edilip, sahte çerçevelerin tespiti iyileştirilmektedir. Önerilen yöntem benzer çalışmalarla [119-121] kıyaslandığında daha iyi çalışma zamanına ve tespit doğruluğuna sahiptir.

10. BoW modeline dayalı yeni bir sahtecilik tespit yöntemi öneren çalışmada, BoW, görsel sözcükler üretmek ve video çerçevelerinden SIFT ile çıkarılmış anahtar noktalarından bir sözlük inşa edebilmek için literatürde çerçeve tekrarlama sahteciliği tespiti alanında ilk kez kullanılmıştır. Çerçeve özelliklerinden yani çerçevelerdeki anahtar noktalardan elde edilen görsel sözcüklerden video içerisindeki tekrar eden bölümlerin tespiti için faydalanılmıştır. Yöntemden literatürdeki benzer çalışmalara [119-126] kıyasla yüksek başarı oranı ve düşük çalışma zamanı elde edilmiştir.

11. Tez kapsamında videoda çerçeve tekrarlama sahteciliği yanında bölgesel tabanlı sahtecilik de tespit edilmektedir. Çalışmada video çerçevelerindeki bulanık kenarların tespiti ile bölgesel sahtecilik izleri belirlenmektedir. Önerilen yöntem videodaki hem kopyala-yapıştır hem de video birleştirme sahteciliğini tespit edebilmektedir. Aynı zamanda yöntem sahtecilik tespiti izlerinden faydalandığı için ayna simetrik kopyala-yapıştır sahteciliğine de dayanıklıdır. Önerilen yöntem literatürdeki diğer çalışmalara [119, 135, 137, 181] kıyasla daha yüksek tanılama performansına sahiptir.

12. Pasif Video Doğrulama sistemi adlı 3001 Tübitak projesi kapsamında video sahteciliklerin tespiti için geliştirilen yöntemler tek bir çatı altında toplanarak, Pasif Video Doğrulama Sistemi isimli bir yazılım geliştirilmiştir.

5. ÖNERİLER

Sayısal görüntü/videolardaki sahteciliklerin tespit edilmesi konusu, literatürde son yıllarda gittikçe artan bir öneme sahiptir. Görüntü sahteciliğinin tespiti alanında özellikle 2004 yılından itibaren birçok araştırma yapılmıştır. Sayısal videolar üzerindeki sahtecilikleri tespit eden çalışmalarda ise son yıllara doğru bir artış olmuştur. Sayısal görüntü/videoların doğrulanması alanındaki araştırmaların son yıllara doğru hız kazanması ve yayın miktarındaki azlık, konunun önemini ve araştırma potansiyelinin yüksekliğini göstermektedir. Araştırmacılar çoğunlukla var olan sistemlerdeki problemlerin üzerine gitmeyi hedeflemektedir.

Kopyala-yapıştır sahteciliğinin tespitinde ilk yapılan çalışma blok tabanlıdır. Bu çalışmayı takiben çıkan birçok çalışmada da kopyala-yapıştır sahteciliği tespiti için önerilmiş olan yöntemler blok tabanlıdır. Sonrasında blok tabanlı yöntemlerde özellik çıkarma aşamasının hesaplama maliyetinin yüksek olmasından dolayı anahtar noktası tabanlı yöntemler önerilmiştir. Blok tabanlı yöntemlerde özellikle blok benzerliklerinin araştırılması daha çok işlem zamanı gerektirmektedir. Bunun için bloklardan hashing algoritmaları kullanılarak özellik çıkarılabilir. Sonrasında aynı hash değerine sahip bloklar kendi içinde karşılaştırılarak işlem zamanı düşürülebilir.

Kopyala-yapıştır sahteciliğinin tespitinde anahtar noktası tabanlı yöntemler ile dokusuz bölgelerden yeterli sayıda özellik çıkarılamamaktadır. Ancak dokulu bölgelerden elde edilen özellikler çoğu zaman yeterli olmaktadır. Bu sebepten ötürü dokusuz ve dokulu bölgelerin ayrı ayrı değerlendirilebilir ve bu bölgelere uygun özellik çıkarım algoritması seçilebilir.

Videoda yapılan sahtecilikler hem çerçeveler arasında hem de çerçeve düzeyinde gerçekleştirilmektedir. Literatürde Çerçeve Tekrarlama sahteciliğinin tespiti için önerilmiş olan yöntemler çerçeveler arası yapılan tüm sahtecilikleri tespit edememekte sadece bu sahteciliği tespit edebilmektedir. Videonun çerçeve ve çerçeveler arası özellikleriyle birlikte videoya ait sıkıştırma özellikleri birlikte incelenerek önerilecek olan yöntemle çerçeve tekrarlama sahteciliği yanında video çerçeveleri arasında yapılan Çerçeve ekleme ve silme sahteciliklerini de tespit edebilir.

Videoda bölge tabanlı sahtecilik tespitinde ise eğer videolar statik kamera ile çekilmiş ve video çerçevelerine durağan bir nesne eklenmişse tek bir çerçeve üzerinde

bulanık kenar izlerinin araştırılması sahtecilik tespiti için yeterli olmaktadır. Ancak videonun hareketli bir kamera ile çekilmiş olması veya hareketli nesnelerin video çerçevelerine eklenmesi söz konusu ise sahtecilik tespiti için her bir çerçevenin teker teker incelenmesi gerekmektedir. Bu da çok büyük bir işlem zamanı demektir. Bunun için video öncelikle literatürdeki alt sahnelere ayrıştırma teknikleri incelenerek alt sahnelere ayrıştırılabilir ve her bir alt sahne için alt sahneleri temsil eden bir anahtar çerçeve çıkarılabilir. Böylece anahtar çerçevelerdeki sahtecilik izleri (bulanık kenarlar) araştırılarak, bulanık kenarların tespit edildiği anahtar çerçevelerin yer aldığı sahneler sahte olarak işaretlenebilir.



