T.C.

FIRAT ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

SAYISAL FİLİGRANLAR İLE KRİPTO İMZALARININ BİRLİKTE KULLANILMASI ve ÇOKLU ORTAM VERİSİ ÜZERİNDEKİ UYGULAMALARI

Metin ERTÜRKLER

Tez Yöneticisi

Yrd. Doç. Dr. Yetkin TATAR

DOKTORA TEZİ

ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

ELAZIĞ, 2007

T.C.

FIRAT ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

SAYISAL FİLİGRANLAR İLE KRİPTO İMZALARININ BİRLİKTE KULLANILMASI ve ÇOKLU ORTAM VERİSİ ÜZERİNDEKİ UYGULAMALARI

Metin ERTÜRKLER

DOKTORA TEZİ

ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

Bu tez, ... tarihinde aşağıda belirtilen jüri tarafından oybirliği /oyçokluğu ile başarılı / başarısız olarak değerlendirilmiştir.

Danışman: Yrd. Doç. Dr. Yetkin TATAR

Üye:

Üye:

Üye:

Üye:

Bu tezin kabulü, Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ...../...../..... tarih ve ..... sayılı kararıyla onaylanmıştır.

TEŞEKKÜR

Tez çalışmam boyunca ilgi ve yardımlarını esirgemeyen, gerek akademik hayattaki çalışmalarıyla gerekse hayata bakış açısıyla bana yön gösteren değerli danışman hocam Yrd. Doç Dr. Yetkin TATAR'a en içten ve samimi duygularımla teşekkür ederim.

Ayrıca çalışmalarım sırasında bilgi birikimlerinden ve rehberliklerinden faydalandığım değerli dostlarım Yrd. Doç Dr. Burhan ERGEN ile Yrd. Doç. Dr. Ahmet ÇINAR'a ve bana emeği geçen tüm hocalarıma teşekkür ederim.

Son olarak doktora tez çalışmamı, tükenmez sabrı ile sıkıntılarımı paylaşan ve motivasyonumu sağlayarak sürekli destek olan değerli eşim Serpil ERTÜRKLER'e; "babam yine okulda mı?" diye soran, bana her gün ilginç ve yeni bir şeyler öğreten kızım Gözde Ceren'e ithaf etmek istiyorum.

İÇİNDEKİLER

TEŞEKKÜR

İÇİNDEKİLER I

ŞEKİLLER LİSTESİ V

TABLOLAR LİSTESİ XII

EKLER LİSTESİ XIII

SİMGELER LİSTESİ XIV

KISALTMALAR LİSTESİ XVI

ÖZET XVII

ABSTRACT XVIII

1. GİRİŞ…….……..……... 1

1.1. Veri Doğrulama…….……...2
1.2. Sayısal Filigranlar Kullanılarak İmgelerin Doğrulanması..……...……..4
1.3. Kırılgan Filigranlama Teknikleri……..……...…. 5
1.4. Tezin Amacı..……...… 7
1.5. Tezin Yapısı..……... 10

2. SAYISAL FİLİGRANLAMA, NİTELİKLERİ ve UYGULAMA ALANLARI……. 11

2.1. Sayısal Filigranlama……11
2.1.1. Sayısal Filigranlama, Gizliyazı ve Bilgi Gizleme……12
2.2. Sayısal Filigranlama Uygulamaları……..14
2.2.1. Telif Hakkı'nın Teşhis Edilmesi ve Kanıtlanması……...15
2.2.2. Yayın İzleme…….…… 15
2.2.3. Kopya Koruma…….16
2.2.4. İçeriğin Doğrulanması……..17
2.2.5. İçerik İzleme…… 18
2.2.6. İçerik Arşivleme…….....19

2.3. Sayısal Filigranların Nitelikleri……19
2.3.1. Gürbüzlük……19
2.3.1.1. İmge Sıkıştırma……20
2.3.1.2. Baskılama ve Tarama……20
2.3.1.3. İmge İyileştirme……20
2.3.1.4. Geometrik Bozulmalar……21
2.3.2. Aslına Benzerlik……21
2.3.3. Veri Yükü ve Kapasite……22
2.3.4. Gömü Etkinliği……22
2.3.5. Yanlış Pozitif, Yanlış Negatif ve Bit Hata Oranı……22
2.3.6. Güvenlik……23
2.3.7. Bilgili veya Bilgisiz Buluculu Sistemler……24
2.3.8. Kriptolama ve Filigranlama Anahtarları……24
2.3.9. Maliyet……25

3. SAYISAL FİLİGRANLAMA SİSTEMİ……26

3.1. Sayısal İmge……26
3.1.1. İmgenin Algılanması ve Elde Edilmesi ……27
3.1.2. Sayısal İmgenin Oluşturulması……27
3.2. Filigranın Oluşturulması……29
3.3. Filigranın Gömülmesi……31
3.4. Filigranın Bulunması……34
3.5. Sayısal Filigranlama Teknikleri……39
3.5.1. Nicemlemeli Filigranlama Teknikleri……42
3.5.2. Yayımlı Spektrum Filigranlama Teknikleri……43
3.5.3. Uyarlamalı Filigranlama Teknikleri……44
3.5.4. EÖB Düzlemini Kullanan Filigranlama Teknikleri ……44
3.5.5. Görünür Filigranlama Teknikleri……45
3.5.6. Gürbüz, Kırılgan ve Yarı Kırılgan Filigranlama Teknikleri……46
3.6. Saldırılar……46
3.6.1. Tip1-A Saldırıları……47
3.6.1.1. Silen Saldırılar……47
3.6.1.2. Senkronizasyon Saldırıları……50
3.6.1.3. Belirsizlik Oluşturan Saldırılar……51

3.6.2. Tip1-B Saldırıları……. 51
3.6.3. Tip2 Saldırıları……. 51
3.6.3.1. Yazılım Değiştirme Saldırıları……. 51
3.6.3.2. Donanımsal Değiştirme Saldırıları…….....52

4. KIRILGAN FİLİGRANLAMA TEKNİKLERİ…….. 53

4.1. Kriptografi……56
4.1.1. Simetrik Anahtarlı Kriptografi……57
4.1.2. Asimetrik Anahtarlı Kriptografi……...58
4.1.3. Tek Yönlü Özüt Fonksiyonları…… 59
4.1.4. Mesaj Doğrulama Kodları……59
4.1.5. Sayısal İmzalar……60
4.2. Wong'un Tekniği……. 60
4.3. Kırılgan Filigranlama Saldırıları……..63
4.3.1. Kes-Yapıştır Saldırısı……...63
4.3.2. Vektör Nicemlemeli Taklit Saldırısı…… 65
4.3.3. Doğum Günü Saldırısı…… 67
4.3.4. Aktarıma Dayalı Saldırı……... 68
4.4. Saldırılara Karşı Alınan Önlemler……... 69
4.4.1. Blok Boyutlarını Büyütme……... 69
4.4.2. Sayısal İmza İçerisine Blok İndisi Ekleme…….. 70
4.4.3. Sayısal İmza İçerisine İmge İndisi Ekleme…… 70
4.4.4. Bağımlı Blok Yapısının Kullanılması……..71
4.4.5. Hash Block Chaining Version 1……...…72
4.4.6. Hash Block Chaining Version 2……...73
4.4.7. Sıradüzensel Filigranlama Tekniği…….. 73

5. SIRADÜZENSEL AĞAÇ YAPILI KIRILGAN FİLİGRANLAMA TEKNİĞİ…… 77

5.1. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniğinin Filigranlama Yordamı..79
5.2. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniğinin Doğrulama Yordamı… 83
5.3. Deneysel Çalışmalar……...…… 85
5.3.1. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniğinin Değişim Bölgesi Belirleme Doğruluğu……86

5.3.2. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniğinin Saldırılara Karşı Güvenirliği……... 97
5.3.2.1. Kes Yapıştır Saldırısına Karşı Güvenirlik……... 97
5.3.2.2. Vektör Nicemlemeli Taklit Saldırısına Karşı Güvenirlik…… 98
5.3.2.3. Doğum Günü Saldırısına Karşı Güvenirlik……..99
5.3.2.4. Aktarıma Dayalı Saldırıya Karşı Güvenirlik……... 100
5.3.3. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniği ile Sıradüzensel Filigranlama Tekniğinin Karşılaştırılması……... 100
5.3.4. Deneysel Sonuçların Değerlendirilmesi……...117

6. ALT BLOK YAKLAŞIMLI KIRILGAN FİLİGRANLAMA TEKNİĞİ…… 119

6.1. Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinin Filigranlama Yordamı…….. 120
6.2. Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinin Doğrulama Yordamı…….126
6.3. Deneysel Çalışmalar……...……. 130
6.3.1. Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinin Değişim Bölgesi Belirleme Doğruluğu…… 138
6.3.2. Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinin Saldırılara Karşı Güvenirliği……... 145
6.3.2.1. Kes Yapıştır Saldırısına Karşı Güvenirlik……... 145
6.3.2.2. Vektör Nicemlemeli Taklit Saldırısına Karşı Güvenirlik…… 147
6.3.2.3. Doğum Günü Saldırısına Karşı Güvenirlik……..148
6.3.2.4. Aktarıma Dayalı Saldırıya Karşı Güvenirlik……... 148
6.3.3. Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniği ile HBC2 Tekniğinin Karşılaştırılması……..……. 149
6.3.4. Deneysel Sonuçların Değerlendirilmesi ……...161

7. SONUÇLAR…… 162

KAYNAKLAR……. 165
ÖZGEÇMİŞ…….. 179
EKLER……...180

ŞEKİLLER LİSTESİ

Şekil 2.1. Banknot üzerindeki filigranlama örneği…… 11
Şekil 2.2. Gizliyazısal mesaj…….....13
Şekil 2.3. Telif hakkı uyarısının silinmesi **(a)** Özgün imge **(b)** Değiştirilmiş imge ……. 15
Şekil 2.4. İçeriğin doğrulanması **(a)** Özgün imge **(b)** Değiştirilmiş imge…….....17
Şekil 3.1. Sayısal Filigranlama Sistemi'nin genel yapısı……..... 26
Şekil 3.2. Bir imgenin oluşması…….28
Şekil 3.3. İmgenin sayısallaştırılması…… 28
Şekil 3.4. 8*8 piksellik bir imge ve onun sayısal değerleri…….. 29
Şekil 3.5. Farklı çekirdek değerleri ile üretilmiş sözde rastlantısal gürültü modelleri **(a)** çekirdek=3 **(b)** çekirdek=4 **(c)** çekirdek=5 **(d)** çekirdek=6…….. 30
Şekil 3.6. İmgesel logolar **(a)** Gri ölçekli **(b)** Siyah-beyaz……31
Şekil 3.7. Sözde rastlantısal gürültü modelinden oluşan bir filigranın eklenerek gömülmesi……. 32
Şekil 3.8. İkili bir imgesel filigranın imgenin EÖB düzlemine gömülmesi……..... 33
Şekil 3.9. Nicemlemeli gömme……..... 34
Şekil 3.10. Filigranın gömülmesi…….37
Şekil 3.11. Çekirdek değeri 0-10 arasında değişen modeller için elde edilen korelasyon değerleri……..... 37
Şekil 3.12. Sayısal Filigranlama Sistemleri'nin sınıflandırılması……... 41
Şekil 3.13. Costa tarafından çalışılan kanal……42
Şekil 3.14. Görünür Filigranlama…… 45
Şekil 3.15. Saldırıların sınıflandırılması……..... 48
Şekil 4.1. Kriptolama ve kripto çözme işlemi……... 57
Şekil 4.2. Simetrik anahtarlı kriptografi…… 58
Şekil 4.3. Asimetrik anahtarlı kriptografi……..... 59
Şekil 4.4. Logo imgesinden imge ile aynı boyutlu ikili filigran imgesinin oluşturulması……. 61
Şekil 4.5. Wong'un tekniğinin filigranlama yordamı..……..……... 61
Şekil 4.6. Wong'un tekniğinin doğrulama yordamı……..... 62
Şekil 4.7. Wong'un Tekniği ile filigranlama **(a)** Özgün imge **(b)** Filigranlı imge……64
Şekil 4.8. Kes yapıştır saldırısı **(a)** Saldırı yapılan bölge **(b)** Saldırı yapılmış Filigranlı imge …….. 64

Şekil 4.9. Vektör Nicemlemeli Taklit Saldırısı……..... 66
Şekil 4.10. Blok bağımlılıkları **(a)** 8 bağımlılık **(b)** 4 bağımlılık **(c)** 2 bağımlılık **(d)** 1 bağımlılık…….....69
Şekil 4.11. Bloksal bağımlılığın kurulduğu kırılgan filigranlama tekniği……...72
Şekil 4.12. HBC1 tekniğinde kullanılan bloksal bağımlılık…… 72
Şekil 4.13. Sıradüzensel Filigranlama Tekniğinin filigranlama yordamı…… 74
Şekil 4.14. Sıradüzensel 4 seviyeli blok yapısı…… 74
Şekil 4.15. Sıradüzensel Filigranlama Tekniğinin doğrulama yordamı……..……76
Şekil 5.1. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniği'nin filigranlama yordamı……. 79
Şekil 5.2. Sıradüzensel ağaç yapısı kullanılarak imgenin bloklara bölünmesi……..80
Şekil 5.3. 4 seviyeli sıradüzensel ağaç yapısında sayısal imzaların birleştirilmesiyle oluşturulan veri yükü…….83
Şekil 5.4. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniği'nin doğrulama yordamı……...83
Şekil 5.5. Sıradüzensel ağaç yapısındaki 4. ve 3. seviyedeki blok imzalarının birleştirilerek oluşturulması……...84
Şekil 5.6. 1200*1600 piksellik özgün imge……...86
Şekil 5.7. Sıradüzensel ağaç yapısı kullanılarak 1200x1600 piksellik bir imgenin Bölünmesi……..87
Şekil 5.8. Önerilen teknik ile filigranlanmış imge (çözünürlük=1200*1600 piksel)…… 89
Şekil 5.9. Değiştirilmiş filigranlı imge (çözünürlük=1200*1600 piksel)……. 90
Şekil 5.10. 2. Seviyede doğrulama işlemi **(a)** $I_{0,0}^2$ bloğu **(b)** $I_{0,1}^2$ bloğu **(c)** $I_{1,0}^2$ bloğu **(d)** $I_{1,1}^2$ bloğu……. 91
Şekil 5.11. Önerilen tekniğin doğrulama sonucu (çözünürlük=1200*1600 piksel)…… 92
Şekil 5.12. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması (çözünürlük=1200*1600 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu…….…… 92
Şekil 5.13. Önerilen teknik ile filigranlanmış imge (çözünürlük=768*1024 piksel)…….. 93
Şekil 5.14. Değiştirilmiş filigranlı imge (çözünürlük=768*1024 piksel)……94
Şekil 5.15. Önerilen tekniğin doğrulama sonucu (çözünürlük=768*1024 piksel)…….. 94
Şekil 5.16. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması (çözünürlük=768*1024 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu……. 94
Şekil 5.17. Önerilen teknik ile filigranlanmış imge (çözünürlük=480*640 piksel)…… 95

Şekil 5.18. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)……..96
Şekil 5.19. Önerilen tekniğin doğrulama sonucu (çözünürlük=480*640 piksel)…… 96
Şekil 5.20. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması (çözünürlük=480*640 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu……..96
Şekil 5.21. Önerilen teknik ile filigranlanmış imge (çözünürlük=240*320 piksel)…… 97
Şekil 5.22. Önerilen teknik ile filigranlanmış imgenin kes yapıştır saldırısı ile değiştirilmesi……. 98
Şekil 5.23. Önerilen tekniğin doğrulama sonucu (çözünürlük 240*320 piksel) **(a)** 5. seviyedeki doğrulama **(b)** 4. seviyedeki doğrulama…… 98
Şekil 5.24. Vektör Nicemlemeli Taklit Saldırısıyla oluşturulmuş 128*192 piksellik imge… 99
Şekil 5.25. VNTS'ye karşı önerilen tekniğin doğrulama sonucu **(a)** 5. seviyedeki doğrulama **(b)** 4. seviyedeki doğrulama…… 99
Şekil 5.26. Sıradüzensel Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=1200*1600 piksel)……101
Şekil 5.27. Değiştirilmiş filigranlı imge (çözünürlük=1200*1600 piksel)……..101
Şekil 5.28. Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=1200*1600 piksel)……102
Şekil 5.29. Tekniklerin doğrulama sonuçlarının karşılaştırılması (çözünürlük=1200*1600 piksel) **(a)** Önerilen tekniğin doğrulama sonucu **(b)** Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu…….. 102
Şekil 5.30. Sıradüzensel Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=768*1024 piksel)……..103
Şekil 5.31. Değiştirilmiş filigranlı imge (çözünürlük=768*1024 piksel)……103
Şekil 5.32. Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=768*1024 piksel)……..104
Şekil 5.33. Tekniklerin doğrulama sonuçlarının karşılaştırılması (çözünürlük=768*1024 piksel) **(a)** Önerilen tekniğin doğrulama sonucu **(b)** Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu…….. 104
Şekil 5.34. Sıradüzensel Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=480*640 piksel)……105
Şekil 5.35. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)……..105
Şekil 5.36. Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=480*640 piksel)……106

Şekil 5.37. Tekniklerin doğrulama sonuçlarının karşılaştırılması (çözünürlük=480*640 piksel) **(a)** Önerilen tekniğin doğrulama sonucu **(b)** Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu…….. 106
Şekil 5.38. 480*640 piksellik test imgesi…… 107
Şekil 5.39. Önerilen teknik ile filigranlanmış imge (çözünürlük=480*640 piksel)…… 108
Şekil 5.40. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)……..108
Şekil 5.41. Önerilen tekniğin doğrulama sonucu (çözünürlük=480*640)……...109
Şekil 5.42. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması (çözünürlük=480*640 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu……..109
Şekil 5.43. Sıradüzensel Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=480*640 piksel)……110
Şekil 5.44. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)……..110
Şekil 5.45. Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=480*640 piksel)……111
Şekil 5.46. Tekniklerin doğrulama sonuçlarının karşılaştırılması (çözünürlük=480*640 piksel) **(a)** Önerilen tekniğin doğrulama sonucu **(b)** Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu…….. 111
Şekil 5.47. 255*340 piksellik test imgesi…… 112
Şekil 5.48. Sıradüzensel Filigranlama Tekniği ile filigranlanmış test imgesi (çözünürlük=255*340 piksel)……112
Şekil 5.49. Değiştirilmiş filigranlı imge (çözünürlük=255*340 piksel)……..112
Şekil 5.50. Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=255*340 piksel)……113
Şekil 5.51. Önerilen teknik ile filigranlanmış imge (çözünürlük=255*340 piksel)…… 113
Şekil 5.52. Değiştirilmiş filigranlı imge (çözünürlük=255*340 piksel)……..114
Şekil 5.53. Önerilen tekniğin doğrulama sonucu (çözünürlük=255*340 piksel)…… 114
Şekil 5.54. Doğrulama sonuçlarının karşılaştırılması (çözünürlük=255*340 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu **(c)** Sıradüzensel Filigranlama Tekniğinin Doğrulama sonucu…….. 115
Şekil 6.1. Alt blok yaklaşımlı kırılgan filigranlama tekniğinin filigranlama yordamı…….. 120
Şekil 6.2. Alt blok yaklaşımı kullanılarak imgenin bloklara bölünmesi……... 121
Şekil 6.3. Her blok içerisine S_1 sayısal imzasının kaç bitinin gömüleceğinin belirlenmesi……... 124

Şekil 6.4. Her blok içerisine S_2 ve S_3 sayısal imzasının kaç bitinin gömüleceğinin belirlenmesi……... 124
Şekil 6.5. Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinin doğrulama yordamı…… 127
Şekil 6.6. Alt blokların doğrulanması için kullanılan algoritma……... 129
Şekil 6.7. 32*32 piksellik test imgesinin sayısal verileri ……..132
Şekil 6.8. 16*32 piksellik alt imgeler **(a)** A alt imgesi **(b)** B alt imgesi…… 133
Şekil 6.9. A ve B alt imgelerinin 8*8 piksellik bloklara bölünmesi…… 134
Şekil 6.10. Blokların 8*4 piksellik alt bloklara bölünmesi……. 135
Şekil 6.11. Blok ve bağımlı blokların oluşturduğu sayısal veriler **(a)** $A_{1,2}$ bloğuna ait sayısal veriler **(b)** $A_{1,2}$ ile $B_{1,1}^{2}$ bağımlılığına ait sayısal veriler **(c)** $A_{1,2}$ ile $B_{1,3}^{1}$ bağımlılığına ait sayısal veriler……..136
Şekil 6.12. Filigranlı $A_{1,2}$ bloğu…….. 137
Şekil 6.13. Değiştirilmiş filigranlı $A_{1,2}$ bloğu……. 137
Şekil 6.14. Önerilen tekniğinin doğrulama yordamının $A_{1,2}$ bloğuna ait sayısal verileri…… 138
Şekil 6.15. Önerilen teknik ile filigranlanmış imge (çözünürlük=1200*1600 piksel)…… 139
Şekil 6.16. Değiştirilmiş filigranlı imge (çözünürlük=1200*1600 piksel)……..139
Şekil 6.17. Önerilen tekniğin doğrulama sonucu (çözünürlük=1200*1600 piksel)…… 140
Şekil 6.18. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması (çözünürlük=1200*1600 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu……140
Şekil 6.19. Önerilen teknik ile filigranlanmış imge (çözünürlük=768*1024 piksel)…….. 141
Şekil 6.20. Değiştirilmiş filigranlı imge (çözünürlük=768*1024 piksel)……142
Şekil 6.21. Önerilen tekniğin doğrulama sonucu (çözünürlük=768*1024 piksel)…….. 142
Şekil 6.22. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması (çözünürlük=768*1024 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu……..143
Şekil 6.23. Önerilen teknik ile filigranlanmış imge (çözünürlük=480*640 piksel)…… 144
Şekil 6.24. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)……..144
Şekil 6.25. Önerilen tekniğin doğrulama sonucu (çözünürlük=480*640 piksel)…… 144
Şekil 6.26. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması (çözünürlük=480*640 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu……..145
Şekil 6.27. Önerilen teknik ile filigranlanmış imge (çözünürlük=240*320 piksel)…… 146

Şekil 6.28. Önerilen teknik ile filigranlanmış imgenin kes yapıştır saldırısı ile değiştirilmesi……. 146
Şekil 6.29. Önerilen tekniğin doğrulama sonucu (çözünürlük=240*320 piksel)…… 147
Şekil 6.30. Vektör Nicemlemeli Taklit Saldırısıyla oluşturulmuş 128*192 piksellik imge… 147
Şekil 6.31. VNTS'ye karşı önerilen tekniğin doğrulama sonucu…… 148
Şekil 6.32. HBC2 Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=1200*1600 piksel)…….....149
Şekil 6.33. Değiştirilmiş filigranlı imge (çözünürlük=1200*1600 piksel)……..150
Şekil 6.34. HBC2 Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=1200*1600 piksel)…….……...150
Şekil 6.35. Tekniklerin doğrulama sonuçlarının karşılaştırılması (çözünürlük=1200*1600) **(a)** Önerilen tekniğin doğrulama sonucu **(b)** HBC2 Filigranlama Tekniğinin doğrulama sonucu…….151
Şekil 6.36. HBC2 Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=768*1024 piksel)…….…152
Şekil 6.37. Değiştirilmiş filigranlı imge (çözünürlük=768*1024 piksel)……152
Şekil 6.38. HBC2 Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=768*1024 piksel)…….…152
Şekil 6.39. Tekniklerin doğrulama sonuçlarının karşılaştırılması (çözünürlük=768*1024 piksel) **(a)** Önerilen tekniğin doğrulama sonucu **(b)** HBC2 Filigranlama Tekniğinin doğrulama sonucu……...……..153
Şekil 6.40. HBC2 Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=480*640 piksel)……....153
Şekil 6.41. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)……..154
Şekil 6.42. HBC2 Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=480*640 piksel)…….…...154
Şekil 6.43. Tekniklerin doğrulama sonuçlarının karşılaştırılması (çözünürlük=480*640 piksel) **(a)** Önerilen tekniğin doğrulama sonucu **(b)** HBC2 Filigranlama Tekniğinin doğrulama sonucu……...…..155
Şekil 6.44. Önerilen teknik ile filigranlanmış imge (çözünürlük=480*640 piksel)…… 156
Şekil 6.45. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)……..156
Şekil 6.46. Önerilen tekniğin doğrulama sonucu (çözünürlük=480*640 piksel)…… 157
Şekil 6.47. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması…..157
Şekil 6.48. HBC2 Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=480*640 piksel)……158

Şekil 6.49. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)……..158
Şekil 6.50. HBC2 Filigranlama Tekniğinin doğrulama sonucu
(çözünürlük=480*640 piksel)…….…...159
Şekil 6.51. Tekniklerin doğrulama sonuçlarının karşılaştırılması
(çözünürlük=480*640 piksel) **(a)** Önerilen tekniğin doğrulama sonucu
(b) HBC2 Filigranlama Tekniğinin doğrulama sonucu…….. 159

TABLOLAR LİSTESİ

Tablo 2.1. Bilgi gizlemenin sınıflandırılması…….… 14
Tablo 3.1. Olası girişlere göre, bulucu çıkışı….……..36
Tablo 5.1. Sıradüzensel Filigranlama Tekniği ile önerilen tekniğin karşılaştırılması…… 116
Tablo 6.1. HBC2 Filigranlama Tekniği ile önerilen tekniğin karşılaştırılması..…… 160

EKLER LİSTESİ

Ek 1……181
Ek 2……... 193

SİMGELER

m **:** mesaj

I : Özgün imge

I' : Filigranlı imge

$\tilde{I}$: En az önemli bit düzlemi sıfır yapılmış imge

$\hat{I}$: Değiştirilmiş Filigranlı imge

x, y : Uzaysal koordinatlar

M : Sayısal imgenin sayır sayısı

N : Sayısal imgenin sütun sayısı

W : Filigran

K : Anahtar

K_G : Filigran gömü anahtarı

K_B : Filigran bulma Anahtarı

α : Etkinlik değeri

Δ : Nicemleme adım boyu

X : Herhangi bir sinyal

t : Eşik değeri

Z_{LK} : Lineer korelasyon değeri

Z_{NK} : Normalize edilmiş korelasyon değeri

Z_{KK} : Korelasyon katsayısı

$\dot{I}$: Birim genliğe normalize edilmiş imge

$\dot{W}$: Birim genliğe normalize edilmiş filigran

$\bar{I}$: Sayısal imgenin ortalama parlaklık değeri

$\overline{W}$: Filigranın ortalama değeri

g : Gürültü

$I_{\max}$: Sayısal imgedeki maksimum parlaklık değerine sahip pikselin değeri

E : Kriptolama algoritması

D : Kripto çözme algoritması

c : Kriptolu metin

K_E : Kriptolama anahtarı

K_D : Kripto çözme anahtarı

h : Özüt değeri

H : Özüt fonksiyonu

b : İkili filigran imgesi

ε : Filigran gömme fonksiyonu

$\mathcal{D}$: Filigran çıkarım fonksiyonu

z : Işıklandırma değeri

q : Yansıma değeri

f : parlaklık değeri

C_K : Eşitlik sınıfları

V : Sahte imge

I_{ID} : İmge indisi

KISALTMALAR LİSTESİ

EÖB : En az Önemli Bit
VNTS : Vektör Nicemlemeli Taklit Saldırısı
JPEG : Joint Photographic Experts Group
OKH : Ortalama Karesel Hata
SGO : Sinyal Gürültü Oranı
İGS : İnsan Görme Sistemi
CCD : Charge Coupled Device
CMOS : Complementary Metal Oxide Semiconductur
AKD : Ayrık Kosinüs Dönüşümü
AFD : Ayrık Fourier Dönüşümü
WRS : Watermark Removing Scheme
RSA : Rivest Shamir Adleman
HBC1 : Hash Block Chaining Version 1
HBC2 : Hash Block Chaining Version 2
DSA : Digital Signature Algorithm
MAC : Message Authentication Code
MD5 : Message Digest 5
SHA : Secure Hash Algorithm

ÖZET

Doktora Tezi

SAYISAL FİLİGRANLAR İLE KRİPTO İMZALARININ BİRLİKTE KULLANILMASI ve ÇOKLU ORTAM VERİSİ ÜZERİNDEKİ UYGULAMALARI

Metin ERTÜRKLER

Fırat Üniversitesi
Fen Bilimleri Enstitüsü
Elektrik-Elektronik Mühendisliği Anabilim Dalı
2007, Sayfa : 198

Bu tez çalışmasında sayısal imgelerin aslıyla aynılığını doğrulamak için hem saldırılara karşı güvenilir hem de yüksek doğrulukta değişim bölgesi belirleme niteliğine sahip iki kırılgan filigranlama tekniği önerilmiştir. Önerilen ilk teknik Sıradüzensel Filigranlama Tekniği temel alınarak geliştirilmiştir. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniği olarak adlandırılan bu teknikte imgenin alt bloklara bölünmesinde tavan ve taban fonksiyonlarının kullanılması önerilmiştir. Ayrıca sıradüzensel ağaç yapısı içerisindeki seviyelerde iki ayrı sayısal imza algoritması kullanılmıştır. Önerilen tekniğin, Sıradüzensel Filigranlama Tekniğine göre daha etkin bir filigranlama tekniği olduğu yapılan deneysel çalışmalar ile gösterilmiştir.

Önerilen ikinci kırılgan filigranlama tekniği ise bloksal bağımlılık kullanılarak oluşturulmuştur. Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniği olarak adlandırılan bu teknikte yeni bir bağımlılık yapısı ve bu bağımlılık yapısını kullanan yeni bir doğrulama algoritması geliştirilmiştir. Bu doğrulama algoritması önerilen teknik tarafından değişim bölgesinin 4 kata kadar daha doğru bir çözünürlükte belirlenebilmesini sağlamıştır. Ayrıca önerilen bu teknikte de iki ayrı sayısal imza algoritmasının kullanılması önerilmiştir. Bu öneri daha güvenilir ve uzun bit çıkışına sahip sayısal imza algoritmaları kullanılması durumunda, tekniklerin güvenirlik seviyesi ile değişim bölgesi belirleme doğruluğu arasında bir denge sağlayarak performansı önemli miktarda artırmıştır. Önerilen tekniğin literatürde önerilmiş diğer bir blok bağımlı teknik olan HBC2 Filigranlama Tekniğine göre daha yüksek bir performansa sahip olduğu yapılan deneysel çalışmalar sonucunda görülmüştür.

Anahtar Kelimeler : Kırılgan Filigranlama, İmge Doğrulama, Bilgi Gizleme, Güvenlik.

ABSTRACT

PhD Thesis

THE USE OF DIGITAL WATERMARKS WITH DIGITAL SIGNATURES AND THEIR APPLICATIONS ON MULTIMEDIA DATA

Metin ERTÜRKLER

Fırat University

Graduate School of Natural and Applied Sciences

Department of Electrical – Electronics Engineering

2007, Page : 198

In this thesis, two fragile watermarking techniques having both high accuracy localization resolution property and security against attacks are proposed for image authentication. The first proposed technique is based on Hierarchical Watermarking Technique. In the proposed technique, named "Hierarchical Tree Structure Watermarking Technique", it has been suggested to divide the image into the sub-blocks using ceiling and floor functions. Furthermore, two different signature algorithms are employed in the hierarchical tree structure. It has been shown by the experimental work that the proposed technique has better performance than the Hierarchical Watermarking Technique.

The second proposed fragile watermarking technique is based on dependent block structure. In this approach, called "A Sub-Block Based Fragile Watermarking Technique", both a novel dependent block structure and a watermark verification algorithm are developed. This verification algorithm provides more accurately resolution at the determination of tamper region up to 4 times. In addition, it has been suggested to use two separate signature algorithms employing the proposed technique. In case of employing more secure and longer signature algorithms, this suggestion provides a superior tradeoff between security and localization resolution. The experimental results demonstrate that the proposed sub-block based fragile watermarking technique shows better performance than the HBC2 Watermarking Technique mentioned in the literature.

Keywords: Fragile Watermarking, Image Authentication, Information Hiding, Security

1. GİRİŞ

Son yıllarda sayısal sistem ve iletişim teknolojilerindeki gelişmelere paralel olarak sayısal imgelerin kullanımında olağanüstü bir artış görülmektedir. Sayısal bir imge (görüntü) sayısal fotoğraf makineleri, sayısal kameralar, tarayıcılar ve kişisel bilgisayarlardaki imge işleme yazılımları kullanılarak kolayca oluşturulabilir. Kullanıcı açısından sayısal imgeler, analog imgelere göre daha kullanışlı ve üstün özelliklere sahiptirler. Bu üstün özellikler arasında kolay ve hatasız kopyalama, hızlı dağıtım, kullanımla eskimeme, imgenin içeriğinin istenildiği gibi değiştirilmesi verilebilir. Analog imgeler üzerinde bu işlemlerin gerçekleştirilmesi oldukça zor ve zaman alıcıdır. Örneğin analog bir imgenin profesyonel olmayan bir kişi tarafından kopyalanması veya içeriğinin değiştirilmesi çok zordur. Konunun uzmanı bir kişi tarafından kopyalanması veya içeriğinin değiştirilmesi durumunda ise, analog imgenin fiziksel yapısından dolayı imgenin görsel kalitesinde bozulmalar oluşmaktadır. Sayısal bir imge için bu durum düşünüldüğünde, kopyalama işlemi herkes tarafından kolaylıkla, istenilen sayıda yapılabilir ve imgenin içeriği bir imge işleme programı vasıtasıyla imgenin görsel kalitesinde bir bozulma olmadan değiştirilebilir.

Sayısal imgelerin ilk bakışta avantajlı görünen kolay kullanım özellikleri, illegal amaçlar için kullanıldığında önemli problemler oluşturmaktadır. Örneğin, telif hakkı sahibi açısından analog ve sayısal bir imgenin illegal olarak kopyalanmasını düşünelim. Analog imgenin illegal olarak kopyalanması durumunda, telif hakkı sahibi kendisinde bulunan özgün negatifi öne sürerek telif haklarını koruyabilir. Sayısal bir imgenin illegal olarak kopyalanması durumunda ise özgün negatif gibi kullanılabilecek bir kanıt olmadığı için, telif hakkı sahibi haklarını koruyamaz. Diğer bir örnek, mahkemede kanıt niteliği olan analog ve sayısal imgeler için verilebilir. Bu imgelerin kanıt niteliğindeki bölgelerinin değiştirilmesi durumunda, analog imge sahibi yine özgün negatifi kullanarak değişim bölgelerini kanıtlayabilir. Ancak sayısal imge sahibi özgün negatife sahip olmadığı için değişimi kanıtlayabilecek herhangi bir delile sahip değildir.

Sayısal imgelerin dağıtımı ve paylaşımı genellikle Internet üzerinden yapılmaktadır. Ancak Internet gibi açık bir ortam içerisinde, kullanıcıların imgelerin bütünlüğünü ve kaynağını doğrulayamaması diğer önemli bir probleme yol açmaktadır.

Sayısal imgelerle ilgili yukarıdaki problemlerin çözümünde ilk başvurulan alan Kriptografi *(Cryptography)* Bilimi'dir. Kriptografi Bilimi'nde verinin doğrulanması *(authentication)* için Mesaj Doğrulama Kodları *(Message Authentication Codes)* ve Sayısal İmzalar *(Digital Signatures)* kullanılmaktadır [1-4].

1.1. Veri Doğrulama

Doğrulama Algoritmaları, Kriptografi Bilimi'nde üzerinde oldukça çalışılan konulardan biridir. Doğrulama, üç önemli niteliğe sahip olan genel bir terimdir. Bu terimin daha iyi anlaşılması için, Kriptografi Bilimi'nde iki kişi arasıdan geçen bir senaryo örnek olarak verilmektedir. Bu senaryoda, Ali göndericiyi ve Ayşe alıcıyı temsil eden iki karakterdir. Senaryo, Ali'nin m mesajını Ayşe'ye göndermesinden oluşur. Bu senaryo üzerinde çalışan bir doğrulama algoritması aşağıdaki üç önemli niteliğe sahip olmalıdır:

Bütünlük Doğrulama: Doğrulama algoritmasının, Ali'nin gönderdiği mesaj ile Ayşe'nin aldığı mesajın birebir aynı olup olmadığını belirleyebilmesi gerekir.

Kaynak Doğrulama: Doğrulama algoritmasının, mesajın gerçekten Ali tarafından gönderildiğini kanıtlaması gerekir.

İnkar Edememe: Doğrulama algoritması, Alinin m mesajını gönderdiğini inkar edememesini sağlamalıdır.

Kriptografi'de iki tip doğrulama algoritması geliştirilmiştir: Sayısal İmzalar ve Mesaj Doğrulama Kodları [1-4]. Sayısal İmza Algoritmaları'nda ilk olarak, bir özüt fonksiyonu *(hash function)* kullanılarak m mesajına ait özüt hesaplanır. Daha sonra, elde edilen özüt göndericinin özel anahtarı ile kriptolanır ve m mesajına ait sayısal imza oluşturulur. Gönderici m mesajını iletirken, mesajı doğrulayacak olan sayısal imzayı da birlikte gönderir. Alıcı m mesajını ve sayısal imzayı aldığında, göndericinin genel anahtarı ile sayısal imzayı çözerek özgün özüt değerini elde eder. Daha sonra alıcı m mesajının özütünü tekrar hesaplar ve özgün özüt ile karşılaştırır. Özütler eşleşirse, alıcı mesajı doğrular.

Mesaj Doğrulama Kodları'nda, gönderici ve alıcı arasında gizli bir anahtar ortak olarak kullanılır. Mesaj Doğrulama Kodları, kriptografik özüt fonksiyonları ve blok şifreleyiciler kullanılarak oluşturulur. Gönderici, alıcıya mesaj ile birlikte mesaj doğrulama kodunu da gönderir. Alıcı, aldığı mesaja ait mesaj doğrulama kodunu tekrar hesaplar ve gönderilen mesaj doğrulama kodu ile karşılaştırır. Mesaj doğrulama kodları eşleşirse alıcı mesajın bütünlüğünü doğrular.

Hem Sayısal İmzalar hem de Mesaj Doğrulama Kodları bir mesajın bütünlüğünü doğrulayabilirler [2]. Ancak, mesaj kaynağının doğrulanması ve göndericinin mesajı gönderdiğini inkar edememesi, sadece Sayısal İmzaların kullanılmasıyla sağlanır. Mesaj Doğrulama Kodları son iki doğrulamayı sağlayamazlar. Sayısal imgeleri doğrulamak için kullanılan Sayısal İmzalar ve Mesaj Doğrulama Kodları'na ait doğrulama verileri ya sayısal imgenin başlığı içerisine yerleştirilir yada ayrı bir dosyada tutulur. Ancak doğrulama verilerinin imgeden ayrı olarak tutulması aşağıdaki problemlere yol açmaktadır [5]:

- Doğrulama verisinin ayrı bir dosya içerisinde tutulması, depolama alanında ilave bir yer ihtiyacı ve ağ ortamında ek bir bant genişliği gerektirir. Pratik uygulamalarda, ilave yer ve bant genişliği ihtiyacı doğrulama verilerinin kullanımını büyük ölçüde kısıtlamaktadır. Örneğin imgelerin tutulduğu bir veritabanında, her bir imgeye ait doğrulama verisinin imgeden ayrı olarak depolanması büyük bir yönetim zorluğu oluşturmaktadır.
- Her imge formatının başlık alanında doğrulama verisini yerleştirecek bir alan bulunmamaktadır. Bu yüzden doğrulama verisinin imge başlığı içerisine yerleştirilmesi imgenin formatına bağımlıdır. Sayısal imgeler kullanım sırasında bir çok format değişimine maruz kalırlar. Örneğin Bitmap formatında kaydedilmiş bir imgenin JPEG formatına dönüştürülmesi sayısal ortamlarda sıkça rastlanılan bir işlemdir. İmge başlığı içerisine yerleştirilmiş doğrulama verileri, format değişim ile birlikte yok olmaktadır.
- Doğrulama verileri sayısal imgelerin değiştirilip değiştirilmediğini katı bir yaklaşımla belirler. Şöyle ki, imgenin bir pikselinin bir bitinin bile değiştirilmesi tüm imgenin doğrulanamamasına neden olur. Bazı uygulamalarda bu katı yaklaşımdan dolayı doğrulama verilerinin kullanılması istenilmez. Örneğin Internet üzerinde bir imgenin iletilmesindeki bir bitlik bozulma alıcı tarafından önemsenmeyebilir. Ancak, doğrulama verisi tüm imgeyi doğrulamayacağı için iletimin tekrarlanması gerekir.
- Doğrulama verileri, imge üzerinde bir değişimin yapılıp yapılmadığını belirleyebilirler, fakat değişim yapılan bölge hakkında bir bilgi vermezler. Bazı uygulamalarda değişim yapılan bölgenin belirlenmesi istenilmektedir. Örneğin mahkemede kanıt nitelliğindeki bir imgenin değiştirilmesi durumunda, değiştirilen bölgenin belirlenmesi oldukça önemli olabilir.

Sayısal imgelerin doğrulanmasında, son yıllarda Doğrulama Algoritmaları'na alternatif ve/veya tamamlayıcı olan Sayısal Filigranlama Teknikleri *(Sayısal Damgalama, Digital Watermarking)* önerilmiştir [6-59]. Sayısal Filigranlama Teknikleri, doğrulama verilerinin kullanılmasıyla oluşan yukarıdaki problemlere çözüm sağlamaktadır. Sayısal Filigranlama, doğrulamanın yanı sıra bir çok uygulama içerisinde farklı amaçlar için kullanılmaktadır [5, 14-18]. Sayısal filigranlar uygulamaya bağımlı olarak kırılgan *(fragile)*, yarı kırılgan *(semi fragile)* veya gürbüz *(robust)* bir özelliğe sahiptirler. Gürbüz filigranlardan, gömüldükleri sayısal veri kullanılamaz hale gelinceye kadar varlığını sürdürmesi beklenir [19-33]. Gürbüz filigranların aksine kırılgan filigranlar, gömüldükleri sayısal veri üzerindeki en küçük değişiklikte bozulacak yapıda tasarlanırlar [34-54]. Yarı kırılgan filigranlar ise gürbüz ve kırılgan filigranların özelliklerini birleştiren bir yapıya sahiptirler [6-13]. Yarı kırılgan filigranlar, sayısal verinin içeriğinin değiştirilmesi gibi kötü niyetli saldırılara karşı kırılgan, fakat sıkıştırma gibi içeriği değiştirmeyen işlemlere karşı gürbüz bir yapıda tasarlanırlar.