6. KAYNAKLAR

1. Shinder, D., L. ve Cross, M., Scene of the Cybercrime, Syngress, 2008.
2. Ng, T., T. ve Chang, S., F., A model for image splicing, In International conference on image processing, Singapore, 2004, 1169-1172.
3. Hsu, Y. F., ve Chang, S. F., "Camera response functions for image forensics: an automatic algorithm for splicing detection." IEEE Transactions on Information Forensics and Security 5,4 (2010) 816-825.
4. Zhao, X., Li, S., Wang, S., Li, J. ve Yang, K. "Optimal chroma-like channel design for passive color image splicing detection." EURASIP Journal on Advances in Signal Processing, 1 (2012) 240.
5. Zhang, Z., Zhou, Y., Kang, J. ve Ren, Y. Study of image splicing detection, Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues, Shanghai, China, 2008, 1103-1110.
6. Qu, Z., Qiu, G. ve Huang, J., Detect digital image splicing with visual cues, International workshop on information hiding, Springer, Berlin, Heidelberg, 2009, 247-261.
7. Li, X., Jing, T. ve Li, X., Image splicing detection based on moment features and Hilbert-Huang Transform, Information Theory and Information Security (ICITIS), IEEE International Conference on. IEEE, Beijing, China, 2010, 1127-1130.
8. Ryu, S. J., Lee, M. J. ve Lee, H. K., Detection of Copy-Rotate-Move Forgery Using Zernike Moments, Information Hiding, 2010, 51-65.
9. Liu, G., Wang, J., Lian, S. ve Wang, Z., A passive image authentication scheme for detecting region-duplication forgery with rotation, Journal of Network and Computer Applications, 34,5 (2011) 1557-1565.
10. Xu, B., Liu, G. ve Dai, Y., A fast image copy-move forgery detection method using faz correlation. Multimedia Information Networking and Security (MINES), Fourth International Conference on. IEEE, Los Alamitos, CA, USA, 2012, 319-322.
11. Qiao, M., Sung, A., Liu, Q., & Ribeiro, B., A novel approach for detection of copy-move forgery. Proceedings of Fifth International Conference on Advanced Engineering Computing and Applications in Sciences (ADVCOMP 2011), Lisbon, Portugal, 2011, 44-47.
12. Cao, Y., Gao, T., Fan, L. ve Yang, Q., A robust detection algorithm for copy-move forgery in digital images, Forensic Science International, 214,1 (2012) 33-43.

13. Li, G., Wu, Q., Tu, D. ve Sun, S., A sorted neighborhood approach for detecting duplicated regions in image forgeries based on DWT and SVD. Multimedia and Expo, IEEE International Conference on IEEE, Beijing, China, 2007, 1750-1753.
14. Bayram, S., Sencar, H. T. ve Memon, N., An efficient and robust method for detecting copy-move forgery, Acoustics, Speech and Signal Processing (ICASSP 2009) IEEE International Conference on. IEEE, Taipei, Taiwan, 2009, 1053-1056.
15. Bin, Y., A., N., G., Xingming, S., U., N., Xianyi, C., H., E., N., Zhang, J. ve Xu, L., I., An Efficient Forensic Method for Copy--move Forgery Detection Based on DWT-FWHT, Radioengineering, 22,4 (2013) 1098–1105.
16. Lin, H. J., Wang, C., W. ve Kao, Y., T., Fast copy-move forgery detection, WSEAS Transactions on Signal Processing, 5,5 (2009) 188-197.
17. Li, L., Li, S., Zhu, H., Chu, S. C., Roddick, J., F. ve Pan, J., S., An efficient scheme for detecting copy-move forged images by local binary patterns, Journal of Information Hiding and Multimedia Signal Processing, 4,1 (2013) 46-56.
18. Wang, T., Tang, J., Zhao, W., Xu, Q. ve Luo, B., Blind detection of copy-move forgery based on multi-scale autoconvolution invariants, Chinese Conference on Pattern Recognition (CCPR '12), Springer, Berlin, Heidelberg, 2012, 438-446.
19. Wang, T., Tang, J. ve Luo, B., Blind detection of region duplication forgery by merging blur and affine moment invariants, Image and Graphics (ICIG), Seventh International Conference (ICIG '13) on. IEEE, 2013, 258-264.
20. Shahid, T. ve Mansoor, A., B., Copy-move forgery detection algorithm for digital images and a new accuracy metric, International Journal of Recent Trends in Engineering and Technology, 2,2 (2009) 159–161.
21. Li, L., Li, S. ve Wang, J., Copy-move forgery detection based on PHT, Information and Communication Technologies (WICT), World Congress on. IEEE, 2012, 1061-1065.
22. Quan, X. ve Zhang, H., Copy-move forgery detection in digital images based on local dimension estimation, Cyber Security, Cyber Warfare and Digital Forensic (CyberSec 2012), International Conference on. IEEE, 2012, 180-185.
23. Davarzani, R., Yaghmaie, K., Mozaffari, S. ve Tapak, M., Copy-move forgery detection using multiresolution local binary patterns, Forensic Science International, 231,1 (2013) 61-72.
24. Ardizzone, E., Bruno, A. ve Mazzola, G., Copy-move forgery detection via texture description, Proceedings of the 2nd ACM workshop on Multimedia in forensics, security and intelligence (MiFor '10), ACM, 2010, 59-64.

25. Li, L., Li, S., Zhu, H. ve Wu, X., Detecting copy-move forgery under affine transforms for image forensics, Computers & Electrical Engineering, 40,6 (2014) 1951-1962.
26. Ghorbani, M., & Firouzmand, M., Detection of copy-move forgery using Haar-like features, International Journal of Computer Science, 2,1 (2007) 1-7.
27. Myna, A. N., Venkateshmurthy, M. G. ve Patil, C. G., Detection of region duplication forgery in digital images using dalgacık and log-polar mapping, Conference on Computational Intelligence and Multimedia Applications (ICCIMA '07). International Conference on. IEEE, Sivakasi, India, 2007, 371-377.
28. Wu, Y., Deng, Y., Duan, H. ve Zhou, L., Dual tree complex dalgacık transform approach to copy-rotate-move forgery detection, Science China Information Sciences, 57,1 (2014) 1-12.
29. Sunil, K., Jagan, D. ve Shaktidev, M., AKD-TBA based method for copy-move forgery detection, ICT and Critical Infrastructure: Proceedings of the 48th Annual Convention of Computer Society of India-Vol II, Springer, New York, NY, USA, 2014, 577-583.
30. Sadeghi, S., Jalab, H., A. ve Dadkhah, S., Efficient Copy-Move Forgery Detection for Digital Images, World Academy of Science, Engineering and Technology, 71 (2012).
31. Bashar, M., Noda, K., Ohnishi, N. ve Mori, K., Exploring duplicated regions in natural images, IEEE Transactions on Image Processing, 99 (2010) 1-40.
32. Singh, V., K. ve Tripathi, R., C., Fast and efficient region duplication detection in digital images using sub-blocking method, International Journal of Advanced Science and Technology, 35 (2011) 93-102.
33. Zhao, J. ve Guo, J., Passive forensics for copy-move image forgery using a method based on AKD and SVD, Forensic Science International, 233,1 (2013) 158-166.
34. Sheng, G., Gao, T., Cao, Y., Gao, L. ve Fan, L., Robust algorithm for detection of copy-move forgery in digital images based on ridgelet transform, Proceedings of the Artificial Intelligence and Computational Intelligence (AICI '12), Springer, 2012, 317-323.
35. Wang, Y., Gurule, K., Wise, J. ve Zheng, J., Dalgacık based region duplication forgery detection, Information Technology: New Generations (ITNG '12), Ninth International Conference on. IEEE, 2012, 30-35.
36. Bashar, M., K., Noda, K., Ohnishi, N., Kudo, H., Matsumoto, T. ve Takeuchi, Y., Dalgacık-Based Multiresolution Features for Detecting Duplications in Images, Proceedings of the 10th IAPR Conference on Machine Vision Applications (MVA '07), 2007, 264-267.

37. Yao, H., Qiao, T., Tang, Z., Zhao, Y. ve Mao, H., Detecting copy-move forgery using non-negative matrix factorization, Multimedia Information Networking and Security (MINES '11), Third International Conference on. IEEE, Shanghai, China, 2011, 591-594.
38. Wu, Q., Wang, S. ve Zhang, X., Detection of image region-duplication with rotation and scaling tolerance, Computational Collective Intelligence. Technologies and Applications, Springer, New York, NY, USA, 2010, 100-108.
39. Muhammad, G., Al-Hammadi, M. H., Hussain, M., Mirza, A. M. ve Bebis, G., Copy move image forgery detection method using steerable pyramid transform and texture descriptor, EUROCON 2013 IEEE, IEEE, Zagreb, Croatia, 2013, 1586-1592.
40. Lin, S., D. ve Wu, T., An integrated technique for splicing and copy-move forgery image detection, Image and Signal Processing (CISP '11), 4th International Congress on. IEEE, 2011, 1086-1090.
41. Hussain, M., Muhammad, G., Saleh, S., Q., Mirza, A., M. ve Bebis, G., Copy-move image forgery detection using multi-resolution Weber descriptors, Signal Image Technology and Internet Based Systems (SITIS '12), Eighth International Conference on. IEEE, 2012, 395-401.
42. Wandji, N., D., Xingming, S. ve Kue, M., F., Detection of copy-move forgery in digital images based on AKD, Journal of Computer Science, 10 (2013) 295-302.
43. Wandji, N., D. ve Xingming, S., Robust detection of copy-move forgery in color images, Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV '13). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), Las Vegas, Nev, USA, 2013, 492-495.
44. Bravo-Solorio, S. ve Nandi, A., K., Automated detection and localisation of duplicated regions affected by reflection, rotation and scaling in image forensics, Signal Processing, 91,8 (2011) 1759-1770.
45. Luo, W., Huang, J. ve Qiu, G., Robust detection of region-duplication forgery in digital image, Pattern Recognition (ICPR '06), 18th International Conference on. IEEE, Hong Kong, 2006, 746-749.
46. Kashyap, A. ve Joshi, S., D., Detection of copy-move forgery using dalgacık decomposition, Signal Processing and Communication (ICSC '13), International Conference on. IEEE, 2013, 396-400.
47. Zhang, J., Feng, Z. ve Su, Y., A new approach for detecting copy-move forgery in digital images, Communication Systems (ICCS '08), 11th IEEE Singapore International Conference on. IEEE, 2008, 362-366.

48. Khan, E., S. ve Kulkarni, E., A., An efficient method for detection of copy-move forgery using discrete dalgacık transform, International Journal on Computer Science and Engineering, 2,5 (2010) 1801–1806.
49. Ghorbani, M., Firouzmand, M. ve Faraahi, A., DWT-AKD (QCD) based copy-move image forgery detection, Systems, Signals and Image Processing (IWSSIP '11), 18th International Conference on. IEEE, 2011, 1-4.
50. Yong, L., Meishan, H. ve Bogang, L., Robust evidence detection of copy-rotate-move forgery in image based on singular value decomposition, Proceedings of the 14th international conference on Information and Communications Security (ICICS'12), Springer, New York, NY, USA, 2012, 357-364.
51. Dybala, B., Jennings, B. ve Letscher, D., Detecting filtered cloning in digital images, Proceedings of the 9th workshop on Multimedia & Security (MMand Sec '07), ACM, New York, NY, USA, 2007, 43-50.
52. Liu, G., Wang, J., Lian, S. ve Wang, Z., A passive image authentication scheme for detecting region-duplication forgery with rotation, Journal of Network and Computer Applications, 34,5 (2011) 1557-1565.
53. Zhong, L. ve Xu, W., A robust image copy-move forgery detection based on mixed moments, Software Engineering and Service Science (ICSESS '13), 4th IEEE International Conference on. IEEE, 2013, 381-384.
54. Wang, J., Liu, G., Li, H., Dai, Y. ve Wang, Z., Detection of image region duplication forgery using model with circle block, Multimedia Information Networking and Security (MINES '09), International Conference on. IEEE, 2009, 25-29.
55. Chaitawittanun, N., Detection of copy-move forgery by clustering technique, International Proceedings of Computer Science & Information Technology, 2012, 50:6.
56. Ulutas, G. ve Ulutas, M., Image forgery detection using color coherence vector, Electronics, Computer and Computation (ICECCO '13), International Conference on. IEEE, 2013, 107-110.
57. Kang, X. ve Wei, S., Identifying tampered regions using singular value decomposition in digital image forensics, Computer Science and Software Engineering (CSSE '08), International Conference on. IEEE, 2008, 926-930.
58. Wang, X., Zhang, X., Li, Z. ve Wang, S., A DWT-AKD based passive forensics method for copy-move attacks, Multimedia Information Networking and Security (MINES '11), Third International Conference on. IEEE, Shanghai, China, 2011, 304-308.
59. Fridrich, A., J., Soukal, B., D. ve Lukáš, A., J., Detection of copy-move forgery in digital images, Proceedings of Digital Forensic Research Workshop (DFRWS '03), 2003, 1-10.