1.4. Sayısal Filigranlar Kullanılarak İmgelerin Doğrulanması

Sayısal Filigranlama imge, ses ve video gibi sıkça kullanılan sayısal verilerin filigranlanması için kullanılır. Sayısal Filigranlama, özgün veriye ilişkin bilginin (filigran) özgün verinin algılanabilir kalitesini bozmadan doğrudan doğruya özgün veri içerisine gömülmesi olarak tanımlanabilir. Sayısal filigranların doğrudan doğruya özgün veri içerisine gömülmesiyle, doğrulama verilerinin kullanımıyla oluşan problemlere çözüm sağlanır. Örneğin, filigran imge formatından bağımsız olarak imgenin piksel değerleri içerisine gömüldüğü için, imgenin format değişimi filigranı etkilemez. Ayrıca ilave bir yer ve bant genişliğine ihtiyaç duyulmaz. İmgelerin doğrulanmasında Yarı Kırılgan Filigranlama Teknikleri kullanılarak, doğrulama işlemi daha yumuşak bir yaklaşım içerisinde yapılabilir. Ayrıca Sayısal Filigranlama Teknikleri'nde bloksal yapılar kullanılarak, değişim yapılan bölgenin tespit edilmesi de mümkündür. Bu özellik, mahkemede kanıt niteliğinde olan imgeler veya askeri alanda kullanılan imgeler gibi içeriği önemli olan uygulamalarda oldukça yararlıdır.

Sayısal Filigranlama Teknikleri sayısal verinin değerlerini değiştirerek filigranı sayısal veri içerisine gömer. Ancak bu değişim sayısal veri üzerinde gömü bozulması olarak bilinen bir bozulmaya neden olur. Sayısal Filigranlama Uygulamaları'nda sayısal verinin görsel kalitesinin korunması önemli bir konudur. Çoğu uygulama için İnsan Görme Sistemi'nin bu bozulmayı algılayamaması yeterlidir. Ancak medikal imgeler gibi içeriği önemli olan uygulamalarda filigranın hiçbir bozulmaya neden olmaması istenir. Bu uygulamalarda Tersinir Filigranlama Teknikleri kullanılır [55-59].

İmgelerin doğrulanması uygulamalarında Kırılgan ve Yarı Kırılgan Filigranlama Teknikleri kullanılır. Bu tez çalışmasında kırılgan filigranlar üzerinde odaklanılmıştır. Kırılgan Filigranlama Teknikleri'nde gömü bozulmasını minimuma indirmek için, filigran imgenin en az önemli bit (EÖB) düzlemi içerisine gömülür. İmgelerin doğrulanması uygulamalarında, kırılgan filigran olarak Sayısal İmzalar veya Mesaj Doğrulama Kodları kullanılabilir. Bu doğrulama verilerinin filigran olarak kullanılmasıyla önemli avantajlar elde edilir. Örneğin sayısal bir imzanın filigran olarak kullanılmasıyla, sayısal imzanın sahip olduğu bütünlük doğrulama, kaynak doğrulama ve inkar edememe gibi önemli nitelikler filigranın bünyesine katılır. Ayrıca filigran, sayısal imzaların sahip olduğu yüksek seviyeli kriptografik güvenlik özelliklerine de sahip olur [38-43].

1.5. Kırılgan Filigranlama Teknikleri

İmge doğrulama uygulamalarında kırılgan ve yarı kırılgan filigranlar kullanılır. Bu uygulamalarda kırılgan filigranlar katı bir yaklaşım içerisinde, yarı kırılgan filigranlar ise daha esnek bir yapıda doğrulama işlemini gerçekleştirirler. Şöyle ki, imge üzerinde yapılan en küçük değişim, imge içerisine gömülen kırılgan filigranın bütünlüğünü bozarken, yarı kırılgan filigranın bütünlüğünü bozabilmesi değişimin türüne bağımlıdır. İmge doğrulama uygulamalarında imgenin doğrulanması, imge içerisinde gömülmüş filigranın varlığı ve bütünlüğü kontrol edilerek yapılır. İmge içerisine birden fazla filigranın gömülmesi, değişim yapılan bölgenin belirlenmesini sağlar. İmge doğrulama uygulamalarında, kırılgan bir filigrandan beklenilen önemli nitelikler şunlardır: doğrulama, güvenliği sağlama, aslına benzerliği koruma ve değişim bölgesi belirlemedir [5,41].

İmgedeki değişim bölgesini belirleyebilen ilk kırılgan filigranlama tekniklerinden biri Walton [44] tarafından önerilmiştir. Bu teknikte, sahte rasgele seçilen piksellerin EÖB düzlemi içerisine, bu piksellerin en önemli 7 bitinin sınama toplamının gömülmesi önerilmiştir. Bu teknik, bir pikselin yalnızca bir bitinde değişim olduğunda bozulmayı belirleyebildiği için, sınırlı bir değişim bölgesi belirleme niteliğine sahiptir. Ayrıca pikseller üzerinde değişimler yapılarak sahte bir filigranın geçerli olarak doğrulanması mümkün olduğundan dolayı, bu teknik önemli bir güvenlik açığına sahiptir.

Yeung ve Mintzer [45] tarafından, imgedeki her bir pikselin bütünlüğünü doğrulayabilecek basit ve hızlı bir kırılgan filigranlama tekniği önerilmiştir. Bu teknik imge içerisinde yapılan piksel değişimlerini belirleyebilmesine rağmen, imgenin boyutlarındaki değişimleri belirleyememektedir [38]. Daha sonra Yeung ve Mintzer tarafından önerilen teknik, JPEG sıkıştırma alanında imge doğrulanması için nicemlenmiş Ayrık Kosinüs Dönüşüm katsayılarında kullanılmıştır [46]. Ancak Yeung ve Mintzer tarafından önerilen teknikte, filigran olarak kullanılan logonun gizliliğinin sağlanamaması durumunda, tekniğin saldırılara karşı açık olduğu gösterilmiştir [47]. Ayrıca filigran olarak aynı logonun birden fazla imge içerisinde kullanılması durumunda, logo ve anahtar bilgisine ihtiyaç duyulmadan bir saldırı yapılabileceği de belirtilmiştir [48].

Değişim bölgesi belirleme yeteneğine sahip diğer bir teknik Wong [37] tarafından önerilmiştir. Bu teknikte doğrulanacak olan imgeler üst üste gelmeyen bloklara bölünerek, her bir bloğun EÖB düzlemi içerisine blok piksellerinden hesaplanan bir mesaj doğrulama kodu gömülmüştür. Mesaj Doğrulama Kodlarında, simetrik anahtarlı bir yapı kullanılmaktadır. Dolayısıyla bu tekniğin güvenliği, kullanıcı anahtarının gizliliğinin korunmasıyla sağlanmaktadır. Simetrik anahtarlı sistemlerde kullanıcı anahtarı hem özütün kriptolanmasında

hem de mesaj doğrulama kodunun çözülmesinde kullanılmaktadır. Bu yüzden filigranlı bir imgenin doğrulanması sadece gizli anahtara sahip kişi tarafından yapılabilmektedir. Gizli anahtarın öğrenilmesi ise sistemin güvenliğini tehlikeye atmaktadır. Bu problemi çözmek için Wong [38] tarafından simetrik anahtarlı yapının asimetrik anahtarlı sürümü önerilmiştir. Bu sürümde Mesaj Doğrulama Kodları yerine asimetrik anahtar yapısına dayanan Sayısal İmzaların kullanılması önerilmiştir. Asimetrik anahtarlı sistemlerde özütün kriptolanmasında özel anahtar, Sayısal İmzaların çözülmesinde ise genel anahtar kullanılır. Sayısal imgenin doğrulanması genel anahtar kullanılarak yapılacağı için sistemin güvenliği de tehlikeye atılmaz. Ayrıca filigran olarak Sayısal İmzaların kullanılmasıyla, Sayısal İmzaların sahip olduğu kaynak doğrulama ve inkar edememe niteliği de filigranın bünyesine katılır. Wong tarafından önerilen bu tekniklerde, değişim bölgesi seçilen blok büyüklüğüne bağımlı olarak belirlenmektedir.

Hem Yeung ve Mintzer hem de Wong tarafından önerilen tekniklerde filigranlama işlemi diğer bloklardan bağımsız olarak yapıldığı için bu teknikler "Bloksal Bağımsız Filigranlama Teknikleri" olarak adlandırılır. Bloksal Bağımsız Filigranlama Teknikleri'nin, Vektör Nicemlemeli Taklit Saldırısı (VNTS) olarak adlandırılan saldırıya karşı korunmasız olduğu gösterilmiştir [39]. Bu saldırıda, filigran olarak kullanılan logoya ve kullanıcı anahtarına ihtiyaç duyulmadan sahte imgelerin özgün bir filigran ile filigranlanması mümkündür. Yeung ve Mintzer'in tekniği daha küçük blok büyüklüğüne (1x1 piksel) sahip olduğu için, VNTS'ye karşı daha açıktır.

VNTS'yi önlemek için blok boyutlarının büyütülmesi, blok özütlerinin hesaplanmasında blok indislerinin eklenmesi, daha karmaşık logoların kullanılması önerilmiştir [39]. Ancak bu öneriler saldırı yapılmasını önleyememiş, sadece saldırı yapılmasını zorlaştırmıştır. Etkili bir çözüm ise bloklar arasında bağımlılığın kurulmasıyla sağlanmıştır [39]. Bloksal bağımlılıkta her bir blok içerisine gömülecek özütün hesaplanmasında, hem bloğun pikselleri hem de bloğa komşu seçilen diğer bloğun pikselleri birlikte kullanılır. Böylece her bir bloğun diğerine bağımlılığı sağlanarak saldırı yapılması önlenir. Ancak bloksal bağımlılıkta, bir bloğun bir pikselinin bile bozulması her iki bloğun birden doğrulanamamasına neden olur. Diğer bir çözüm ise her bir blok içerisine o bloğu kapsayan daha büyük bir bloğun özütünün gömülmesi ile elde edilmiştir [49]. Ancak bu çözüm saldırı yapılmasını bir dereceye kadar önlerken, değişim bölgesi belirleme doğruluğunu kaybetmiştir [50]. Alternatif bir çözüm olarak blok özütleri içerisine her bir imgeye ait benzersiz bir indisin gömülmesi önerilmiştir [51]. Bu öneri VNTS'nin yapılmasını tamamıyla önler. Ancak aynı imge indisinin doğrulama yordamında da kullanılması gerekir. İmge veritabanlarında yapılacak doğrulamalarda imge indislerinin yönetilmesi büyük bir yük getirecektir. Bu problemi çözmek için imge indisinin imge piksellerinden hesaplanması önerilmiştir [40, 60, 61]. Ancak imge üzerinde bir değişim

yapılması sonucunda yanlış imge indisinin çıkarılması, önerilen tekniğin değişim bölgesi belirleme niteliğini yok edecektir. Literatürde VNTS'nin yanı sıra Bloksal Bağımsız Kırılgan Filigranlama Teknikleri üzerinde etkili olan Doğum Günü Saldırısı [2], Kes-Yapıştır Saldırısı [42] ve Aktarıma Dayalı Saldırı [42] önerilmiştir.

Literatürde hem saldırılara karşı güvenilir hem de değişim bölgesi belirleme nitelliğine sahip önemli bir kırılgan filigranlama tekniği Çelik ve diğ. [52] tarafından önerilmiştir. Sıradüzensel Filigranlama Tekniği *(Hierarchical Watermarking Technique)* olarak adlandırılan bu teknikte imge sıradüzensel (hiyerarşik) bir yapı içerisinde alt bloklara bölünür. Sıradüzenin her bir seviyesi bir alt seviyedeki 4 bloğun birleşmesinden oluşur. Sıradüzenin en üst seviyesindeki bloğa (imgenin kendisi) ait sayısal imza saldırılara karşı tam bir güvenlik sağlarken, en alt seviyedeki bloklara ait sayısal imzalar değişim bölgesinin belirlenmesinde kullanılırlar. Sıradüzensel Filigranlama Tekniğinin değişim bölgesi belirleme doğruluğu imge çözünürlüğüne bağımlıdır. Eğer imge sıradüzenin en üst seviyesinde bir alt seviyeye bölünemezse, Sıradüzensel Filigranlama Tekniği'nin değişim bölgesi belirleme niteliği yok olmaktadır. İmgenin sıradüzenin üst seviyelerinde bir alt seviyeye bölünememesi durumunda ise, Sıradüzensel Filigranlama Tekniği'nin değişim bölgesi belirleme doğruluğu oldukça azalmaktadır. Sıradüzensel yapının en alt seviyesindeki bloklar hem kendi bloklarına ait sayısal imzaları hem de sıradüzensel seviyeye bağımlı olarak üst seviyelerdeki bloklara ait sayısal imzaların bir parçasını taşımaktadırlar. Dolayısıyla seçilen sayısal imza algoritmasının bit uzunluğu en alt seviyedeki blok büyüklüğünü belirlemektedir. Daha yüksek bir güvenlik seviyesi için daha uzun bit çıkışına sahip bir sayısal imza algoritmasının kullanılması, Sıradüzensel Filigranlama Tekniği'nin değişim bölgesi belirleme doğruluğunu azaltmaktadır.

1.4. Tezin Amacı

Bu tez çalışmasında doğrulama uygulamalarında kullanılmak için hem saldırılara karşı güvenilir hem de yüksek doğrulukta değişim bölgesi belirleme niteliğine sahip iki kırılgan filigranlama tekniği önerilmiştir.

Önerilen ilk teknik, Sıradüzensel Filigranlama Tekniği temel alınarak geliştirilmiştir. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniği olarak adlandırılan bu teknikte imgenin bloklara bölünmesinde sıradüzensel yapı yerine sıradüzensel ağaç yapısının kullanılması önerilmiştir. Sıradüzensel ağaç yapısında tavan ve taban fonksiyonları kullanılarak imgenin sıradüzen içerisinde sürekli olarak alt seviyelere bölünmesi garanti altına alınmıştır. İmge çözünürlüğünden bağımsız olarak imgenin istenilen alt seviyeye kadar bölünebilmesi, önerilen tekniğin tüm imge çözünürlüklerinde yüksek doğrulukta değişim bölgesi belirleme niteliğine

sahip olmasını sağlamıştır. Ayrıca önerilen teknikte daha yüksek kriptografik güvenlik seviyesine ve daha uzun bit uzunluğuna sahip bir sayısal imza algoritmasının, en alttaki blok büyüklüğünü minimum seviyede etkilemesi için, iki ayrı sayısal imza algoritmasının kullanılması önerilmiştir. Bu öneride alt seviyedeki blok imzalarının hesaplanmasında daha kısa bit çıkışına sahip sayısal imza algoritmasının kullanılması, daha üst seviyelerde ise daha güvenilir ve daha uzun bit çıkışına sahip sayısal imza algoritmasının kullanılması öngörülmüştür. Seviyeler içerisinde hangi sayısal imzanın kullanılacağı ise geliştirilen bir algoritma ile belirlenmiştir. Bu algoritma ile önerilen tekniğin kriptografik güvenlik seviyesi artırılırken, değişim bölgesi belirleme doğruluğunun azalmaması temin edilmiştir.

Önerilen ikinci kırılgan filigranlama tekniği ise bloksal bağımlılığa dayanmaktadır. Bloksal bağımlılığı kullanan kırılgan filigranlama tekniklerinde değişim bölgesi belirleme doğruluğu kullanılan bağımlılık sayısına göre değişmektedir. Değişim bölgesini daha yüksek doğrulukta belirleyebilmek için, HBC2 *(Hash Block Chaining Version 2)* Filigranlama Tekniğinde blok başına bir bağımlılığın kullanılması önerilmiştir [42]. HBC2 Filigranlama Tekniği, hem saldırılara karşı güvenilir bir yapıya hem de değişim bölgesi belirleme niteliğine sahip olduğundan dolayı önemli bir kırılgan filigranlama tekniğidir. Ancak bir bağımlılığın kullanılması durumunda bile, bir bloğun bir pikselindeki bozulma iki bloğun birden doğrulanamamasına neden olur. Ayrıca bloksal yapılı kırılgan filigranlama tekniklerinde blok büyüklüğü, kullanılan sayısal imzanın bit uzunluğuna bağımlı olarak seçilir [37, 38, 42, 43, 52]. Daha uzun bit çıkışına sahip sayısal imzaların kullanılması değişim bölgesi belirleme doğruluğunu azaltır.

Önerilen ikinci kırılgan filigranlama tekniği Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniği olarak adlandırılmıştır. Bu teknikte imgenin bloklara bölünmesinde yeni bir yaklaşım geliştirilmiştir. Alt bloklama yapısı olarak adlandırılan bu yaklaşımda, doğrulanacak olan imge iki alt imgeye, her bir alt imge üst üste gelmeyen bloklara, ve bloklar alt bloklara bölünmüştür. Daha sonra her bir alt imge içerisindeki bloklar, diğer alt imgedeki iki farklı alt blok ile bağımlı hale getirilmiştir. Bu bağımlılıklar hem saldırılara karşı güvenliği sağlamak amacıyla hem de değişim bölgesini yüksek bir doğrulukta belirlemek amacıyla kurulmuştur. Önerilen teknikte saldırılara karşı tam bir güvenlik seviyesi elde edebilmek için tüm imgeye ve alt imgelere ait sayısal imzalar kullanılmıştır.

Bu imzaların önerilen tekniğin değişim bölgesi belirleme doğruluğunu etkilememesini sağlamak için, tüm imgeye ait sayısal imza her iki alt imge içerisindeki bloklara ve alt imgelere ait sayısal imzalarda her bir alt imgenin kendi blokları içerisine dağıtılmıştır. Önerilen teknikte her bir alt imgedeki bloğun diğer alt imgedeki iki alt blok ile bağımlı hale getirilmesi, bir pikseldeki bozulmanın, pikselin ait olduğu bloğun ve bu bloğun bağımlı olduğu alt blokların

doğrulanmamasını gerektirir. Ancak önerilen teknikte yeni bir doğrulama algoritması geliştirilerek, bir pikseldeki bozulmanın sadece o bloğun yarısının doğrulanamamasına neden olması sağlanmıştır. Bu çözüm değişim bölgesinin 4 kata kadar daha doğru bir şekilde belirlenmesine imkan sağlamaktadır. Dolayısıyla geliştirilen algoritma diğer bloksal bağımlılığa sahip teknikler açısından da önemlidir. Ayrıca bu teknikte iki ayrı sayısal imza algoritmasının kullanılması önerilmiştir. Daha yüksek güvenlik seviyesine ve uzun bit çıkışına sahip sayısal imza algoritması, tüm imgenin ve alt imgelerin doğrulanmasında kullanılırken, daha düşük güvenlik seviyesine ve kısa bit çıkışına sahip sayısal imza algoritması blok ve bu blokların alt bloklarla oluşturdukları bağımlılıkların doğrulanmasında kullanılmıştır. Bu öneri daha küçük blok büyüklüklerinin seçilmesini sağlayarak, tekniğin değişim bölgesi belirleme doğruluğunu artırmıştır.

Önerilen kırılgan filigranlama tekniklerinin performansını değerlendirebilmek amacıyla 5 farklı çözünürlükte ve 19 farklı test imgesi üzerinde deneysel çalışmalar yapılmıştır.

İlk deneysel çalışma önerilen tekniklerin değişim bölgesi belirleme doğruluklarını değerlendirebilmek amacıyla yapılmıştır. Bu deneysel çalışmada test imgeleri, önerilen tekniklerin filigranlama algoritması kullanılarak filigranlanmıştır. Daha sonra filigranlı imgeler üzerinde değişimler yapılarak önerilen tekniklerin doğrulama algoritmaları çalıştırılmış ve değişim bölgesi belirleme doğrulukları incelenmiştir.

İkinci deneysel çalışma önerilen tekniklerin; Kes Yapıştır, Vektör Nicemlemeli Taklit Saldırısı, Doğum Günü Saldırısı ve Aktarıma Dayalı Saldırıya karşı güvenirliğini değerlendirebilmek amacıyla yapılmıştır.

Üçüncü deneysel çalışmada ise daha uzun bit çıkışına sahip, daha güvenilir bir sayısal imza algoritması kullanılarak tekniklerin değişim bölgesi belirleme doğrulukları incelenmiştir. Bu çalışma önerilen tekniklerin daha yüksek kriptografik güvenlik seviyesindeki performanslarını değerlendirebilmek amacıyla yapılmıştır.