60. Hu, J., Zhang, H., Gao, Q. ve Huang, H., An improved lexicographical sort algorithm of copy-move forgery detection, Networking and Distributed Computing (ICNDC '11), Second International Conference on. IEEE, Beijing, China, 2011, 23-27.
61. Sarode, T., K. ve Vaswani, N. Copy-move forgery detection using orthogonal dalgacık transforms, International Journal of Computer Applications, 88,8 (2014) 41-45.
62. Zhang, T. ve Wang, R., D., Copy-move forgery detection based on SVD in digital image, Image and Signal Processing, (CISP '09), 2nd International Congress on. IEEE, 2009, 1-5.
63. L. Kang and X. Cheng, "Copy-move forgery detection in digital image," in Proceedings of the 3rd International Congress on Image and Signal Processing (CISP '10), Yantai, China, 2010, 2419–2421.
64. Ketenci, S. ve Ulutas, G., Copy-move forgery detection in images via 2D-Fourier transform, Telecommunications and Signal Processing (TSP '13), 36th International Conference on. IEEE, Rome, Italy, 2013, 813-816.
65. Junhong, Z., Detection of copy-move forgery based on one improved LLE method, Advanced Computer Control (ICACC '10), 2nd International Conference on. IEEE, Shenyang, China, 2010, 547-550.
66. Hsu, H., C. ve Wang, M., S., Detection of copy-move forgery image using Gabor descriptor, Anti-Counterfeiting, Security and Identification (ASID '12), International Conference on. IEEE, 2012, 1-4.
67. Nguyen, H. C. ve Katzenbeisser, S., Detection of copy-move forgery in digital images using radon transformation and faz correlation, Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP '12), Eighth International Conference on. IEEE, 2012, 134-137.
68. Mahdian, B. ve Saic, S., Detection of copy-move forgery using a method based on blur moment invariants, Forensic Science International, 171,2 (2007) 180-189.
69. Imamoglu, M. B., Ulutas, G. ve Ulutas, M., Detection of copy-move forgery using krawtchouk moment, Electrical and Electronics Engineering (ELECO '13), 8th International Conference on. IEEE, 2013, 311-314.
70. Popescu, A. C. ve Farid, H., Exposing digital forgeries by detecting duplicated image regions, Department Computer Science, Dartmouth College, Technology Report TR2004-515, Hanover, Conn, USA, 2004, 1-11.
71. Li, Y., Image copy-move forgery detection based on polar cosine transform and approximate nearest neighbor searching, Forensic Science International, 224,1 (2013) 59-67.

72. Huang, Y., Lu, W., Sun, W. ve Long, D., Improved AKD-based detection of copy-move forgery in images, Forensic Science International, 206,1 (2011) 178-184.
73. Wu, Q., Wang, S. ve Zhang, X., Log-polar based scheme for revealing duplicated regions in digital images, IEEE Signal Processing Letters, 2011, 18,10 (2011) 559-562.
74. Bravo-Solorio, S. ve Nandi, A., K., Passive forensic method for detecting duplicated regions affected by reflection, rotation and scaling, Signal Processing Conference (EUSIPCO '09), 17th European. IEEE, Glasgow, Scotland, 2009, 824-828.
75. Gharibi, F., RavanJamjah, J., Akhlaghian, F., Azami, B., Z. ve Alirezaie, J., Robust detection of copy-move forgery using texture features, Electrical Engineering (ICEE '11), 19th Iranian Conference on. IEEE, 2011, 1-4.
76. Ryu, S., J., Kirchner, M., Lee, M., J. ve Lee, H., K., Rotation invariant localization of duplicated image regions based on Zernike moments, IEEE Transactions on Information Forensics and Security, 8,8 (2013) 1355-1370.
77. Li, W. ve Yu, N., Rotation robust detection of copy-move forgery, Image Processing (ICIP '10), 17th IEEE International Conference on. IEEE, 2010, 2113-2116.
78. Langille, A. ve Gong, M., An efficient match-based duplication detection algorithm, Computer and Robot Vision (CRV '06), The 3rd Canadian Conference on. IEEE, 2006, 1-8.
79. Gan, Y. ve Cang, J., A detection algorithm for image copy-move forgery based on improved circular projection matching and TBA, Sensors & Transducers, 2013, 159,11 (2013) 19-25.
80. Zimba, M. ve Xingming, S., Detection of image duplicated regions affected by rotation, scaling and translation using block characteristics of DWT coefficients, International Journal of Digital Content Technology and Its Applications, 5,11 (2011) 143-150.
81. Sekeh, M., A., Maarof, M., A., Rohani, M., F. ve Motiei, M., Sequential straightforward clustering for local image block matching, World Academy of Science, Engineering and Technology, 50 (2011) 774-778.
82. Christlein, V., Riess, C. ve Angelopoulou, E., On rotation invariance in copy-move forgery detection, Information Forensics and Security (WIFS '10), IEEE International Workshop on. IEEE, 2010, 1-6.
83. Muhammad, N., Hussain, M., Muhamad, G. ve Bebis, G., A non-intrusive method for copy-move forgery detection, Advances in Visual Computing, Springer, Berlin, Germany, 2011, 516-525.

84. Huang, H., Guo, W. ve Zhang, Y., Detection of copy-move forgery in digital images using SIFT algorithm, Computational Intelligence and Industrial Application, (PACIIA '08), Pacific-Asia Workshop on. IEEE, 2008, 272-276.
85. Ardizzone, E., Bruno, A. ve Mazzola, G., Detecting multiple copies in tampered images, Image Processing (ICIP '10), 17th IEEE International Conference on. IEEE, 2010, 2117-2120.
86. Amerini, I., Ballan, L., Caldelli, R., Del Bimbo, A. ve Serra, G., A SIFT-based forensic method for copy-move attack detection and transformation recovery, IEEE Transactions on Information Forensics and Security, 6,3 (2011) 1099-1110.
87. Pan, X. ve Lyu, S., Detecting image region duplication using SIFT features, Acoustics Speech and Signal Processing (ICASSP '10), IEEE International Conference on. IEEE, 2010, 1706-1709.
88. Pan, X. ve Lyu, S., Region duplication detection using image feature matching, IEEE Transactions on Information Forensics and Security, 5,4 (2010) 857-867.
89. Zhang, C., Guo, X. ve Cao, X., Duplication localization and segmentation, Pacific-Rim Conference on Multimedia. Springer, Berlin, Heidelberg, 2010, 578-589.
90. Shivakumar, B., L. ve Baboo, S., S., Automated forensic method for copy-move forgery detection based on Harris interest points and SIFT descriptors, International Journal of Computer Applications, 27,3 (2011) 9-17.
91. Zhu, Y., Shen, X. ve Chen, H., Copy-move forgery detection based on scaled ORB, Multimedia Tools and Applications, 75,6 (2016) 3221-3233.
92. Bo, X., Junwen, W., Guangjie, L. ve Yuewei, D., Image copy-move forgery detection based on SURF, Multimedia information networking and security (MINES '10), international conference on. IEEE, 2010, 889-892.
93. Shivakumar, B., L. ve Baboo, S., Detection of region duplication forgery in digital images using SURF, International Journal Of Computer Science Issues, 8,4 (2011) 199-205.
94. Li, J., Li, X., Yang, B. ve Sun, X., Segmentation-based image copy-move forgery detection scheme, IEEE Transactions on Information Forensics and Security, 10,3 (2015) 507-518.
95. Pun, C., M., Yuan, X., C. ve Bi, X., L., Image forgery detection using adaptive oversegmentation and feature point matching, IEEE Transactions on Information Forensics and Security, 10,8 (2015)1705-1716.
96. Zandi, M., Mahmoudi-Aznaveh, A. ve Talebpour, A., Iterative copy-move forgery detection based on a new interest point detector, IEEE Transactions on Information Forensics and Security, 11,11 (2016) 2499-2512.

97. Yang, F., Li, J., Lu, W. ve Weng, J., Copy-move forgery detection based on hybrid features, Engineering Applications of Artificial Intelligence, 59 (2017) 73-83.
98. Lowe, D., G., Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision, 60,2 (2004) 91-110.
99. Bay, H., Ess, A., Tuytelaars, T. ve Van Gool, L., Speeded-up robust features (SURF). Computer Vision and Image Understanding, 110,3 (2008) 346-359.
100. Halsall, F., Multimedia communications: applications, networks, protocols, and standards. Pearson education, 2001.
101. Wang, W. ve Farid, H., Exposing digital forgeries in video by detecting double MPEG compression, Proceedings of the 8th workshop on Multimedia and security. ACM, 2006, 37-47.
102. Wang, W. ve Farid, H., Exposing digital forgeries in video by detecting double quantization, Proceedings of the 11th ACM workshop on Multimedia and security. ACM, 2009, 39-48.
103. Su, Y. ve Xu, J., Detection of double-compression in MPEG-2 videos, Intelligent Systems and Applications (ISA '10), 2nd International Workshop on. IEEE, 2010, 1-4.
104. Xu, J., Su, Y. ve You, X., Detection of video transcoding for digital forensics, Audio, Language and Image Processing (ICALIP '12), International Conference on. IEEE, 2012, 160-164.
105. Xu, J., Su, Y. ve Liu, Q., Detection of double MPEG-2 compression based on distributions of AKD coefficients, International Journal of Pattern Recognition and Artificial Intelligence, 27,1 (2013) 1354001.
106. Sun, T., Wang, W. ve Jiang, X., Exposing video forgeries by detecting MPEG double compression. In: Acoustics, Speech and Signal Processing (ICASSP '12), IEEE International Conference on. IEEE, 2012, 1389-1392.
107. Milani, S., Bestagini, P., Tagliasacchi, M. ve Tubaro, S., Multiple compression detection for video sequences, Multimedia Signal Processing (MMSP '12), IEEE 14th International Workshop on. IEEE, 2012, 112-117.
108. Vazquez-Padin, D., Fontani, M., Bianchi, T., Comesaña, P., Piva, A. ve Barni, M., Detection of video double encoding with GOP size estimation, Information Forensics and Security (WIFS '12), IEEE International Workshop on. IEEE, 2012, 151-156.
109. Jiang, X., Wang, W., Sun, T., Shi, Y., Q. ve Wang, S., Detection of double compression in MPEG-4 videos based on Markov statistics, IEEE Signal Processing Letters, 20,5 (2013) 447-450.