Ayrıca önerilen Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniği ile Sıradüzensel Filigranlama Tekniği; saldırılara karşı güvenirlik, aynı güvenlik seviyesinde değişim bölgesi belirleme doğruluğu, imge çözünürlüğünden bağımsız olarak filigranlama, daha yüksek kriptografik güvenlik seviyesi altında değişim bölgesi belirleme doğruluğu nitelikleri göz önünde tutularak karşılaştırılmıştır. Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniği ise diğer bir blok bağımlı yapıya sahip HBC2 Filigranlama Tekniği ile saldırılara karşı güvenirlik, aynı güvenlik seviyesinde değişim bölgesi belirleme doğruluğu, daha yüksek kriptografik güvenlik seviyesi altında değişim bölgesi belirleme doğruluğu nitelikleri göz önünde tutularak karşılaştırılmıştır. Karşılaştırılmalarda önerilen teknikler ile karşılaştırılan tekniklerin aynı filigran niteliklerine sahip olmasına dikkat edilmiştir. Bu nitelikler arasında; filigran olarak

sayısal imzaların kullanılması, filigranın imgenin EÖB düzlemine gömülmesi, filigranlama yordamında bağımlı blok yapısının kullanılması verilebilir.

Yapılan deneysel çalışmaların sonucunda önerilen tekniklerin dört önemli özelliğe sahip olduğu görülmüştür.

1) Önerilen teknikler imge çözünürlüğünden bağımsız olarak değişim bölgesi belirleme niteliğine sahiptirler.

2) Önerilen teknikler saldırılara karşı oldukça güvenilir bir yapıya sahiptirler.

3) Önerilen tekniklerin değişim bölgesi belirleme doğruluğu karşılaştırılan diğer tekniklere göre daha yüksektir.

4) Önerilen tekniklerde yüksek kriptografik güvenlik seviyesine sahip bir sayısal imza algoritması kullanılması durumunda, önerilen tekniklerin değişim bölgesi belirleme doğruluğu karşılaştırılan diğer tekniklere göre oldukça yüksektir.

1.5. Tezin Yapısı

Tezin bundan sonraki bölümleri aşağıdaki yapı içerisinde düzenlenmiştir. Bölüm 2 içerisinde Sayısal Filigranlama'yla ile ilgili ayrıntılı bir bilgi verilmesi amaçlanmıştır. Bu bölümde Sayısal Filigranlama, Gizliyazı ve Bilgi Gizleme arasındaki ilişki incelenmiş ve sayısal filigranların nitelikleri ve uygulama alanları ayrıntılı olarak verilmiştir.

Bölüm 3 içerisinde, Sayısal Filigranlama Sistemi ve sayısal bir imgenin oluşması incelendikten sonra, Sayısal Filigranlama Teknikleri ve bu tekniklere karşı yapılan saldırılar geniş bir literatür taraması yapılarak sunulmuştur.

Bölüm 4, doğrulama uygulamaları için önerilen önemli Kırılgan Filigranlama Teknikleri, bu tekniklere karşı yapılan saldırılar ve saldırılara karşı alınan önlemler incelenerek hazırlanmıştır.

Bölüm 5, önerilen Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniğinden ve bu tekniğin performansını değerlendirebilmek için yapılan deneysel çalışmalardan oluşmaktadır. Ayrıca bu bölümde Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniği ile Sıradüzensel Filigranlama Tekniği'nin karşılaştırılmasına yer verilmiştir.

Bölüm 6 ise, önerilen Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinden ve bu tekniğin performansını değerlendirebilmek için yapılan deneysel çalışmalardan oluşmaktadır. Ayrıca bu bölüm içerisinde Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniği ile literatürde önerilmiş diğer bir blok bağımlı yapıya sahip olan HBC2 Filigranlama Tekniğinin karşılaştırılması yapılmıştır.

Son olarak bölüm 7, önerilen kırılgan filigranlama tekniklerinin katkılarını tartışmaktadır.

2. SAYISAL FİLİGRANLAMA, NİTELİKLERİ ve UYGULAMA ALANLARI

2.1. Sayısal Filigranlama

Tarihsel olarak ilk filigranlar 1282 yılında İtalya'da görülmektedir [62]. Bu filigranlar kağıt kalıplar üzerine ince metal modeller eklenerek yapılmıştır. Böylece kağıt, metalin bulunduğu yerde daha ince ve saydam olmuştur. Ancak 18. yy.'a kadar bu filigranlar, günümüzdeki kullanım amaçlarından farklı olarak kağıt yapraklarının yapıldığı kalıpları teşhis etmek, gizemli işaretleri temsil etmek veya dekorasyon olarak kullanılmışlardır. 18. yy.'da ise filigranlar Avrupa ve Amerika'da ticari marka oluşturmak ve para üzerinde sahtekarlığı önlemek amacıyla kullanılmaya başlanmıştır. Filigran, Almanca bir terim olan *"wassermarke"* kelimesinden türetilmiştir. Muhtemelen damgalar, kağıt üzerindeki suyun etkisine benzediği için bu kelime kullanılmıştır [5].

Türkçe'de filigran kelimesi Fransızca *"filigrane"* sözcüğünden türetilmiştir. Türk Dil Kurumu Sözlüğü'nde bu kelime "Bazı kâğıtların dokusunda bulunan, ancak aydınlığa tutulduğunda görülen çizgi, resim, yazı vb." olarak tanımlanmıştır [63]. Günümüzde kağıt üzerindeki filigranlar genellikle banknot gibi değerli dokümanların sahtesinin yapılmasını önlemek için kullanılırlar. Şekil 2.1.'de 10 YTL'lik banknot ve üzerinde bulunan filigran gösterilmiştir.



Şekil 2.1. Banknot üzerindeki filigranlama örneği

Banknot üzerindeki filigranlama işleminde, filigranın banknot içerisine yerleştirilmesi veya filigranın banknot üzerinde bulunması için özel işlemler kullanılmaktadır. Örneğin banknot içerisindeki filigran, sadece banknot ışığa tutulduğunda görülmektedir. Banknot üzerindeki filigranın görülmesi ile banknotun güvenilir olduğu anlaşılır. Ayrıca, sahte bir banknot içerisine filigranın gömülmesi veya yasal bir banknot içerisinden filigranın silinmesi oldukça zordur.

Sayısal filigranlar ise ses, video ve imge gibi sayısal ortamlarda sıkça kullanılan sayısal verilerin filigranlanması için kullanılırlar [5, 15-17, 64-67, 70-74]. Sayısal Filigranlama, "özgün veriye ilişkin bilgiyi (filigran), özgün verinin algılanabilir kalitesini bozmadan doğrudan doğruya özgün veri içerisine gömen ve gerektiğinde gömülen bilgiyi geri çıkartabilen bir yöntem" olarak tanımlanabilir. Banknot üzerindeki filigranın gömülmesi ve görülebilmesi için nasıl özel bir işlem gerekliyse, sayısal filigranlarında sayısal veriler içerisine gömülebilmesi veya geri çıkartılabilmesi için uygun bir filigran gömücüye ve bulucuya ihtiyaç duyulur. Ayrıca banknotun filigranlanmasında olduğu gibi, sayısal bir filigranın da yetkisiz kişiler tarafından sayısal veri içerisinden silinmesi veya başka bir sayısal veri içerisine gömülmesi mümkün olmamalıdır. Buradaki zorluk derecesi sayısal bir filigranın güvenlik seviyesini belirlemektedir.

2.1.1. Sayısal Filigranlama, Gizliyazı ve Bilgi Gizleme

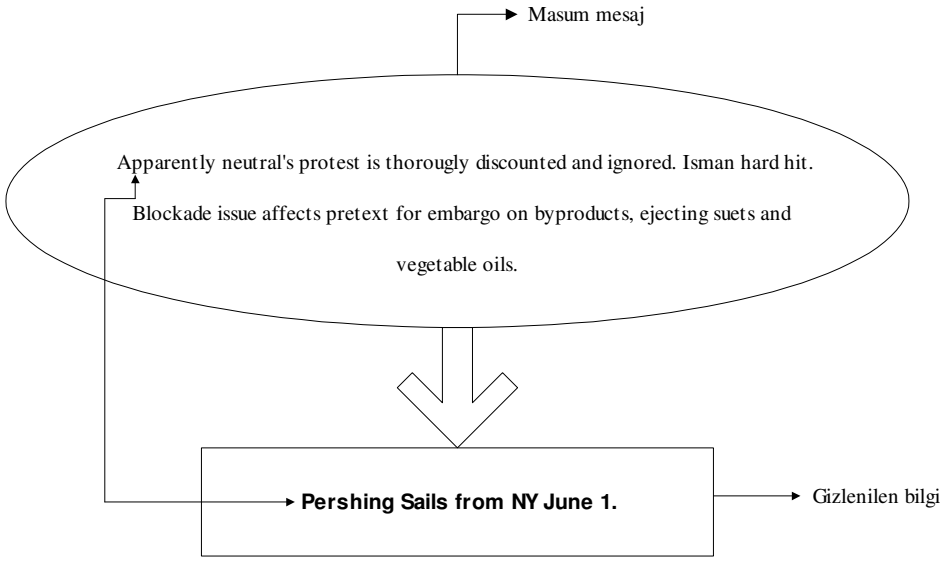
Sayısal Filigranlama, Gizliyazı *(Steganography)* ve Bilgi Gizleme *(Information Hiding, Data Hiding)* alanları birbirleri ile yakından ilişkilidir. Bilgi Gizleme çok genel bir ifadedir ve veritabanı üzerindeki verinin yetkisiz kişiler tarafından kullanılmasının engellenmesi, bir verinin başka bir veri içerisinde saklanarak gizlenmesi veya bir verinin gizlice bir yerden başka bir yere gönderilmesi gibi farklı uygulamalar içerisinde kullanılmaktadır. Bu terim içerisindeki bilgi kelimesi herhangi bir mesaj, gönderici ve alıcının kimlikleri veya korunmak istenen herhangi bir veri olabilir. Gizlilik ile ifade edilmek istenen ise bilginin istenilmeyen üçüncü kişiler tarafından okunmasının veya silinmesinin mümkün olmamasıdır. Ancak gizlilik, gizlenilen bilginin varlığından üçüncü kişilerin haberdar olup olmamasına göre uygulamaları sınıflandırmaktadır. Örneğin hem Sayısal Filigranlama hem de Gizliyazı bir içerik içerisine bir mesajı saklamak için önerilirler [75]. Ancak Gizliyazı Uygulamaları'nda içerik (sayısal imge, video veya ses) içerisine saklanan mesajın varlığından üçüncü kişilerin özellikle haberdar olmaması istenirken, Sayısal Filigranlama Uygulamaları'nda mesajın varlığı açıkça ilan edilmektedir.

Gizliyazı, Yunanca'da örtülü ve yazı anlamlarında kullanılan "*steganos"* ve "*graphia"* kelimelerinin birleşiminden oluşur. Gizliyazı, iletilmek istenen verinin masum gibi görünen bir

taşıyıcı içerisinde saklanarak iletilmesi sanatıdır [75-80]. Gizliyazı'daki en önemli kavram, mesajın varlığının üçüncü kişiler tarafından kesinlikle bilinmemesidir. İkinci dünya savaşında bir Alman casusu tarafından gönderilen masum bir mesaj içerisine gizlenilmiş gizliyazısal bir bilgi şekil 2.2. ile verilmiştir. Burada gizlenilen bilgi, mesaj içerisindeki her bir kelimenin ikinci harflerinin çıkartılıp birleştirilmesiyle elde edilmektedir [81].

Kriptografi bir mesajı gizlemek için başvurulan diğer bir bilim dalıdır [1-4]. Kriptografi Bilimi'nde mesaj bir kriptolama fonksiyonu kullanılarak anlamsız bir karakter dizisine dönüştürüldükten sonra gönderilir. Bu karakter dizisi, kriptolu metin olarak tanımlanır. Kriptografi Bilimi'nde gönderim esnasında üçüncü kişilerin kriptolu metni görmesinde bir sakınca görülmezken, Gizliyazı'da mesajın varlığından üçüncü kişilerin kesinlikle şüphelenmemesi gerekir.

Cox ve diğ. [5], Bilgi Gizleme'yi tablo 2.1.'de verildiği gibi dört sınıf içerisinde incelemiştir. Bu sınıflandırma mesaj ile içerik arasındaki ilişkinin olup olmaması ve mesajın varlığının bilinip bilinmemesi göz önüne tutularak yapılmıştır.



Şekil 2.2. Gizliyazısal mesaj

Tablo 2.1. Bilgi gizlemenin sınıflandırılması

	İçerik Bağımlı Mesaj	**İçerik Bağımsız Mesaj**
Mesaj Varlığı Gizlenilen	Gizliyazısal Filigranlama	Gizli Haberleşme
Mesaj Varlığı Bilinen	Filigranlama	Açık Gömülü Haberleşme

Sayısal Filigranlama ve Gizliyazı bir içerik içerisine bir mesajı yerleştirmeyi amaçladıkları için önerilen tekniklerin çoğunu paylaşırlar. Ancak aralarında dört önemli farklılık mevcuttur:

- Sayısal Filigranlama'da içerik içerisine gömülecek mesaj, içerik üzerindeki telif hakkını gösteren bir logo, içeriğin bütünlüğünü doğrulayan bir sayısal imza veya yasal alıcıyı tanımlayan bir veri gibi içerik ile ilişkili bir bilgi taşır. Gizliyazı'da ise mesaj içerikten tamamen bağımsız herhangi bir bilgi olabilir.
- Sayısal Filigranlama'da gömülen mesajın varlığı açıkça bildirilirken, Gizliyazı'da mesajın varlığından üçüncü kişilerin haberdar olmaması gerekir.
- Sayısal Filigranlama ve Gizliyazı güvenlik kriterine farklı bakış açısıyla yaklaşırlar. Sayısal Filigranlama için güvenlik, gömülen mesajın telif hakkı sahibi dışında başka biri tarafından değiştirilmesinin veya silinmesinin mümkün olmamasıdır. Gizliyazı'da ise güvenlik gömülen mesajın gizliliğinin sürdürülmesi olarak tanımlanır.
- Gizliyazı'da genellikle bir gönderici ve bir alıcı mevcutken, Sayısal Filigranlama'da birden fazla gönderici ve alıcı mevcut olabilir.

Sayısal Filigranlama günümüzde bir çok uygulamada kullanılmaktadır [5, 14-18]. Her bir uygulamanın kendine özgü amaçları ve nitelikleri mevcuttur. Uygulamalar için önerilen Sayısal Filigranlama Teknikleri, nitelikler göz önünde tutularak değerlendirilir.

2.2. Sayısal Filigranlama Uygulamaları

Sayısal Filigranlama Teknikleri gürbüzlük *(robustness)*, sadakat *(fidelity)*, veri yükü *(payload)* vb. gibi ortak özellikler göz önünde tutularak değerlendirilir. Ancak bu özelliklerin bağıl önemi uygulamaya bağımlı olduğu için, ilk olarak uygulamaların tanımlanması gerekir. Aşağıda sayısal filigranların kullanıldığı farklı uygulama alanları verilmiştir.

2.2.1. Telif Hakkı'nın Teşhis Edilmesi ve Kanıtlanması

Telif Hakkı Uyarısı, içerik üzerindeki telif haklarını güvence altına almamasına rağmen hala kullanılmaktadır. Genellikle telif hakkı uyarısı "© tarih, sahip" formunda oluşturulur. Kitap ve imgeler üzerindeki telif hakkı uyarısı içerik üzerinde açıkça görülürken, filmlerde genellikle tanıtım yazısının sonunda, ses kaseti ve CD'lerde ambalaj üzerinde görülmektedir. Ancak bu formda oluşturulan telif hakkı uyarıları korunan içerik üzerinden kolaylıkla silinebilmektedir [14]. Şekil 2.3'de telif hakkı uyarısının imgenin görünür kalitesinde bir bozulmaya neden olmadan silinmesi gösterilmiştir.



(a) **(b)**

Şekil 2.3. Telif hakkı uyarısının silinmesi **(a)** Özgün imge **(b)** Değiştirilmiş imge

Sayısal filigranların, sayısal bir içerik üzerindeki telif haklarını koruma uygulamalarında kullanılması oldukça yararlı olabilir. Bu uygulamalarda sayısal içerik içerisine gömülecek olan filigran telif hakkı sahibinin kimliği ve telif hakkı tarihi gibi içerik üzerindeki telif hakkını tanımlayan bir bilgiden oluşur. Telif hakkı ile ilgili bir anlaşmazlığın oluşması durumunda, telif hakkı sahibi içerik içerisine gömdüğü filigranı bularak içerik üzerindeki telif haklarını koruyabilir. Bu uygulamalarda, içeriğin görsel kalitesini etkilemeden içerik içerisine gömülen filigranın değiştirilmesi veya silinmesi çok zor olmalıdır.

2.2.2. Yayın İzleme

Japonya'da 1997 yılında televizyon reklamlarına ilişkin büyük bir skandal ortaya çıkmıştır. Bu skandalda, yayıncıların aynı yayın zamanında birden fazla reklamı kabul ettiği belirlenmiştir. Dolayısıyla reklamcılar yayınlanmayan binlerce reklam için ödeme yapmışlardır

[5, 82]. Yayın izleme ile ilgilenen bireysel kişiler ve çeşitli organizasyonlar mevcuttur. Örneğin reklamcılar, yayıncılardan aldıkları yayın zamanlarını garanti altına almayı, müzisyenler gibi telif hakkı sahipleri gerçek telif hakkı ücretlerini almayı ve illegal olarak yayınlanan içeriklerini korumayı istemektedirler.

Yayınları denetlemek için, yayınları izleyen ve gördüklerini kaydeden gözlemleyicilerin istihdam edilmesi pahalı ve hata oranı yüksek bir çözümdür. Bu yüzden gözlemleyiciler yerine, otomatik denetleme sistemlerinin geliştirilmesi istenmektedir. Otomatik denetleme sistemleri pasif ve aktif olarak sınıflandırılabilir [5]. Pasif bir sistemde bir bilgisayar, yayınlanan içeriği alır ve veritabanında kayıtlı olan içerikler ile eşleştirmeye çalışır. Eşleşme olduğunda, şarkı, film, TV programı veya reklam gibi içerik tanımlanarak denetleme yapılır. Ancak böylesi bir sistemin pratik bir kullanımı yoktur. Çünkü, milyonlarca bitten oluşan bir yayının veritabanı içerisinde eşleştirilmesi mümkün değildir. Bu yüzden pasif sistemler yayınlara ait imzalar oluşturarak bu imzaları eşleştirmeye çalışırlar. Ancak bu durumda da yayınlanan içerikte bir miktar bozulma olduğunda, imzaların eşleşmemesi problemi ile karşılaşılır.

Aktif sistemlerde, yayınlanan sinyalin ayrı bir alanı içerisine, yayınlanan içeriği tanımlayan bir bilginin yerleştirilmesi önerilmektedir. Ancak, ayrı bir alan içerisine tanımlama bilgisinin yerleştirilmesi önemli problemlere neden olmaktadır [5]. Filigranlama, yayınlanan içeriğe ait bilgiyi doğrudan doğruya içerik içerisine gömdüğü için bu probleme çözüm olabilir. Yayınlanan içeriğe ait bilginin filigran olarak gömülmesiyle, reklamcılar ve telif hakkı sahipleri aktif denetleme sistemleri içerisine bir filigran bulucuyu dahil ederek programlarının gerçekten yayınlanıp yayınlanmadığını kontrol edebilirler [83].

2.2.3. Kopya Koruma

Kriptografi illegal kopyalamaya karşı önerilen güvenilir bir bilim dalıdır. Kriptografi'de, korunmak istenen içerik eşsiz bir anahtar kullanılarak kriptolanır. Kriptolanmış veriyi, kripto çözme anahtarına sahip olmayan birinin çözmesi oldukça zordur. Anahtar ise sadece içeriği satın almak isteyen yasal alıcılara verilir. Kriptolanmış bir içeriği çözüp kopyalamak isteyen bir kişi için üç yol mevcuttur. İlki ve en zor olanı bir anahtara sahip olmadan içeriği çözmeye çalışmaktır. Günümüzde 1024 ve 2048 bitlik anahtar uzunluklarının kullanılmasıyla, bu çözme şekli neredeyse imkansız bir hale getirilmiştir. Birinciye göre daha kolay olduğu düşünülen diğer bir yaklaşım ise, tersine mühendislik yapılarak geçerli bir anahtar elde etmeye çalışmaktır. Üçüncü ve en kolay yol ise yasal bir alıcı olarak bir anahtar elde etmek ve bu anahtarı kullanarak kriptolanmış içeriği çözdükten sonra kopyalamaktır. Kriptolama sayısal içeriğin dağıtımı ve tesliminde kopyalamaya karşı oldukça etkili olmasına

rağmen, kripto çözüldükten sonra herhangi bir koruyuculuğa sahip değildir. Bu yüzden kriptolanmış içerik çözüldükten sonra, içeriği koruyacak yöntemlere ihtiyaç duyulmaktadır [5].

Kopya koruma uygulamalarında filigran, filigranlı içeriğin kopyalamaya karşı korumalı olduğunu bildirir. Ancak kopyalamayı önlemek için, kayıt cihazları ve oynatıcı cihazların filigranlama sistemi ile birlikte çalışması gerekir. Kopya koruma protokolüne riayet eden bir cihaz, içerik içerisine gömülmüş olan filigranı bulur ve içeriğin kullanılmasına veya kopyalanmasına müsaade etmez. Bazı kopya koruma uygulamalarında kullanıcıya, özgün içerikten sadece bir kopya hakkı verilir. Ancak, kopyalanmış içerikten tekrar kopya oluşturmasına izin verilmez [84]. Böylesi sistemlerde gömülen filigran, "kopyalamaya daima izin ver", "bir kopya oluşturmaya izin ver" ve "kopyalamaya izin verme" gibi bir bilgiyi temsil eder.

2.2.4. İçeriğin Doğrulanması

Sayısal içerik üzerinde bir değişim yapmak oldukça kolaydır. Örneğin şekil 2.4.(a)'da verilen özgün imge, Photoshop imge işleme programı kullanılarak değiştirilmiş ve şekil 2.4.(b)'de görülen imge oluşturulmuştur. Sayısal bir imge polis araştırmasındaki veya mahkemedeki kanıtın önemli bir parçası ise, imgelerin fark edilemeyecek derecede kolaylıkla değiştirilebilmesi ile önemli bir sorun ortaya çıkmaktadır. Burada değişim ile ifade edilmek istenen bir nesnenin eklenmesi, değiştirilmesi, maskelenmesi veya silinmesidir. Aynı durum sayısal video ve ses verisi içinde geçerlidir [5].



(a) **(b)**

Şekil 2.4. İçeriğin doğrulanması **(a)** Özgün imge **(b)** Değiştirilmiş imge

Kriptografi Bilimi'nde mesajları doğrulamak için Sayısal İmzaların kullanılması önerilir [1-4]. Sayısal İmza, içerikten elde edilen özütün kriptolanmasıyla oluşur. Doğrulama uygulamalarında genellikle asimetrik bir kriptolama algoritması kullanılır. Asimetrik yapıda, sayısal bir imza oluşturmak için kullanılan anahtar ile, imzayı çözmek için kullanılan anahtar birbirinden farklıdır. Sayısal imzayı oluşturmak için kullanılan anahtar, yalnızca mesajı oluşturan kişi tarafından bilinir. Böylece yetkisiz kişiler tarafından mesaja ait yeni bir sayısal imzanın oluşturulması engellenir. Sayısal İmzaların sayısal fotoğraf makineleri içerisinde kullanılması önerilmiştir [85, 86]. Bu öneride, sayısal imzayı oluşturmak için ihtiyaç duyulan anahtar sadece kamera tarafından bilinmektedir. Böylece, imgenin sayısal imzayla eşleşen bir kopyası bulunduğunda, bulunan imgeyle çekilen imgenin birebir aynı olduğundan emin olunur.

Sayısal imgelerin doğrulanmasında, sayısal imge verisinden elde edilen sayısal imzalar ya imgenin başlığı içerisine gömülür yada ayrı bir dosyada tutulur. Ancak sayısal imzaların bu şekilde kullanılması problemlere neden olur. Örneğin imge başlığı içerisine yerleştirilen sayısal imza, imgenin format değişimi ile birlikte kaybolur. Yine sayısal imzanın ayrı bir dosyada tutulması kayıt ortamı için ilave bir yer ihtiyacı, ağ ortamı için ise ilave bir bant genişliği gerektirir. Ayrıca sayısal imzanın tutulduğu dosyanın kaybolması veya silinmesi durumunda, imgenin doğrulanması mümkün olmayacaktır.

Daha iyi bir çözüm ise sayısal imzanın filigran olarak imge içerisine gömülmesi ile sağlanabilir [37-40, 42, 43, 51, 52]. Böylece sayısal imza imge ile birlikte kalır ve istenildiğinde çıkartılarak imgenin aslıyla aynılığı doğrulanır. İmgenin aslıyla aynılığının doğrulanmasında gömülen filigran, imgenin yeniden hesaplanan sayısal imzasını etkilememesi gerekir. İçerik iki parçaya ayrılarak bu problemin üstesinden gelinebilir. İlk parça imzanın hesaplanacağı ve ikinci parçada imzanın gömüleceği yer olarak seçilebilir. Diğer bir öneri ise imgenin en önemli bit düzlemleri kullanılarak hesaplanılan imzanın, imgenin EÖB düzlemi içerisine gömülmesidir [37-40, 42, 43, 51, 52]. İmgenin aslıyla aynılığını doğrulamak için imge içerisine gömülen filigranlar, doğrulama damgaları olarak da bilinir. Sayısal Filigranlama'da, içerik üzerinde yapılan en küçük değişiklikte bile geçersiz olması için tasarlanan doğrulama damgaları kırılgan filigranlar olarak adlandırılır.

2.2.5. İçerik İzleme

İçerik İzleme Uygulamaları'nda, içerik sahibi içeriğin her bir kopyası içerisine kopyayı vereceği kişiyi tanımlayan bir bilgiyi filigran olarak gömer. Filigran olarak gömülen bilgi, farklı kişileri veya kuruluşları tanımlayacağı için sürekli değişir. Filigranlı içeriği alan kişi veya kuruluş, filigranlı içeriği tekrar kopyalayacaksa, kopyayı alacak yeni alıcının bilgisini filigran

olarak filigranlı içerik içerisine eklemeye devam eder. Bu şekilde, kopyanın yasal olarak kimlere verildiği takip edilebilir. Şüpheli bir kopya bulunduğu zaman, gömülen filigran bulunarak şüpheli kopyayı dağıtan kaynak ortaya çıkarılabilir. Literatürde bu filigranlar parmak izi damgaları olarak da bilinir [14, 87].

İçerik izleme ile ilgili bir uygulama, filmlerin günlük çekimlerinde görülebilir. Bir filmin yapımı esnasında, çekilen günlük kesitler filmin prodüksiyonuyla ile uğraşan birkaç kişiye dağıtılır. Bu günlük çekimlerin son derece gizli kalması gerekir, fakat bunlar bazen basına sızdırılır. Bu durumda, stüdyolar hızlıca sızıntının kaynağını teşhis etmeye çalışırlar. Eğer günlük kopyaların her birinde alıcıyı tanımlayan bir filigran kullanılırsa, sızıntının kaynağını teşhis etmek mümkün olmaktadır [5].

2.2.6. İçerik Arşivleme

Sayısal bir içerik genellikle bir dosya ismi ile etiketlenerek saklanır. Ancak kullanım sırasında dosya ismi kolaylıkla değiştirilebildiği için, bu yöntem medikal imgelerin arşivlenmesi gibi uygulamalarda oldukça kullanışsızdır. Sayısal Filigranlama imge, video ve ses gibi sayısal içeriği arşivlemede, sınıflandırmada veya düzenlemede kullanılabilir [16, 88]. Bu uygulamalarda filigran içeriği tanımlayan bir seri numarası, ISBN veya bir tanıtıcıdan oluşur. Dosya ismindeki herhangi bir bozulmada, gömülen filigran içerik içerisinden çıkartılarak yeniden arşivlenebilir.

2.3. Sayısal Filigranların Nitelikleri

Sayısal Filigranlama Tekniklerinin performansı gürbüzlük, aslına benzerlik gibi bazı nitelikler göz önünde tutularak değerlendirilir. Bu niteliklerin bağıl önemi uygulamaya bağımlıdır.

2.3.1. Gürbüzlük

Gürbüzlük *(robustness)*, sıkça rastlanılan imge işleme operasyonlarına karşı gömülen filigranın direnci olarak tanımlanır. İmge üzerinde sıkça uygulanan işlemlere örnek olarak imgenin kayıplı olarak sıkıştırılması, filtrelenmesi, baskılanması ve taranması, iyileştirilmesi ve geometrik dönüşümler (döndürme, ölçeklendirme, taşıma, vb.) verilebilir. İmge içerisine gömülen filigranın tüm bu işlemlere karşı gürbüz olması gerekmez. Filigrandan sadece, gömülmesi ile bulunması arasında gerçekleşebilecek işlemlere karşı gürbüz olması istenilir.

Bazı durumlarda ise filigranın gürbüzlüğü hiç istenilmez. Örneğin imge doğrulama uygulamaları için önerilen Kırılgan Filigranlama Teknikleri'nde özellikle filigranın gürbüz olmaması istenir [34-54]. Sıkça rastlanılan imge işleme operasyonlarından bazıları aşağıda verilmiştir.

2.3.1.1. İmge Sıkıştırma

Sayısal bir imge oluşturmak için 2 boyutlu ışık yoğunluk fonksiyonu örneklenip nicemlendiğinde, üretilen veri miktarı çok büyüktür. Sayısal imgeyi temsil eden bu büyük miktardaki veri, ham imge verisi olarak tanımlanır. Ham imge verisinin iletilmesi, depolanması ve uygulamalarda kullanılması oldukça elverişsizdir. Bu elverişsizliği gidermek için sayısal imgeyi temsil eden ham veri miktarının azaltılması istenilir ki bu imgenin sıkıştırılmasıyla sağlanır. İmgenin sıkıştırılması kayıplı ve kayıpsız olarak yapılabilir. Kayıplı sıkıştırmada genellikle JPEG *(Joint Photographic Experts Group)* sıkıştırma yöntemi kullanılır. Sıkıştırma oranı, %5 ile %100 aralığında seçilen bir kalite faktörüne göre belirlenir.

İmgenin kayıpsız olarak sıkıştırılması gömülen filigranı etkilemez. Ancak kayıplı sıkıştırmada, imge içerisine gömülen filigran seçilen sıkıştırma parametresinin değerine bağımlı olarak etkilenir. Daha yüksek sıkıştırma oranının seçilmesi gömülen filigranın daha fazla etkilenmesine neden olur.

2.3.1.2. Baskılama ve Tarama

Sayısal bir imgenin baskılanarak fiziksel formdaki kullanımı oldukça yaygındır. Baskılama, imgenin sayısal formdan analog forma dönüştürülmesidir. Tarama ise fiziksel formdaki imgenin sayısallaştırılmasıdır. Baskılama ve tarama işlemlerinde sayısal-analog ve analog-sayısal dönüştürücüler kullanılır. Bu dönüştürücüler, baskılama ve tarama işlemlerinde kullanılan cihazların karakteristiğine bağımlı olarak imgeler üzerinde önemli bozulmalara neden olurlar. Bazı uygulamalarda, imge içerisine gömülen filigranın özellikle bu tür bozulmalara karşı gürbüz olması istenir.

2.3.1.3. İmge İyileştirme

İmgenin oluşması esnasında, algılayıcı cihaz lensinin yanlış odaklanması veya cihazın hareket ettirilmesi imgenin bulanıklaşmasına veya geometrik deformasyonuna neden olur. İmge iyileştirmenin amacı deformasyona uğramış imgelerin görsel kalitesini iyileştirmektir. İmge

iyileştirme teknikleri, uzaysal alan ve frekans alan imge iyileştirme teknikleri olarak iki ayrı kategori içerisinde sınıflandırılır. Uzaysal alan imge iyileştirme teknikleri doğrudan doğruya bir imgenin piksellerinin işlenmesine dayanırken, frekans alan teknikleri imgenin frekans alanındaki katsayılarının işlenmesine dayanır. Uzaysal alan imge iyileştirme teknikleri arasında zıtlık germe, histogram eşitleme ve maskeleme örnek olarak verilebilir. İmgenin frekans alanı kullanılarak imgenin iyileştirilmesinde genellikle doğrusal ve doğrusal olmayan filtreler kullanılmaktadır.

2.3.1.4. Geometrik Bozulmalar

İmge işleme uygulamalarında taşıma, ölçeklendirme ve döndürme gibi geometrik dönüşümler sıkça kullanılır. Bu dönüşümler filigranlı imge içerisindeki filigranın pozisyonunun değişmesine neden olur ve filigran bulucunun filigranı bulamamasına yol açar.

2.3.2. Aslına Benzerlik

Sayısal bir filigranın fark edilmezliği, özgün imge ile filigranlı imge arasındaki algısal benzerlik *(fidelity)* ile tanımlanabilir. Özgün imge ile filigranlı imge yan yana konulduğunda, İGS tarafından filigranlı imgeden özgün imgenin ayırt edilememesi, gömülen filigranın gerçekten fark edilemez olduğunu gösterir [65]. Ancak filigranlı imge sahiplerinin özgün imgeye erişiminin olmaması durumunda, herhangi bir karşılaştırma yapılamaz. Bu durumda, özgün imge içerisine gömülen filigranın fark edilememesi yeterlidir [89]. İGS bir filigranın fark edilip edilmediğini belirlemede temel ölçüttür [5]. Ancak uygulamalarda İGS'nin kullanılması pratik ve kolay değildir. Özgün imge ile filigranlı imge arasındaki fark edilebilirliği ölçmede Ortalama Karesel Hata (OKH) kullanılabilir. OKH, eşitlik 2.1. kullanılarak hesaplanır.

$$OKH(I,I') = \frac{1}{MN}\sum_{x=0}^{M}\sum_{y=0}^{N}\left(I(x,y) - I'(x,y)\right)^2 \tag{2.1}$$

Burada I', filigranlı imgeyi tanımlamaktadır. Ancak OKH'ın fark edilebilirliği ölçmede oldukça zayıf bir tahmin sağladığı bilinmektedir [5, 90]. Algısal benzerliği ölçmek için kullanılan diğer bir yöntem ise decibel cinsinden Sinyal-Gürültü Oranı'nın (SGO) ölçülmesidir. SGO filigranlı imgenin özgün imgeye göre ne kadar gürültülü olduğunu ölçer. SGO eşitlik 2.2. ile verilmiştir.

$$SGO(I,I') = 10\log_{10}\left(\frac{(I_{\max})^2}{OKH(I,I')}\right) \quad (2.2)$$

Burada $I_{\max}$, I ve I' imgeleri içerisindeki en yüksek parlaklık oranına sahip pikselin değeridir. İGS'nin karakteristiği göz önünde tutularak, OKH ve SGO'ya göre algısal benzerliği daha iyi ölçen yöntemler önerilmiştir [91, 92].

2.3.3. Veri Yükü ve Kapasite

Gömülen filigran tarafından temsil edilen bilgi miktarı veri-yükü *(payload)* olarak tanımlanır. Sayısal bir imge için veri yükü imge içerisine gömülen bit sayısıdır. Daha yüksek veri yüküne sahip bir teknik, sabit bir sinyal içerisine daha fazla miktarda bilgi gömebilir. Bu genellikle istenilen bir özelliktir. Özgün imgenin karakteristiği müsait olduğunda, Uyarlamalı Filigranlama Teknikleri daha fazla bilgi gömme yeteneğine sahiptirler [93-95]. Bu tekniklerde bulucu, sadece bir filigranın mevcut olup olmadığını belirlemeye çalışır.

2.3.4. Gömü Etkinliği

İmge filigranlama sistemlerinde, bulucu kendisine giriş olarak verilen imgelerin bir filigran barındırıp barındırmadığını belirlemeye çalışır. Bulucunun çıkışı pozitif bir bulma ile sonuçlandığı zaman, imge artık filigranlı bir imge olarak tanımlanır. Bir filigranlama sisteminin gömü etkinliği, gömücünün verilen bir imgeyi filigranlamadaki başarısı ile ölçülür. Diğer bir ifadeyle etkinlik, gömücünün imge içerisine filigranı gömdükten hemen sonra bulucunun imge içerisinde filigranı bulma oranıdır. İmge filigranlama sistemlerinde genellikle %100'lük bir etkinlik oranı istenir. Ancak bazı uygulamalarda diğer nitelikler ile ilgili daha iyi bir performans elde etmek için etkinlikten bir miktar fedakarlık edilebilir [5].

2.3.5. Yanlış Pozitif, Yanlış Negatif ve Bit Hata Oranı

Bulucunun filigranlanmamış bir imge içerisinde bir filigran bulması yanlış pozitif olarak tanımlanır. Yanlış pozitif olasılığı, bulucunun belirli bir sayıda çalışması sonucunda beklenilen yanlış pozitiflerin sayısı ile belirlenir. Yanlış negatif ise filigranlı bir imge içerisinde bulucunun filigranı bulamamasıdır. Yanlış negatif olasılığı, bulucunun belirli bir sayıda çalışması sonucunda beklenilen yanlış negatiflerin sayısı ile tanımlanır. Bu olasılıkların bağıl

önemi uygulamaya bağımlıdır. Örneğin DVD video uygulamalarında, yanlış pozitif olasılığının oldukça düşük olması istenir [84].

Bulucu filigranı bulduktan sonra, filigran tarafından temsil edilen bilginin çözülmesi gerekir. Hatalı bit, çözülen bit değerinin gömülenden farklı olmasıyla oluşur. Belirli bir bit uzunluğu içerisinde bulunan hatalı bit sayısı bit hata oranını tanımlar.

2.3.6. Güvenlik

Bir filigranın güvenliği düşmanca saldırılara karşı direnme yeteneğidir [5, 96-98]. Düşmanca bir saldırı filigranı geçersiz veya kullanışsız bir duruma getirmek için yapılan herhangi bir işlem olabilir. Saldırılar, yetkisiz silme, yetkisiz yerleştirme ve yetkisiz bulma olarak sınıflandırılır. Yetkisiz silme ve yerleştirme, filigranlı imgeyi değiştirdiği için aktif; yetkisiz bulma ise herhangi bir değişime neden olmadığı için pasif saldırı olarak bilinir. Bu saldırıların bağıl önemi uygulamaya bağımlıdır ve bazı uygulamalar için güvenlik göz önünde tutulması gereken bir nitelik bile olmayabilir.

Yetkisiz silme bir içerik içerisindeki filigranın bulunmasını önlemek amacıyla yapılan saldırılardır. Bu saldırı türü, yok eden ve maskeleyen saldırılar olarak iki kategori içerisinde incelenebilir [5]. Yok eden saldırılarda filigranın bulunması olanaksızdır. Bu saldırının amacı, algısal olarak özgün içeriğe benzeyen fakat içerisinde bir filigran olmayan yen bir içerik yaratmaktır. Ancak yeni içerik oluşturulurken, saldırının neden olduğu bozulmanın aslına benzerlik sınırlaması içerisinde kalması gerekir. Buradaki aslına benzerlik miktarı, filigranlı içerik ile saldırı sonucu oluşturulan içeriğin algısal olarak ayırt edilememesi ile ölçülür. Bu miktar, filigranın gömülmesi esnasında duyulan aslına benzerlik ihtiyacından daha azdır [99, 100].

Maskeleyen saldırılardaki amaç ise bir filigranın yok edilmesi değil, mevcut bulucular ile bulunmasını engellemektir. Örneğin imge filigranlama sistemlerindeki bulucuların çoğu, imge çok az miktarda döndürüldüğünde filigranı bulamayabilir. Böylece, bir saldırgan imgeyi algılanamayacak bir miktarda döndürdüğünde, döndürme işlemine duyarlı bir bulucu filigranı bulamaz. Ancak döndürmenin etkisini yok edecek bir filigran bulucu, filigranı hala bulabilir.

Yetkisiz yerleştirme saldırı türünde, saldırgan içerik içerisindeki filigranı silmek yerine bulucunun bulup doğrulayacağı bir filigran yerleştirmeyi amaçlar. Doğrulama uygulamalarında en çok kaygı duyulan saldırı türüdür. Eğer saldırgan geçerli bir doğrulama izini gömebilirse, bulucunun sahte veya değiştirilmiş içeriği doğrulamasını sağlayabilir.

Yetkisiz bulma veya pasif saldırılar üç seviyede incelenmektedir [5]. En üst seviye bir saldırganın içerik içerisinden filigranı bulup çözmesinden oluşurken, orta seviyede saldırgan

filigranı bulur, fakat çözme işlemini gerçekleştiremez. En alt seviyede ise saldırgan filigranın varlığından haberdardır, fakat ne içerik içerisinde filigranı bulabilir nede filigranı çözme işlemini gerçekleştirebilir.

2.3.7. Bilgili veya Bilgisiz Buluculu Sistemler

İçerik İzleme veya Telif Hakkı Koruma gibi uygulamalarda, filigranlama sistemindeki çıkarım algoritması filigranı bulmak için özgün imgeyi kullanabilmektedir. Bu sistemlere bilgili buluculu, unutkan olmayan veya özel *(Informed detection, non-oblivious, private)* filigranlama sistemleri denir. Filigran, filigranlı imgeden özgün imge çıkarılarak elde edildiği için, bu sistemlerdeki bulucu performansı oldukça yüksektir. Ancak kopya koruma, aslıyla aynılığı kanıtlama gibi bir çok filigranlama uygulamasında bulucunun özgün içeriğe erişimi mümkün değildir. Bu sistemlerde filigranın çıkarımı bilgili buluculu sistemler göre daha zordur. Bu tür sistemler ise kör buluculu, unutkan veya genel *(blind, oblivious, public)* filigranlama sistemleri olarak adlandırılır [17].