110. Ravi, H., Subramanyam, A., V., Gupta, G. ve Kumar, B., A., Compression noise based video forgery detection, Image Processing (ICIP '14), IEEE International Conference on. IEEE, 2014, 5352-5356.
111. Subramanyam, A., V. ve Emmanuel, S., Pixel estimation based video forgery detection, Acoustics, Speech and Signal Processing (ICASSP '13), IEEE International Conference on. IEEE, 2013, 3038-3042.
112. Luo, W., Wu, M. ve Huang, J., MPEG recompression detection based on block artifacts Security, Forensics, Steganography, and Watermarking of Multimedia Contents. 2008, 68190X.
113. Bestagini, P., Allam, A., Milani, S., Tagliasacchi, M. ve Tubaro, S., Video codec identification, Acoustics, Speech and Signal Processing (ICASSP '12), IEEE International Conference on. IEEE, 2012, 2257-2260.
114. Huang, Z., Huang, F. ve Huang, J., HUANG, Jiwu. Detection of double compression with the same bit rate in MPEG-2 videos. Signal and Information Processing (ChinaSIP '14), IEEE China Summit & International Conference on. IEEE, 2014, 306-309.
115. Tagliasacchi, M. ve Tubaro, S., Blind estimation of the QP parameter in H. 264/AVC decoded video. Image Analysis for Multimedia Interactive Services (WIAMIS '10), 11th International Workshop on. IEEE, 2010, 1-4.
116. Li, H. ve Forchhammer, S., MPEG2 video parameter and no reference PSNR estimation, Picture Coding Symposium, (PCS '09), IEEE, 2009, 1-4.
117. Valenzise, G., Tagliasacchi, M. ve Tubaro, S., Estimating QP and motion vectors in H. 264/AVC video from decoded pixels, Proceedings of the 2nd ACM workshop on Multimedia in forensics, security and intelligence. ACM, 2010, 89-92.
118. Yammine, G., Wige, E. ve Kaup, A., Blind GOP structure analysis of MPEG-2 and H. 264/AVC decoded video, Picture Coding Symposium (PCS '10), IEEE, 2010, 258-261.
119. Wang, W. ve Farid, H., Exposing digital forgeries in video by detecting duplication, Proceedings of the 9th workshop on Multimedia & Security, ACM, 2007, 35-42.
120. Lin, G., S. ve Chang, J., F., Detection of frame duplication forgery in videos based on spatial and temporal analysis, International Journal of Pattern Recognition and Artificial Intelligence, 26,07 (2012) 1250017.
121. Yang, J., Huang, T. ve Su, L., Using similarity analysis to detect frame duplication forgery in videos, Multimedia Tools and Applications, 75,4 (2016) 1793-1811.
122. Singh, V., K., Pant, P. ve Tripathi, R., C., Detection of frame duplication type of forgery in digital video using sub-block based features, International Conference on Digital Forensics and Cyber Crime, Springer, Cham, 2015, 29-38.

123. Zhang, Z., Hou, J., Ma, Q. ve Li, Z., Efficient video frame insertion and deletion detection based on inconsistency of correlations between local binary pattern coded frames, Security and Communication Networks, 8,2 (2015) 311-320.
124. Zheng, L., Sun, T. ve Shi, Y., Q., Inter-frame video forgery detection based on block-wise brightness variance descriptor, International Workshop on Digital Watermarking, Springer, Cham, 2014, 18-30.
125. Aghamaleki, J., A. ve Behrad, A., Inter-frame video forgery detection and localization using intrinsic effects of double compression on quantization errors of video coding, Signal Processing: Image Communication, 47 (2016) 289-302.
126. Liu, Y. ve Huang, T., Exposing video inter-frame forgery by Zernike opponent chromaticity moments and coarseness analysis, Multimedia Systems, 23,2 (2017) 223-238.
127. Wang, W. ve Farid, H., Exposing digital forgeries in interlaced and deinterlaced video, IEEE Transactions on Information Forensics and Security, 2,3 (2007) 438-449.
128. Hsu, C., C., Hung, T., Y., Lin, C., W. ve Hsu, C., T., Video forgery detection using correlation of noise residue, Multimedia Signal Processing, IEEE 10th Workshop on. IEEE, 2008, 170-174.
129. Li, L., Wang, X., Zhang, W., Yang, G. ve Hu, G., Detecting removed object from video with stationary background, The International Workshop on Digital Forensics and Watermarking, Springer, Berlin, Heidelberg, 2013, 242-252.
130. Chen, S., Tan, S., Li, B. ve Huang, J., Automatic detection of object-based forgery in advanced video, IEEE Transactions on Circuits and Systems for Video Technology, 26,11 (2016) 2138-2151.
131. Zhang, J., Su, Y. ve Zhang, M., Exposing digital video forgery by ghost shadow artifact, Proceedings of the First ACM workshop on Multimedia in forensics, ACM, 2009, 49-54.
132. Kobayashi, M., Okabe, T. ve Sato, Y., Detecting forgery from static-scene video based on inconsistency in noise level functions, IEEE Transactions on Information Forensics and Security, 5,4 (2010) 883-892.
133. Chetty, G., Biswas, M. ve Singh, R., Digital video tamper detection based on multimodal fusion of residue features, Network and System Security (NSS '10), 4th International Conference on. IEEE, 2010, 606-613.
134. Goodwin, J. ve Chetty, G., Blind video tamper detection based on fusion of source features, Digital Image Computing Techniques and Applications (DICTA '11), International Conference on. IEEE, 2011, 608-613.