2.3.8. Kriptolama ve Filigranlama Anahtarları

Kriptografi Bilimi'nde anahtarlar kullanılmadan önce, kriptolama algoritmalarının güvenliği algoritmaların gizliliğinin korunmasıyla sağlanırdı. Ancak algoritmaların gizliliğini koruyarak güvenliği sağlamak iki önemli probleme yol açmaktaydı. Birincisi, algoritmanın gizliliğinden şüphe edilmesi durumunda, tamamen yeni bir algoritmanın geliştirilmesi gerekmekteydi. İkincisi, algoritmanın gizli tutulması algoritmanın araştırmacılar tarafından yeterince araştırılmamasına ve dolayısıyla güvenlik açıklarının tam keşfedilememesine yol açıyordu. Günümüz kriptolama algoritmalarında güvenlik, Kerckhoffs'un ilkelerine dayanmaktadır. Kerckhoffs [101], saldırganın gizli anahtar olarak adlandırılan bir parametrenin dışında, algoritmanın tüm ayrıntılarını bildiğini farz ederek bir kriptolama sistemi tasarlamayı önermiştir. Anahtar bir mesajın nasıl kriptolanacağını tanımlayan gelişigüzel oluşturulmuş bit dizileridir. Kriptografi Bilimi'nde, mesajı kriptolamak ve çözmek için aynı anahtarı kullanan algoritmalar özel anahtarlı algoritmalar, mesajı kriptolamak ve çözmek için farklı anahtarlar kullanan algoritmalarda genel anahtarlı algoritmalar olarak adlandırılır.

Sayısal bir filigranlama sisteminin güvenliğini sağlamak için de Kerckhoffs'un ilkelerinin kullanılması önerilmektedir [102]. Bu öneride filigranı oluşturmak, gömmek, bulmak ve çözmek için kullanılan tüm algoritmalar açıklanırken, bu algoritmalarda kullanılan üretme

anahtarı, gömme anahtarı, bulma anahtarı ve çözme anahtarı gibi parametrelerin saklı kalması gerekmektedir.

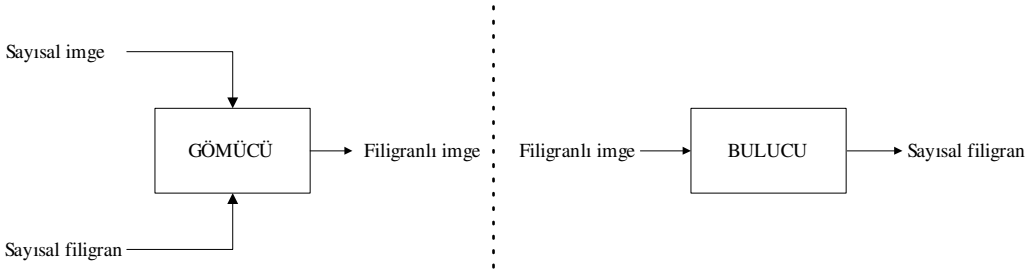
2.3.9. Maliyet

Farklı uygulamalar, farklı hızlarda çalışan gömücü ve bulucular gerektirirler. Örneğin yayın izleme uygulamalarında, hem gömücülerin hem de bulucuların gerçek zamanlı çalışması, gömücülerin yayını yavaşlatmaması ve bulucunun yayınlanan içerik esnasında denetlemeyi yapması istenmektedir. Bu uygulamanın tam aksine, sahipliğin doğrulanması veya kanıtlanması uygulamalarında, gömücünün filigranı gömmesinde veya bulucunun filigranı bulmasında herhangi bir zaman kısıtlaması yoktur. Önemli olan filigranın doğru olarak bulunup, çözülmesidir. Ayrıca uygulamalara bağımlı olarak, filigran gömücü ve bulucu sayısı da değişebilir. Örneğin yayın izleme uygulamalarında, gömücü sayısı içeriği yayınlayacak kanal sayısına göre seçilirken, bulucu sayısı içeriği denetleyecek olan kullanıcı sayısına göre belirlenir. Kopya koruma uygulamalarında ise çok az sayıda gömücüye, fakat milyonlarca bulucuya ihtiyaç duyulmaktadır.

3. SAYISAL FİLİGRANLAMA SİSTEMİ

Sayısal Filigranlama Sistemleri'nde imge, video ve ses gibi sayısal içerik genellikle ana veri *(host data)* veya örten veri *(cover data)* olarak adlandırılır. Tezin bundan sonraki bölümlerinde ana veri olarak sayısal imgeler kullanıldığı için bölüm 3.1.'de sayısal imgelerle ilgili temel bilgiler verilmiştir. Ana verinin korunması veya doğrulanması amacıyla ana veri içerisine gömülen bilgi "filigran" olarak tanımlanır. Filigranlı veri, ana verinin değerlerinin filigranı barındıracak şekilde değiştirilmesiyle elde edilir. Ancak bu değişim ana veri üzerinde bir bozulmaya neden olur. Bu bozulma, gömü bozulması olarak bilinir. Görünmez Filigranlama Teknikleri'nde bu bozulmanın İnsan Görme Sistemi (İGS) tarafından algılanmaması istenir [93-95, 103-107].

Sayısal Filigranlama Sistemleri genel olarak filigranın üretilmesi, gömülmesi ve bulunmasından oluşur. Ana veriyi koruyacak veya doğrulayacak olan filigran uygulamaya bağımlı olarak ikili sayı sisteminde bir veri, imgesel bir logo, sayısal bir imza veya sahte rasgele üretilmiş gaussian veya bir biçimli *(uniform)* bir dağılıma sahip bir sayı dizisi olabilir. Sayısal Filigranlama Teknikleri'nde filigran üretildikten sonra, ana veri içerisine filigranın gömülmesi için bir gömücüye ve filigranlı veri içerisinden filigranın çıkarılması içinde bir bulucuya ihtiyaç duyulur. Şekil 3.1'de bir gömücü ve bir bulucudan oluşan basit bir Sayısal Filigranlama Sistemi'nin yapısı gösterilmiştir.



Şekil 3.1. Sayısal Filigranlama Sistemi'nin genel yapısı

3.1. Sayısal İmge

Bu tez çalışmasında ana veri olarak sayısal imgeler kullanılmıştır. Sayısal bir imge; sayısal fotoğraf makineleri, sayısal kameralar, tarayıcılar v.b. cihazlar vasıtasıyla veya bilgisayar ortamında bir imge işleme programı kullanılarak oluşturulabilir.

3.1.1. İmgenin Algılanması ve Elde Edilmesi

Bir imge, iki boyutlu bir fonksiyon olarak $f(x, y)$ biçiminde tanımlanır. Burada (x, y), uzaysal koordinatları; f ise, (x, y) koordinatlarındaki ışıklandırma (z) ve yansıma bileşeninin (q) çarpımından oluşan parlaklık değerini ifade etmektedir. Işıklandırma, görüntü üzerine düşen kaynak ışık miktarıdır. Günümüzde kullanılan imgeler (görsel imgeler , X-ray imgeler v.b.) ışıklandırma kaynaklarına göre sınıflandırılırlar. İmgelerin oluşmasındaki başlıca enerji kaynakları elektromanyetik enerji spektrumuna göre değerlendirilir. Yansıma ise görüntü üzerindeki nesnelerden yansıyan ışık miktarıdır. $f(x, y)$ eşitlik 3.1. kullanılarak hesaplanır. [108].

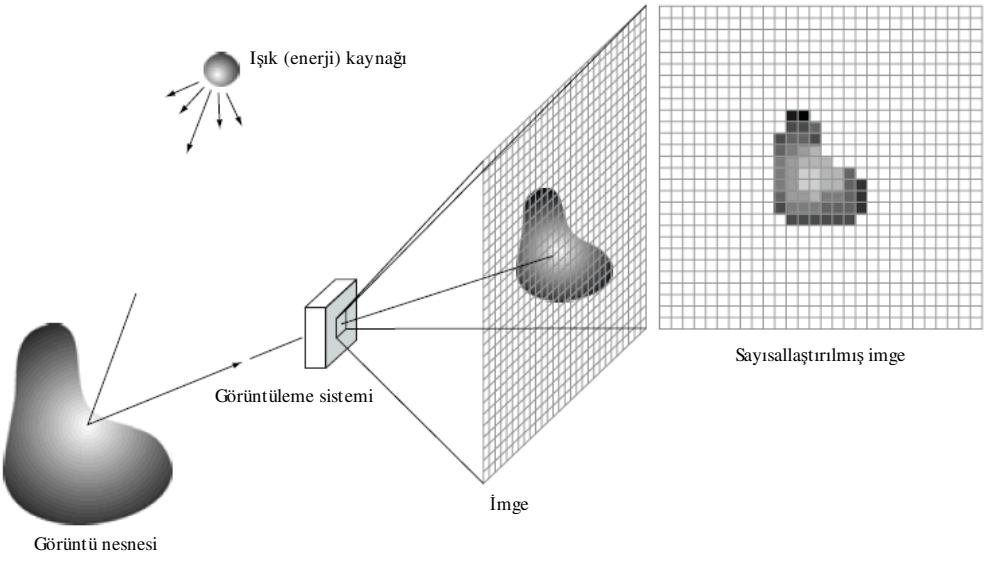
$$f(x, y) = z(x, y)q(x, y) \tag{3.1}$$

Işıklandırma bileşeni $0 < z(x, y) < \infty$ aralığında ve yansıma bileşeni $0 < q(x, y) < 1$ aralığında tanımlıdır.

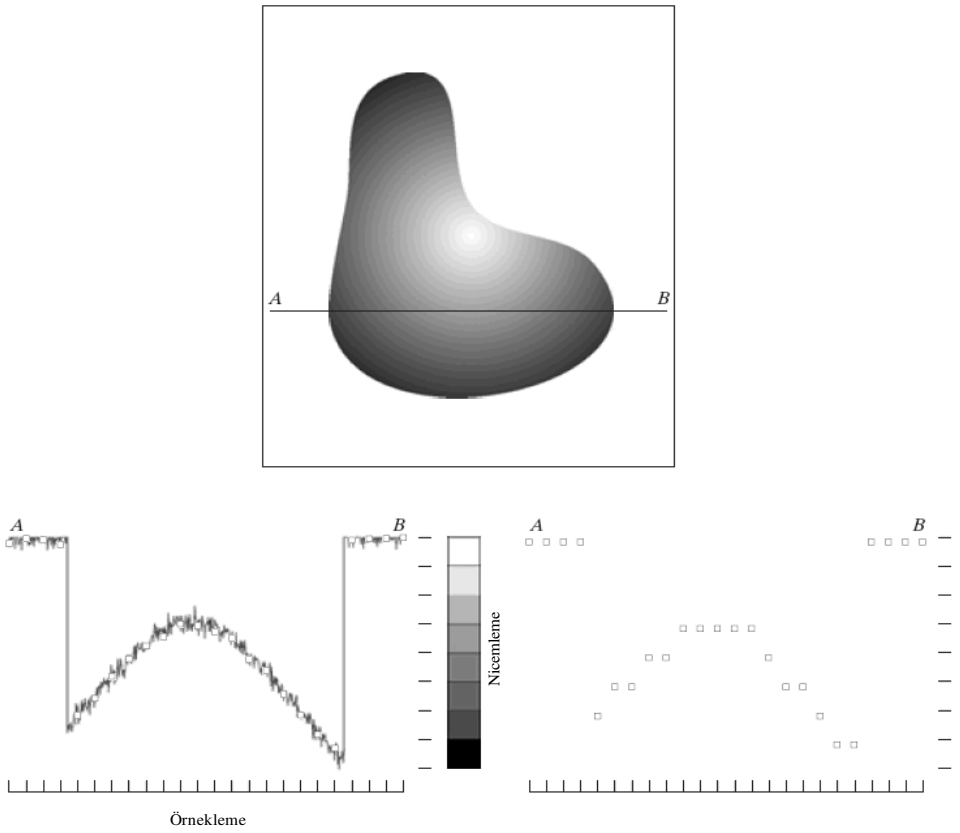
3.1.2. Sayısal İmgenin Oluşturulması

Şekil 3.2.'de bir görüntüleme sistemi tarafından imgenin algılanması şematize edilmektedir [108]. Buradaki görüntüleme sistemi; (x, y) koordinat değerleri ile f parlaklık değerlerini sürekli ve sonsuz sayıda değerler ile ifade edebiliyorsa, imge analog olarak oluşturulur.

İmge içerisindeki (x, y) ve f değerleri sonlu ve ayrık niceliklerden oluşuyorsa imge, sayısal imge olarak adlandırılır. Sayısal bir imge sonlu sayıda elemandan oluşur ve her bir eleman belirli bir konuma ve değere sahiptir. Bu elemanlar piksel olarak adlandırılır. Sayısal görüntüleme sistemlerinde ışığı elektronlara dönüştürmek için CCD *(Charge Coupled Device)* veya CMOS *(Complementary Metal Oxide Semiconductor)* algılayıcı kullanılır. Görüntüleme sistemindeki her bir algılayıcı, imge üzerindeki bir pikseli oluşturmaktadır. Sayısallaştırma, imgenin örneklenmesi ve nicemlenmesi işlemidir. Örnekleme ve nicemleme işlemi şekil 3.3'de gösterilmiştir [108]. Örnekleme, uzaysal çözünürlüğü belirleyen temel faktördür. Uzaysal çözünürlük, bir imgedeki ayırt edilebilen en küçük detaydır. Gri seviye çözünürlük ise gri seviye aralığında ayırt edilebilen en küçük değişim olarak tanımlanır. Uzaysal çözünürlük ve gri-seviye çözünürlük bir imgenin kalitesini belirleyen iki önemli etkendir. Çözünürlük değerlerinin artırılmasıyla daha yüksek kalitede sayısal imgeler elde edilir.



Şekil 3.2. Bir imgenin oluşması

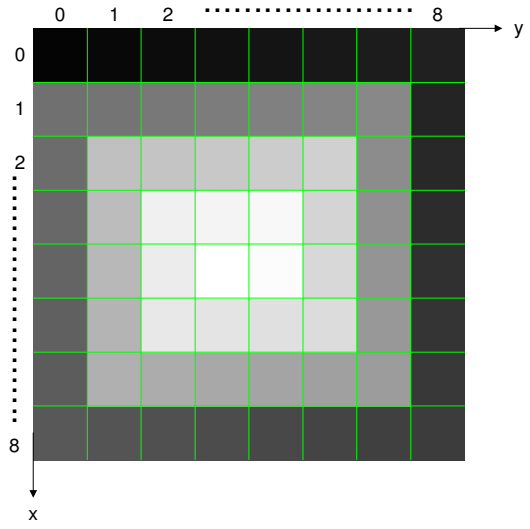


Şekil 3.3. İmgenin sayısallaştırılması

Örnekleme ve nicemleme işleminin sonucunda sayısal imgeyi temsil eden ve tam sayılardan oluşan bir matris elde edilir. $f(x,y)$ imgesi örneklendiğinde, elde edilen sayısal imge M satır ve N sütundan oluşur. (x,y) koordinatlarının değerleri artık tam sayılardan oluşan ayrık değerlerdir. $M * N$ piksellik uzaysal çözünürlükteki gri seviyeli bir imgenin genel matematiksel ifadesi eşitlik 3.2 ile verilmiştir.

$$f(x,y)=\begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & \cdots & f(1,N-1) \\ \vdots & \vdots & & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(M-1,N-1) \end{bmatrix} \tag{3.2}$$

Renkli resimlerde ise, her pikseldeki parlaklık değeri kırmızı, yeşil ve mavi renklerin bileşkesinden oluştuğu için, renkli imge 3 tane $M * N$ boyutlu matris ile ifade edilir. Şekil 3.4'de 8*8 piksellik uzaysal çözünürlüğe ve 8 bitlik gri seviye çözünürlüğe sahip bir sayısal imge ve onun sayısal değerleri gösterilmiştir.



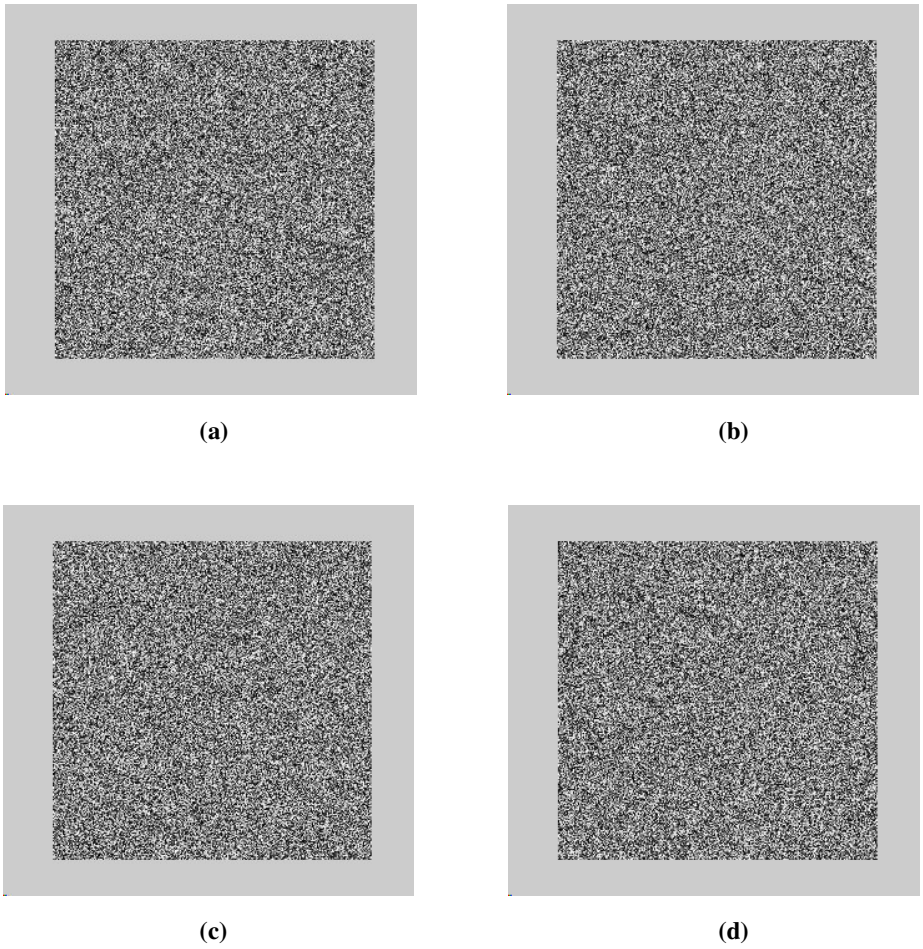
	0	1	2	.....	8
0	4	8	12	16	20	24	28	32
1	112	116	120	124	128	132	136	36
2	108	192	196	200	204	208	140	40
	104	188	240	244	248	212	144	44
	100	184	236	255	252	216	148	48
	96	180	232	228	224	220	152	52
	92	176	172	168	164	160	156	56
8	88	84	80	76	72	68	64	60

Şekil 3.4. 8*8 piksellik bir imge ve onun sayısal değerleri

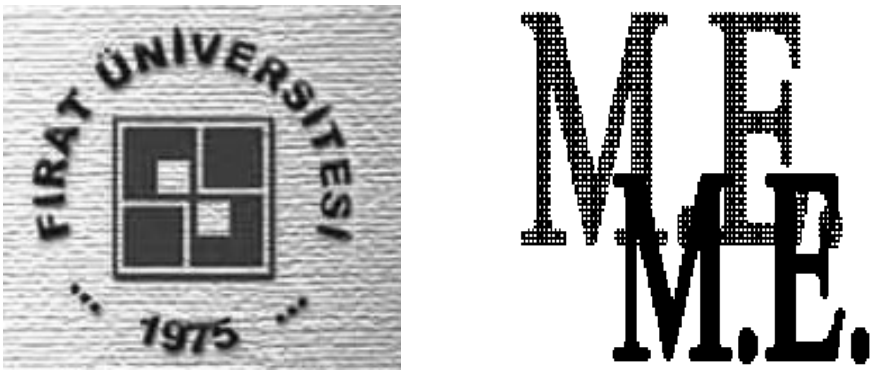
3.2. Filigranın Oluşturulması

Literatürde önerilmiş Sayısal Filigranlama Teknikleri'nin çoğu, filigran olarak sözde rastlantısal üretilmiş gürültü modellerini kullanmaktadır [109-116]. Sözde rastlantısal gürültü modelleri sözde rastlantısal sayı üreteçleri kullanılarak oluşturulur [117, 118]. Bu üreteçler, bir

başlangıç değeri veya çekirdek olarak tanımlanan bir değere göre istenilen istatistiksel özelliklere uygun çıkış değerleri üretirler. Bu çıkış değerleri $\{-1,0,1\}$ gibi tamsayı veya gerçel sayı değerlerinden oluşabilir. Çekirdek değerin değişmesi, üretecin çıkış değerlerinin değişmesine neden olur. Filigran olarak sözde rastlantısal gürültü modellerini kullanan tekniklerde her bir çekirdek farklı bir filigranı tanımlar. Bu yüzden Sayısal Filigranlama Tekniklerinde farklı kullanıcılar, kullanılan algoritmadan bağımsız olarak kendi filigranlarını oluşturabilirler. Filigranların güvenliği ise kullanılan çekirdek değerin gizliliği ile sağlanmaktadır. Şekil 3.5.'de farklı çekirdekler kullanılarak oluşturulmuş bir biçimli dağılıma sahip sözde rastlantısal gürültü modelleri gösterilmiştir.



Şekil 3.5. Farklı çekirdek değerleri ile üretilmiş sözde rastlantısal gürültü modelleri **(a)** çekirdek=3 **(b)** çekirdek=4 **(c)** çekirdek=5 **(d)** çekirdek=6



(a) **(b)**

Şekil 3.6. İmgesel logolar **(a)** Gri ölçekli **(b)** Siyah-beyaz

Bazı Sayısal Filigranlama Teknikleri ise sözde rastlantısal gürültü modelleri yerine, imgesel logolar gibi daha anlamlı bir bilgiyi veya imgenin piksel değerlerinden hesaplanan bir sayısal imzayı filigran olarak kullanırlar [37-40, 42, 43, 52]. İmgesel logolar şekil 3.6.'da gösterildiği gibi gri ölçekli veya ikili bir imge modelinden oluşturulabilir.