135. Subramanyam, A., V. ve Emmanuel, S. Video forgery detection using HOG features and compression properties, Multimedia Signal Processing (MMSP '12), IEEE 14th International Workshop on. IEEE, 2012, 89-94.
136. Lin, C., S. ve Tsay, J., J., A passive approach for effective detection and localization of region-level video forgery with spatio-temporal coherence analysis, Digital Investigation, 11,2 (2014) 120-140.
137. Pandey, R., C., Singh, S., K. ve Shukla, K., K., Passive copy-move forgery detection in videos, Computer and Communication Technology (ICCCT '14), 2014 International Conference on. IEEE, 2014, 301-306.
138. Su, L., Huang, T. ve Yang, J., A video forgery detection algorithm based on compressive sensing, Multimedia Tools and Applications, 74,17 (2015) 6641-6656.
139. D'Amiano, L., Cozzolino, D., Poggi, G. ve Verdoliva, L., Video forgery detection and localization based on 3D patchmatch, Multimedia & Expo Workshops (ICMEW '15), IEEE International Conference on. IEEE, 2015, 1-6.
140. Cozzolino, D., Poggi, G. ve Verdoliva, L., Copy-move forgery detection based on patchmatch, Image Processing (ICIP '14), IEEE International Conference on. IEEE, 2014, 5312-5316.
141. Bidokhti, A., & Ghaemmaghami, S., Detection of regional copy/move forgery in MPEG videos using optical flow, Artificial intelligence and signal processing (AISP '15), International symposium on. IEEE, 2015, 13-17.
142. Ustubioglu, B., Ulutas, G., Ulutas, M. ve Nabiyeve, V., V., Improved copy-move forgery detection based on the CLDs and colour moments, The Imaging Science Journal, 64,4 (2016) 215-225.
143. Stricker, M., A. ve Orengo, M., Similarity of Color Images, Storage and Retrieval for Image and Video Databases (SPIE '95), , San Jose, USA, 1995, 381-392.
144. Ustubioglu, B., Ulutas, G., Ulutas, M. ve Nabiyeve, V., V., A new copy move forgery detection technique with automatic threshold determination, AEU-International Journal of Electronics and Communications, 70,8 (2016) 1076-1087.
145. Bracamonte, J., Ansorge, M., Pellandini, F. ve Farine, P., A., Low complexity image matching in the compressed domain by using the AKD-faz, Proceedings of the 6th COST 276 Workshop on Information and Knowledge Management for Integrated Media Communication. COST (European Cooperation in the field of Scientific and Technical Research), 2004, 88-93.
146. Ayyalasomayajula, P., Grassi, S. ve Farine, P., A., Retrieval of occluded images using AKD faz and region merging, Image Processing (ICIP '12), 19th IEEE International Conference on. IEEE, 2012, 2441-2444.

147. Fu, D., Shi, Y. Q. ve Su, W., A generalized Benford's law for JPEG coefficients and its applications in image forensics, Security, Steganography, and Watermarking of Multimedia Contents. 2007, 65051L.
148. Ustubioglu, B., Ulutas, G., Ulutas, M., Nabiyeve, V. ve Ustubioglu, A., LBP-AKD based copy move forgery detection algorithm, 30th International Symposium on Computer and Information Sciences, Springer, Londra, İngiltere, 2015, 127-136.
149. Üstübioğlu B., Baykal E., Muzaffer G. ve Ulutaş G., Blur Invariant Image Forgery Detection Method Using Local Faz Quantization, Journal Of Energy And Power Enginnering, 2016, 358-363.
150. Ojansivu, V. ve Heikkilä, J., Blur insensitive texture classification using local faz quantization, International Conference On Image and Signal Processing, Springer, Berlin, Heidelberg, 2008, 236-243.
151. Üstübioğlu, B., Ayas, S., Doğan, H. ve Ulutaş, G., Image forgery detection based on Colour SIFT, Signal Processing and Communications Applications Conference (SIU '15), 23th. IEEE, Malatya, Türkiye, 2015, 1741-1744.
152. Ustubioglu, B., Muzaffer, G., Ulutas, G., Nabiyeve, V. ve Ulutas, M., A novel keypoint based forgery detection method based on local faz quantization and SIFT, Electrical and Electronics Engineering (ELECO '15), 9th International Conference on. IEEE, Bursa, Türkiye, 2015, 185-189.
153. Ulutas, G., Ustubioglu, A., Ustubioglu, B., Nabiyeve, V., V. ve Ulutas, M., Medical Image Tamper Detection Based on Passive Image Authentication, Journal of Digital Imaging, (2017) 1-15.
154. Ojala, T., Pietikainen, M. ve Maenpaa, T., Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Transactions on pattern analysis and machine intelligence, 24,7 (2002) 971-987.
155. Yang, X. ve Cheng, K., T., Local difference binary for ultrafast and distinctive feature description, IEEE transactions on pattern analysis and machine intelligence, 36,1 (2014) 188-194.
156. Calonder, M., Lepetit, V., Strecha, C. ve Fua, P., Brief: Binary robust independent elementary features, European Conference on Computer Vision (ECCV '10), Heraklion, Greece, 2010, 778-792.
157. Rublee, E., Rabaud, V., Konolige, K. ve Bradski, G., ORB: An efficient alternative to SIFT or SURF, Computer Vision (ICCV '11), IEEE international conference on. IEEE, 2011, 2564-2571.
158. Leutenegger, S., Chli, M. ve Siegwart, R., Y., BRISK: Binary robust invariant scalable keypoints, Computer Vision (ICCV '11), IEEE International Conference on. IEEE, Barcelona, Spain, 2011, 2548-2555.

159. Alahi, A., Ortiz, R. ve Vanderghenst, P., Freak: Fast retina keypoint, Computer vision and pattern recognition (CVPR '12), IEEE conference on. IEEE, 2012, 510-517.
160. Ulutas, G., Ustubioglu, B., Ulutas, M. ve Nabiyev, V., Frame duplication/mirroring detection method with binary features, IET Image Processing, 11,5 (2017) 333-342.
161. Bosch, A., Muñoz, X. ve Martí, R., Which is the best way to organize/classify images by content?, Image and Vision Computing, 25,6 (2007) 778-791.
162. Tsai, C., F., Bag-of-words representation in image annotation: A review, ISRN Artificial Intelligence, (2012).
163. Sivic, J. ve Zisserman, A., Video Google: A text retrieval approach to object matching in videos, Proceedings of the 9th IEEE International Conference on Computer Vision (ICCV '03), IEEE, 2003, 1470-1477.
164. Ballan, L., Bertini, M., Del Bimbo, A. ve Serra, G., Video event classification using bag of words and string kernels, Image Analysis and Processing (ICIAP '09), 2009, 170-178.
165. Yang, J., Jiang, Y., G., Hauptmann, A., G. ve Ngo, C., W., Evaluating bag-of-visual-words representations in scene classification. Proceedings of The International Workshop On Workshop On Multimedia Information Retrieval, ACM, 2007, 197-206.
166. Mironică, I., Duță, I., Ionescu, B. ve Sebe, N. Beyond bag-of-words: Fast video classification with fisher kernel vector of locally aggregated descriptors, Multimedia and Expo (ICME '15), IEEE International Conference on. IEEE, 2015, 1-6.
167. Elshourbagy, M., Hemayed, E. ve Fayek, M., Enhanced bag of words using multilevel k-means for human activity recognition, Egyptian Informatics Journal, 17,2 (2016) 227-237.
168. Shukla, P., Biswas, K., K. ve Kalra, P., K., Action Recognition using Temporal Bag-of-Words from Depth Maps. International Conference on Machine Vision Applications (MVA '13). 2013, 41-44.
169. Iosifidis, A., Tefas, A. ve Pitas, I., Merging linear discriminant analysis with bag of words model for human action recognition, Image Processing (ICIP '15), IEEE International Conference on. IEEE, Quebec City, QC, 2015, 832-836.
170. VLFeat open source library, available at: <http://www.vlfeat.org/>, 13 Kasım 2015.
171. Visalakshi, N., K. ve Thangavel, K., Impact of normalization in distributed k-means clustering. International Journal of Soft Computing, 4,4 (2009) 168-172.
172. K means ++, available at: <https://en.wikipedia.org/wiki/K-means%2B%2B>, 13 Kasım 2015.

173. Arthur, D. ve Vassilvitskii, S., k-means++: The advantages of careful seeding, Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms. Society for Industrial and Applied Mathematics, 2007, 1027-1035.
174. Liu, G., Wang, J., Lian, S. ve Dai, Y., Detect image splicing with artificial blurred boundary, Mathematical and Computer Modelling, 57,11 (2013) 2647-2659.
175. Do, M., N. ve Vetterli, M., The contourlet transform: an efficient directional multiresolution image representation, IEEE Transactions on image processing, 14,12 (2005) 2091-2106.
176. da Cunha, A. L., Zhou, J. ve Do, M., N., Nonsubsampled contourlet transform: filter design and applications in denoising, Image Processing, (ICIP '05). IEEE International Conference on. IEEE, 2005, 749–752.
177. CoMoFoD database, available at: <http://www.vcl.fer.hr/comofod/>, 13 Kasım 2013
178. Kodak Lossless True Color Image Suite. <http://r0k.us/graphics/kodak/>, 13 Kasım 2013
179. Google Image Search: <http://images.google.com/>, 13 Kasım 2013
180. Barre S 1999 Available at: <http://www.barre.nom.fr/medical/samples/>, 13 Kasım 2014
181. Su, L. ve Li, C., A novel passive forgery detection algorithm for video region duplication, Multidimensional Systems and Signal Processing, (2017) 1-18.

ÖZGEÇMİŞ

Beste ÜSTÜBİOĞLU; 1988 yılında Trabzon'da doğdu. İlk ve orta öğrenimini Trabzon'da tamamladı. 2006 yılında Kanuni Anadolu Lisesi'nden mezun oldu. 2006 yılında, OSYS ile yerleştirildiği Kocaeli Üniversitesinin Bilgisayar Mühendisliği Bölümü'nden 2007 yılında yatay geçişle Karadeniz Teknik Üniversitesi Bilgisayar Mühendisliği'ne geçiş yaptı. 2010 yılında lisans derecesini aldığı KTÜ Bilgisayar Mühendisliği bölümünü birincilik derecesi ile bitirdi. Aynı yıl KTÜ Bilgisayar Mühendisliği Bölümünde Fen Bilimlerine bağlı Araştırma Görevlisi olarak atandı. 2013 yılında aynı bölümde yüksek lisansını tamamladı. Aynı yıl Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı'nda doktora çalışmalarına başladı. KTÜ Bilgisayar Mühendisliği Bölümünde halen çalışmaya devam etmektedir.

Doktora çalışması esnasında, TÜBİTAK destekli 115E214 numaralı Pasif Video Doğrulama sistemi adlı projede bursiyer statüsünde görev aldı. 5'i SCI/expanded indeksli dergilerde, 1' i diğer dergilerde, 8'i uluslararası konferanslarda, 4'ü ulusal konferanslarda olmak üzere toplam 18 yayını mevcuttur. Yabancı dil olarak İngilizce bilmektedir. Başlıca yayınları aşağıda verilmiştir.

SCI, SSCI, AHCI indekslerine giren dergilerde yayınlanan makaleler

Üstübioğlu B., Ulutaş G., Ulutaş M., Nabiye V., "Improved copy-move forgery detection based on the CLDs and colour moments", IMAGING SCIENCE JOURNAL, vol.64, pp.215-225, 2016.

Üstübioğlu B., Ulutaş G., Ulutaş M., Nabiye V., "A new copy move forgery detection technique with automatic threshold determination", AEU-INTERNATIONAL JOURNAL OF ELECTRONICS AND COMMUNICATIONS, vol.70, pp.1076-1087, 2016.

Ulutaş G., **Üstübioğlu B.**, Ulutaş M., Nabiye V., "Frame duplication/mirroring detection method with binary features", IET IMAGE PROCESSING, vol.11, pp.333-342, 2017.

Ulutaş G., Üstübioğlu A., **Üstübioğlu B.**, Nabiye V., Ulutaş M., "Medical Image Tamper Detection based on Passive Image Authentication", JOURNAL OF DIGITAL IMAGING, vol.10, pp.1-15, 2017.

Ulutaş G., **Üstübioğlu B.**, Ulutaş M., Nabiye V., "Frame duplication detection based on BoW model", MULTIMEDIA SYSTEMS, pp.1-19, 2017.