3.3. Filigranın Gömülmesi

Özgün bir imgenin piksel değerlerinin filigranı barındıracak şekilde değiştirilmesi işlemi filigranın gömülmesi olarak tanımlanır. Filigran gömücü; I özgün imgesini, W filigranını, K_G gömü anahtarını giriş olarak alır ve I' filigranlı imgesini çıkış olarak verir. Filigranlama işlemi sonucunda özgün imgenin piksel değerlerinin değiştirilmesiyle bir bozulmaya neden olunur. Bu bozulma gömü bozulması olarak bilinir. Görünmez Filigranlama Teknikleri'nde gömü bozulmasının İGS tarafından algılanmaması istenilir [93-95, 103-107].

Sayısal bir filigran özgün imge içerisine eklemeli, çarpımlı ve nicemlemeli olarak gömülebilir. Eklemeli gömmede, filigranın değeri imgenin piksel değeri üzerine eşitlik 3.3.'deki gibi eklenir.

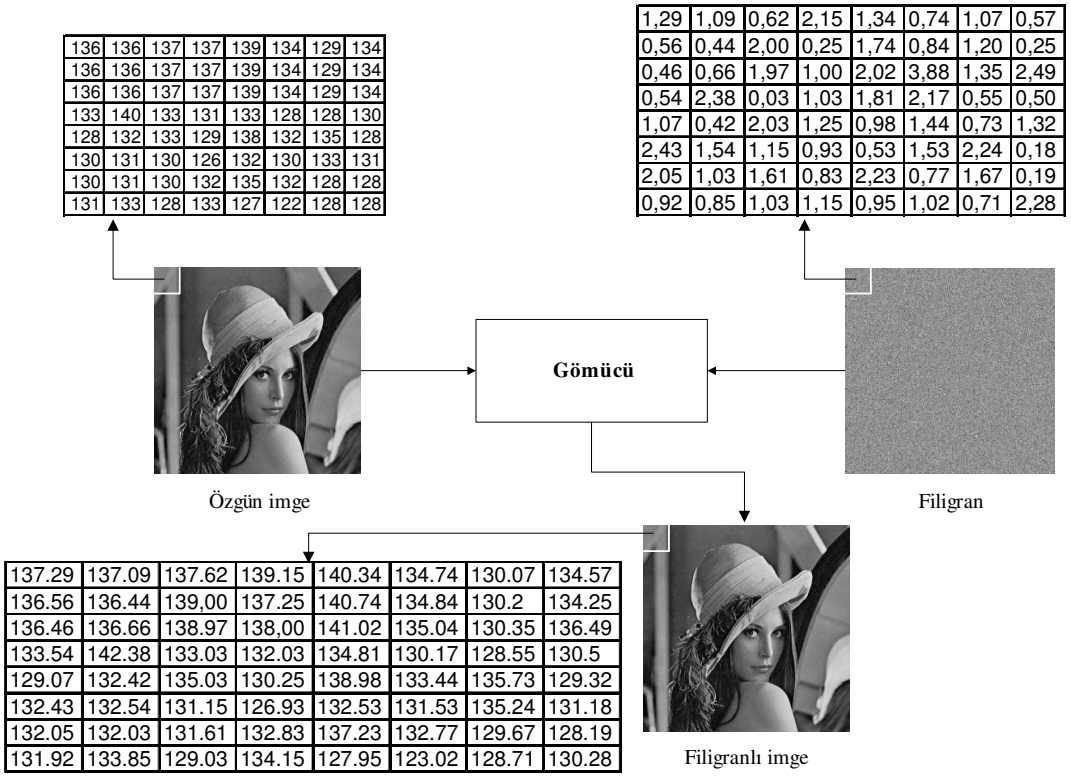
$$I' = I + \alpha W \tag{3.3}$$

Burada α, gömülen filigranın genlik değerini ölçeklendirmek için kullanılan bir parametreyi göstermektedir. Bu parametrenin değerinin artırılması, filigranı gömülen imge içerisinde daha gürbüz bir hale getirir, fakat imgenin algılanabilir kalitesinde bozulmalara neden olur. Bu parametre etkinlik değeri olarak adlandırılabilir. Çarpımlı gömmede ise filigran imgenin piksel

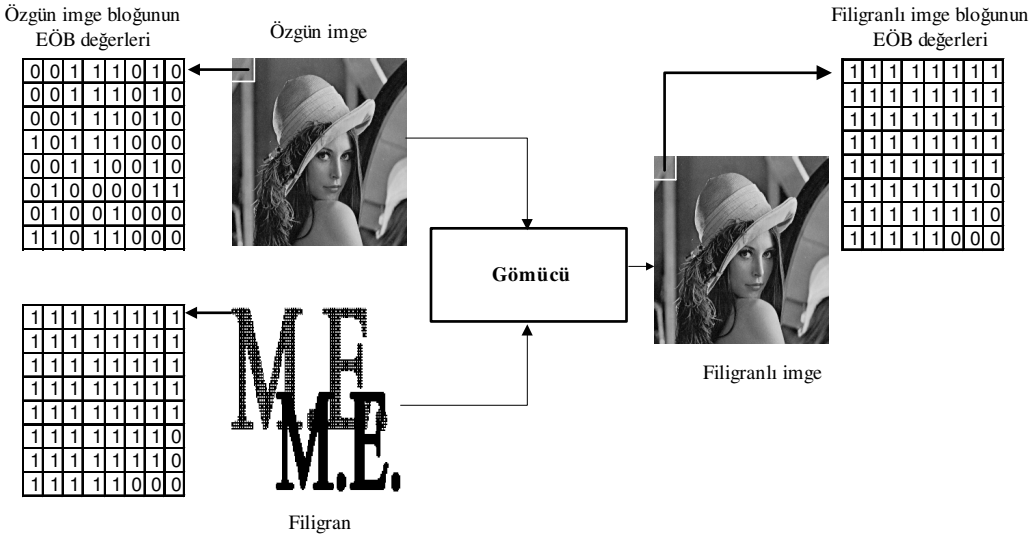
değeri ile çarpıldıktan sonra piksel değerine eklenir [114, 119, 120]. Çarpımlı gömme eşitlik 3.4. ile tanımlanır.

$$I' = (1 + \alpha w) I \tag{3.4}$$

Şekil 3.7.'de sözde rastlantısal gürültü modelinden oluşturulan bir biçimli bir dağılıma sahip bir filigran imge içerisine eklenerek gömülmüştür. Burada, imgenin en üst sol köşesindeki 8*8 pikselinin parlaklık değerlerine bu bölgeye karşılık gelen filigran değerleri eklenerek filigranlı imge bloğunun oluşturulması görülmektedir. Şekil 3.8.'de ise imgenin EÖB düzlemi içerisine ikili bir imgesel logonun gömülmesi gösterilmiştir. Burada filigranlı imgenin en üst sol köşesindeki 8*8 pikselinin EÖB düzleminin değerleri imgesel logonun değerleri ile değiştirilmiştir. Filigranlı imgenin EÖB düzlemi artık ikili imgesel logoyu barındırmaktadır.



Şekil 3.7. Sözde rastlantısal gürültü modelinden oluşan bir filigranın eklenerek gömülmesi

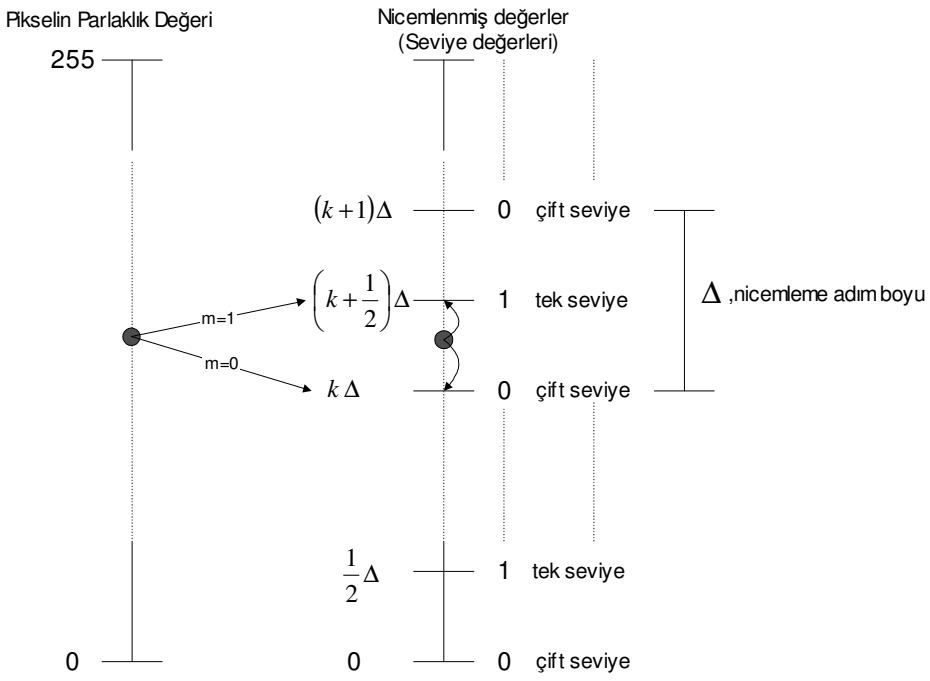


Şekil 3.8. İkili bir imgesel filigranın imgenin EÖB düzlemine gömülmesi

Nicemlemeli gömmede özgün imge değerleri, gömülmek istenen mesajın değerine göre nicemlenmiş değerlere atanır. Nicemleme, sürekli yada büyük bir değerler kümesinin değerlerini sonlu sayıda ve eşit olmayabilen ayrı aralıklara bölme ve her bir aralığa nicemlenmiş değer diye adlandırılan bir değer atama işlemidir. Nicemlemeli gömme işlemindeki temel yapı aşağıda açıklanmıştır: m mesajı 0 ve 1 sembollerinden ($m \in \{0,1\}$) ve l uzunluğundaki bir $m_i, i \in \{1,2,...,l\}$ bit dizisinden; 8 bitlik bir imgenin $I(x,y)$ piksel değerleri $Z_{255} = \{z \in Z \mid 0 \le z \le 255\}$ kümesindeki parlaklık değerlerinden oluşsun. Nicemlemeli indis modülasyonu kullanılarak pikseller $\left\{ z \in Z_{255} \;\middle|\; z = \frac{k}{2}\Delta,\ k \in Z \right\}$ kümesindeki değerlere nicemlenir. İmge içerisindeki piksel değerleri artık Z_{255} kümesindeki değerler ile temsil edilemez. Nicemleme sonucunda oluşan yeni kümede daha az sayıda değer mevcuttur. Bu yeni değerler nicemleme seviyeleri olarak adlandırılır. $k\Delta, k \in Z$ değere sahip bir seviye, 0 mesaj biti ile ilişkilendirilir ve çift seviyeleri tanımlar. 1 mesaj bit değeri ile ilişkilendirilen seviyeler ise çift seviyelerin $\frac{\Delta}{2}$ kadar kaydırılmış seviyelerinden oluşur. Bu seviyeler $\left(k + \frac{1}{2}\right)\Delta, k \in Z$ değerine sahiptirler ve tek seviyeler olarak tanımlanırlar. Burada Δ, çift veya tek seviyeler arasındaki en az mesafedir ve nicemleme adım boyunu belirtmektedir. Filigranlama işleminde, imgenin piksel değerleri mesajın değeri göz önünde tutularak en yakın nicemleme seviyesine yuvarlanır. $I(x,y)$ pikseli içerisine $m = 1$ mesajını gömmek için, $I(x,y)$'nin parlaklık değeri en

yakın tek seviyedeki değere nicemlenir. Benzer şekilde $m=0$ mesajını gömmek için, $I(x,y)$'nin parlaklık değeri en yakın çift seviyedeki değere nicemlenir [121]. Pikselin gerçek değeri ile nicemlenmiş değeri arasındaki fark ise nicemleme hatası olarak tanımlanır. Şekil 3.9.'da nicemlemeli gömme işlemi verilmiştir.

Bir piksel içerisine nicemlemeli indis modülasyonu kullanarak bir mesaj biti gömme işlemi sayısal değerler üzerinde aşağıda açıklanmıştır: Bir piksel için parlaklık değeri $I(x,y)=133$ ve nicemleme adım boyu olarak da $\Delta=20$ değeri seçilsin. Çift seviyeli değerler kümesi $\{0,20,40,...,120,140,...\}$ ve tek seviyeli değerler kümesi $\{10,30,50,...,110,130,...\}$ değerlerinden oluşacaktır. Eğer $m=0$ mesajı piksel içerisine gömülecekse, pikselin değeri en yakın çift seviye değeri olan 140 değerine nicemlenirken, $m=1$ için pikselin değeri en yakın tek seviye değeri olan 130 değerine nicemlenir [121].



Şekil 3.9. Nicemlemeli gömme

3.4. Filigranın Bulunması

Filigran bulucunun amacı, içerik içerisinde bir filigran olup olmadığını belirlemektir. Filigran bulucu, araştırılması istenen içeriği ve K_B bulma anahtarını giriş olarak alır. Bazı buluculara, özgün içerik veya özgün içerikle ilgili bir bilgi de giriş olarak verilmektedir. Filigran

bulucu, içerik içerisinde filigranın var olup olmamasına bağlı olarak iki çıkış üretir. Filigranın bulunmaması durumunda "Filigran yok" bilgisi, filigranın bulunması durumunda ise filigranın temsil ettiği mesaj çıkış olarak verilir. Eğer filigranlı içerik üzerinde filigran gömüldükten sonra herhangi bir değişim yapılmamışsa, gömülen mesaj ile bulunan mesajın aynı olması gerekir.

İçerik içerisine gömülen bir filigran, yalnızca K_B anahtar bilgisi ile bulunabilir. Eğer filigran bulmak için kullanılan anahtar, filigranı gömmek için kullanılan anahtar ile aynı ise $(K_G = K_B)$, bu filigranlama tekniği simetrik veya genel anahtarlı filigranlama tekniği olarak adlandırılır. Filigranı gömmek ve bulmak için farklı anahtarlar kullanan $(K_G \neq K_B)$ teknikler ise asimetrik veya özel anahtarlı teknikler olarak tanımlanırlar. Asimetrik filigranlama tekniklerinde, filigranı gömmek için kullanılan anahtar bilgisi paylaşılmadığı için daha güvenlidir. Bir filigran bulucunun girişinde dört olası değer mevcuttur [70]:

1. I , özgün imge
2. I', filigranlı imge
3. $\hat{I}$, saldırı yapılmış filigranlı imge
4. X, herhangi bir sinyal

Bu giriş değerlerine ve bulma anahtarının doğruluğuna göre bulucudan beklenilen çıkış değerleri tablo 3.1.'de verilmiştir. Filigranlama sistemlerinde, bulucular dan yalnızca doğru anahtar kullanıldığında bir filigran bulması beklenir. Tablo 3.1'e göre filigran bulucuya; özgün imge, herhangi bir sinyal veya yanlış bulma anahtarı ile birlikte bir filigranlı imge giriş olarak verildiğinde, filigran bulucudan bir filigran bulmaması beklenir. Ancak filigranlı imge doğru bir bulma anahtarı ile birlikte giriş olarak verilirse, filigran bulucunun "filigran var" çıkışını üretmesi beklenir. Saldırı yapılması durumunda, bir filigran bulucunun çıkışı filigranlama tekniğinin gürbüz veya kırılgan olmasına bağımlı olarak değişmektedir. Gürbüz Filigranlama Teknikleri'nde saldırı içeriği kullanılamaz bir hale getirecek şekilde bir hasar vermedikçe, filigran bulucunun filigranı bulması gerekir. Kırılgan Filigranlama Teknikleri'nde ise bulucu en küçük bir bozulmada bile filigranı doğrulamamalıdır.

Filigran bulucu giriş sinyalini inceledikten sonra, bulucunun ürettiği çıkış değerinin doğruluğu üç durum içerisinde incelenir [70]. İlk durumda bulucu, giriş değerine göre tablo 2'deki gibi doğru bir çıkış değeri üretir. İkinci durumda bulucu, filigranlanmamış bir imge içerisinde bir filigran bulur. Bu durum yanlış pozitif olarak bilinir. Üçüncü durumda ise bulucu, doğru anahtar ile birlikte filigranlı bir imge içerisinde filigranı bulamaz. Bu durum yanlış negatif olarak tanımlanır.

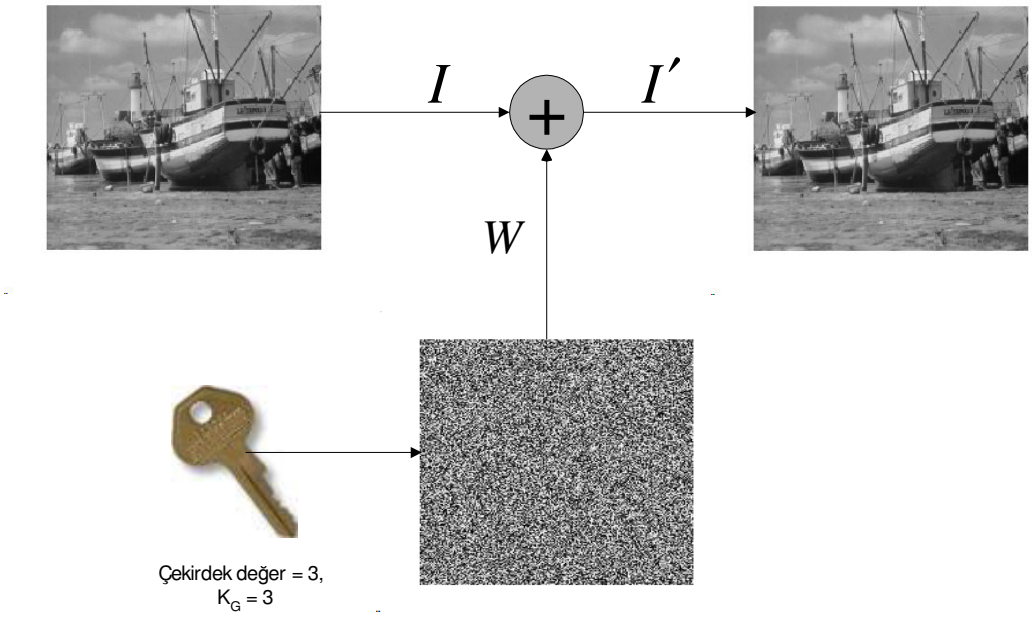
Tablo 3.1. Olası girişlere göre, bulucu çıkışı (* saldırı, imgeyi kullanılamaz bir hale getirmemişse)

Bulucu Girişi	**Bulma Anahtarı**	**Bulucu Çıkışı**
I , özgün imge	Doğru veya Yanlış K_B	Filigran yok
X , herhangi bir işaret	Doğru veya Yanlış K_B	Filigran yok
I', filigranlı imge	Doğru K_B	Filigran var
I', filigranlı imge	Yanlış K_B	Filigran yok
$\hat{I}$, saldırıya uğramış filigranlı imge	Doğru K_B	Filigran var*
$\hat{I}$, saldırıya uğramış filigranlı imge	Yanlış K_B	Filigran yok

Basit bir filigran bulucunun çıkışı, incelenen içerik içerisinde bir filigranın bulunup bulunmamasına göre tablo 3.1.'deki gibi basitçe "filigran var" veya "filigran yok" şeklindedir. Ancak daha karmaşık filigran buluculardа çıkış, imgesel bir logo, sayısal bir imza veya içerikle ilgili bir metin olabilir. Basit filigran buluculu sistemlerde, içerik içerisinde bir filigranın var olup olmadığını belirlemek için içerik ile filigran arasındaki korelasyon değerine bakılır. Eğer korelasyon değeri eşik değerinden büyük ise içerik içerisinde filigranın mevcut olduğu sonucuna varılır. I' imgesi ile W filigranı arasındaki lineer korelasyon değeri eşitlik 3.5. ile hesaplanır.

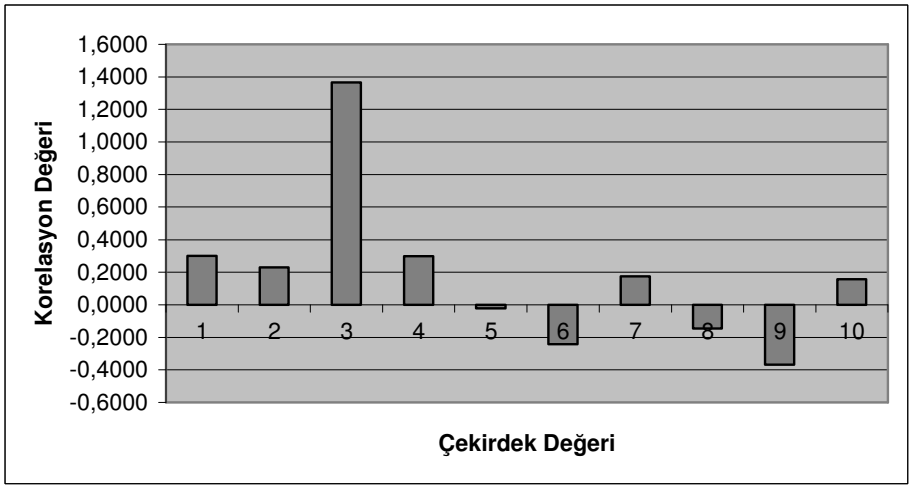
$$Z_{LK}(I',W) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} I'[x,y]W[x,y] \tag{3.5}$$

Farklı çekirdek değeri kullanılarak üretilen gürültü modelleri arasındaki korelasyon değeri çok düşüktür. Bu yüzden bulma işleminde, sadece doğru çekirdek değerinin kullanılmasıyla eşik değerini aşan bir korelasyon değeri elde edilebilir. Çekirdek değer, filigranlama sistemlerinde K_G anahtarı olarak kullanılabilir. Şekil 3.10.'da verilen I özgün imgesi içerisine, çekirdek değeri 3 olan ($K_G = 3$) bir biçimli dağılıma sahip sözde rastlantısal gürültü modeli filigran olarak eşitlik 3.3. kullanılarak gömülmüştür. Etkinlik değeri olarak $\alpha = 1$ seçilmiştir.



Şekil 3.10. Filigranın gömülmesi

Şekil 3.11.'de ise, çekirdek değeri 0-10 arasında değişen sözde rastlantısal gürültü modelleri ile filigranlı imge arasındaki lineer korelasyon değerleri verilmiştir. Bu şekilde görüldüğü gibi, doğru çekirdek değeri ile üretilmiş filigran ile filigranlı imge arasında oldukça yüksek bir korelasyon değeri varken, yanlış çekirdek değeri ile oluşturulan filigranlar ile filigranlı imge arasında çok düşük bir korelasyon değeri vardır.



Şekil 3.11. Çekirdek değeri 0-10 arasında değişen modeller için elde edilen korelasyon değerleri

Bulma işlemi sırasında, incelenen imge içerisinde bir filigranın var olup olmadığına karar vermek için bir eşik değeri (t) belirlenir. Eşik değeri, maksimum yanlış pozitif olasılığı üzerinde bir kısıtlamaya göre seçilebilir [122]. Hesaplanan korelasyon değeri eşik değerini aşarsa, bulucu I' imgesi içerisinde W filigranının var olduğunu bildirirken, eşik değerinin aşılmaması durumunda bulucu filigranın var olmadığını bildirir.

Lineer korelasyona dayanan buluculu sistemlerde korelasyon değeri pikselin parlaklık değerine son derece bağımlıdır. Dolayısıyla bu sistemler, imgenin parlaklığının değiştirilmesi gibi basit işlemlere karşı oldukça duyarlıdırlar. Bu problemi gidermek için buluculardа, normalize edilmiş korelasyon kullanılır [5]. Normalize edilmiş korelasyonda, filigranlı imge ile filigran değerleri arasında iç çarpım hesaplanmadan önce, filigranlı imge ve filigran değerleri birim genliğe normalize edilir. Bu işlemler eşitlik 3.6., 3.7. ve 3.8. ile verilmiştir.

$$\dot{I} = \frac{I'}{|I'|} \tag{3.6}$$

$$\dot{W} = \frac{W}{|W|} \tag{3.7}$$

$$Z_{NK}(I', W) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \dot{I}(x, y) \dot{W}(x, y) \tag{3.8}$$

Burada $\dot{I}$ ve $\dot{W}$ birim genliği normalize edilmiş değerlerdir. İmgenin tüm piksellerine sabit bir değer ekleme gibi imgenin DC teriminde bir değişime neden olan saldırılara karşı korelasyon katsayısını barındıran buluculаrın kullanılması önerilmektedir [5]. Korelasyon katsayısı, filigranlı imge ve filigran değerlerinin ortalamalarının kendilerinden çıkartılmasıyla elde edilen değerler arasındaki normalize edilmiş korelasyonun hesaplanmasıyla bulunur. Korelasyon katsayısı eşitlik 3.9., 3.10. ve 3.11. kullanılarak hesaplanır.

$$\ddot{I} = I' - \bar{I} \tag{3.9}$$

$$\ddot{W} = W - \overline{W} \tag{3.10}$$

$$Z_{KK}(I', W) = Z_{NK}(\ddot{I}, \ddot{W}) \tag{3.11}$$

Burada $\bar{I}$ ve $\overline{W}$ ortalama değerleri göstermektedir.

3.5. Sayısal Filigranlama Teknikleri

Sayısal bir filigran, imgenin uzaysal ve frekans alanı içerisine gömülebilir. İmgenin uzaysal alanı içerisine filigran, doğrudan doğruya imgenin parlaklık değerleri değiştirilerek gömülür. Filigranın imgenin frekans alanı içerisine gömülebilmesi için ise, ilk olarak imgenin uzaysal alandan frekans alanına dönüştürülmesi gerekir. Bu dönüştürme işlemi, Ayrık Kosinüs Dönüşümü (AKD), Ayrık Fourier Dönüşümü (AFD) gibi dönüşüm fonksiyonları vasıtasıyla yapılır. Sayısal Filigranlama Teknikleri'nde sıkça kullanılan dönüşüm fonksiyonları aşağıda verilmiştir:

- Ayrık Fourier Dönüşümü: Bu dönüşüm işaret işleme, imge işleme alanında oldukça kullanılan ve iyi bilinen bir dönüşümdür. Ana sinyalin ve filigranın frekans alanındaki değerlerini kontrol etmek için kullanılır [123-125]. $M * N$ boyutlu $I(x, y)$ imgesi için Ayrık Fourier Dönüşümü eşitlik 3.12. ile ve Ters Ayrık Fourier Dönüşümü eşitlik 3.13. ile verilmiştir.

$$F(u,v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I(x,y)\, e^{-j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)} \tag{3.12}$$

$$I(x,y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v)\, e^{j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)} \tag{3.13}$$

- Ayrık Kosinüs Dönüşümü: Ayrık Kosinüs Dönüşümü JPEG ve MPEG kayıplı sıkıştırma tekniklerinde ve imgeler arasındaki görsel farklılıklardaki algılanabilirliği ölçme çalışmalarında kullanılmaktadır [92, 126]. Sayısal Filigranlama Teknikleri'nde bu dönüşümün kullanılmasıyla JPEG ve MPEG gibi sıkıştırmalara karşı daha gürbüz filigranlar oluşturulurken, filigranlı verinin aslına benzerliğinin ölçülmesi de daha kolay bir şekilde yapılmaktadır. Literatürde Ayrık Kosinüs Dönüşümüne dayanan bir çok Sayısal Filigranlama Tekniği önerilmiştir [93, 127-130]. İmgeler için Ayrık Kosinüs Dönüşümü ve Ters Ayrık Kosinüs Dönüşümü sırasıyla eşitlik 3.14. ve 3.15. ile verilmiştir.

$$C(u,v) = \frac{2}{\sqrt{MN}} \varphi(u)\, \varphi(v) \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I(x,y) \cos\left[\frac{(2x+1)u\pi}{2M}\right] \cos\left[\frac{(2y+1)v\pi}{2N}\right] \tag{3.14}$$

$$I(x,y)=\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}\frac{2}{\sqrt{MN}}\varphi(u)\varphi(v)C(u,v)\cos\left[\frac{(2x+1)u\pi}{2M}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right] \quad (3.15)$$

Burada,

$$\varphi(u),\varphi(v)=\begin{cases}\frac{1}{\sqrt{2}} & u=0;v=0\\ 1 & u=1,2,...,M-1;v=1,2,...,N-1\end{cases}$$

- Fourier-Mellin Dönüşümü: Sayısal Filigranlama Teknikleri'nin çoğu öteleme, ölçeklendirme ve döndürme gibi geometrik saldırılara karşı gürbüz değildir. Bu işlemlere karşı gürbüzlük sağlamak için Fourier-Mellin dönüşümüne dayanan filigranlama teknikleri önerilmiştir [131, 132]. Fourier-Mellin Dönüşümü, Fourier dönüşümünün öteleme değişmezliği özelliğine dayanmaktadır [121]. Bu dönüşüm eşitlik 3.16. ile verilmiştir.

$$I(x+a,y+b)\Leftrightarrow F(u,v)e^{[-j2\pi(ua+vb)]} \quad (3.16)$$

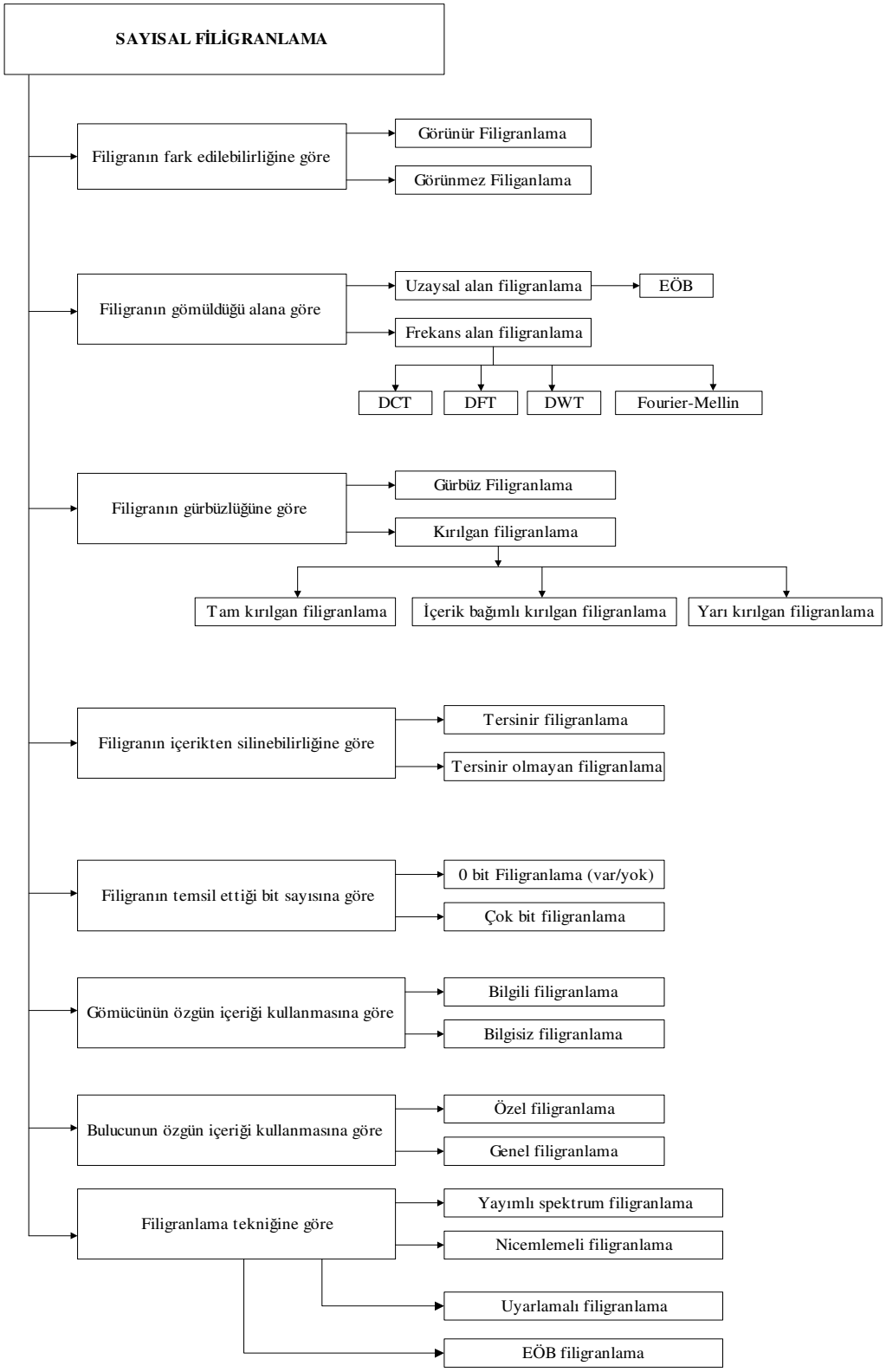
Bu özelliğin kullanımı ile, filigranın uzaysal ötelemelere karşı gürbüzlüğü sağlanır. Döndürme ve ölçeklendirmeye karşı filigranın sağlamlığı ise log-polar eşleme kullanılarak sağlanır [121]. Bu işlem eşitlik 3.17. ile verilmiştir.

$$(x,y)\rightarrow\begin{cases}x=e^{(\varphi)\cos(\theta)}\\ y=e^{(\varphi)\sin(\theta)}\end{cases},\varphi\in R\ ve\ \theta\in[0,2\pi] \quad (3.17)$$

Burada R, ölçeklendirme faktörlerinin olası kümesini göstermektedir. Döndürme işlemi logaritmik koordinat sistemi üzerinde bir öteleme ile sonuçlanırken, ölçeklendirme işlemi polar koordinat sistemindeki öteleme ile sonuçlanır.

- Ayrık Dalgacık Dönüşümü: Ayrık Dalgacık Dönüşümü bir sinyalin hem frekans bileşeni hem de zamansal veya uzaysal bileşeni hakkında bilgi verir [133-135]. Ayrık Dalgacık Dönüşümü, Sayısal Filigranlama Teknikleri'nde sıkça kullanılan diğer bir dönüşümdür [105, 135-138]. Bu dönüşümde sinyal, yüksek ve düşük frekanslı iki bölüme ayrıştırılır. Sinyalin kenar bileşenleri büyük miktarda yüksek frekanslı bölüm içerisinde kalır. Düşük frekanslı bölüm tekrar yüksek ve düşük frekanslı bölümlere ayrıştırılır. Bu işlem ihtiyaç duyulan sayıda tekrar edilir.

Sayısal Filigranlama Sistemleri'nin genel bir sınıflandırılması şekil 3.12.'de verilmiştir.



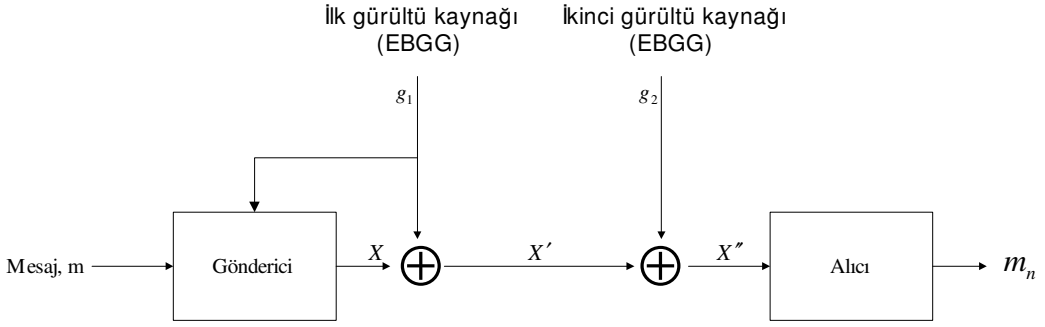
Şekil 3.12. Sayısal Filigranlama Sistemleri'nin sınıflandırılması

3.5.1. Nicemlemeli Filigranlama Teknikleri

Nicemlemeli Filigranlama Teknikleri'nde, filigran özgün imge içerisine imgenin piksel değerleri nicemlenerek gömülür [26]. Nicemleme, sürekli yada büyük bir değerler kümesinin değerlerini sonlu sayıda ve eşit olmayabilen ayrı aralıklara bölme ve her bir aralığa nicemlenmiş değer diye adlandırılan bir değer atama işlemidir. Nicemlemeli gömmeyi kullanan filigranlama tekniklerindeki genel yapı Costa'nın [139] çalışmasına dayanmaktadır [5]. Bu çalışmada, normal olarak dağılmış bağımsız kir lekeleri ile kaplanmış bir kağıt üzerine sınırlı bir mürekkep ile bir mesajın yazılması istenmektedir. Daha sonra kirli kağıt üzerine yazılmış mesaj bir başkasına gönderilirken yol boyunca normal olarak dağılmış lekelerin eklendiği farz edilmiştir. Burada alıcının mürekkep ile kir arasındaki farkı ayırt edememesi durumunda, güvenilir olarak ne kadar bilginin gönderilebileceği sorulmuştur.

Bu problem şekil 3.13.'de verilen haberleşme kanalı için incelenmiştir [5]. Kanal iki adet eklemeli beyaz gaussian gürültü kaynağına (g_1, g_2) sahiptir. Gönderici X sinyalini göndermek için seçmeden önce, ilk gürültü kaynağı tarafından gönderilmek istenen sinyale eklenecek gürültü (g_1) göndericiye bildirilmektedir. Bu gürültü, kağıt üzerine mesajı yazmadan incelenilen ilk lekelere benzetilmektedir. Daha sonra sinyal eşitlik 3.18. ile verilen G_K güç kısıtlaması ile sınırlandırılarak gönderilir.

$$\frac{1}{L}\sum_{i}^{L} X[i]^2 \le G_K \ , \ i = 0,1,...,L \tag{3.18}$$



Şekil 3.13. Costa tarafından çalışılan kanal

Bu sınırlama, mesajı yazmak için kullanılan sınırlı mürekkep ile eşleştirilmektedir. İkinci gürültü kaynağı (g_2) ise mesaj bir başkasına gönderilirken eklenilen ikinci lekelere benzetilmektedir. Costa oldukça şaşırtıcı bir sonuç olarak kanalın kapasitesi üzerinde ilk gürültü kaynağının bir etkisi olmadığını göstermiştir [139].

Costa'nın kirli kağıt üzerine yazma problemi, filigranlama içinde uygundur. Şöyle ki kirli kağıt sayısal imge, kağıt üzerine yazılan mesaj filigran, güç sınırlaması özgün imgenin değiştirilmesi ile neden olunan algısal bozulma miktarı ve ikinci kir lekesi saldırılar ile eşleştirilebilir. Eğer bir filigranlama sistemi kirli kağıt kanalı gibi davranırsa, imge içerisine gömülen maksimum veri yükünün özgün imgeye bağımlı olmaması gerekir. Ancak sayısal bir filigranlama sisteminde iki gürültü kaynağı olarak tanımlanan özgün imge ve saldırılar nadiren gaussian gürültüsüne benzerdir. Özgün imgenin piksellerinin dağılımı gaussian gürültüsünden çok daha karmaşıktır. Yine ikinci gürültü kaynağına benzetilen saldırılar hem gaussian gürültüsüne benzer değildir hem de filigranlı imgeye bağımlıdır. Ayrıca filigranın neden olduğu algısal bozulmanın belirlenmesinde ortalama karesel hata gibi metriklerin kullanılmasından dolayı güç sınırlaması gibi kesin bir ölçüm değeri elde edilemez. Bu yüzden imge içerisine gömülen maksimum veri yükünün tamamıyla özgün imgeden bağımsız olduğu sonucuna varılamaz [5]. Ancak, Chen ve Wornell [21-26] Costa'nın önerisine dayanarak filigranlama için Nicemlemeli İndis Modülasyon Tekniği'ni önermişlerdir. Costa'nın önerisini filigranlamada pratik olarak kullanabilmek için Eggers ve diğ. [27, 28] tarafından Scalar Costa Scheme olarak adlandırılan diğer bir teknik önerilmiştir.

3.5.2. Yayımlı Spektrum Filigranlama Teknikleri

Yayımlı spektrumda *(spread spectrum)*, dar bant bir giriş sinyali geniş bant bir sinyal üretmek için kodlanır veya modüle edilir. Daha sonra üretilen geniş bantlı sinyal, gürültülü veya kayıplı bir kanalın içerisinden iletilir. Giriş sinyalinden geniş bantlı bir sinyal üretme işlemi yayma olarak tanımlanır. Geniş bantlı sinyalin iletilmesi daha fazla bant genişliği gerektirir. Ancak dar bant sinyalin yayılması kanalın neden olduğu gürültü etkilerini azaltır [29]. Alıcı, geniş bant sinyali aldıktan sonra çözme işlemini uygulayarak giriş sinyalini elde eder. Yayımlı spektrum, gürbüz filigranlama tekniklerinde kullanılmaktadır [30-33]. Yayımlı spektrumu kullanan filigranlama tekniklerinde, veri yükü dar bant sinyal olarak ve veri yükünü kodlamak için kullanılan filigran ise geniş bant sinyal olarak kullanılır. Yayımlı spektrumun iki önemli karakteristiği filigranlama açısından önemlidir. Birincisi herhangi bir frekans içerisine eklenilen sinyalin enerjisi çok küçüktür. Bu küçük enerji çok düşük sinyal-gürültü oranına neden olarak algısal bozulmaların oluşmasını engeller. Ayrıca, bulucu içerisinde çok sayıda frekans içerisine

gömülmüş olan enerji bir araya getirildiğinde yüksek sinyal-gürültü oranı tekrar elde edilebilir ki bu filigranın tekrar bulunabilmesi için gereklidir. İkincisi, filigranın çok sayıda frekans üzerine yayılması çoğu sinyal bozulmalarına karşı filigranın gürbüzlüğünü artırır [5, 114].

3.5.3. Uyarlamalı Filigranlama Teknikleri

Literatürde İGS'nin karakteristiği göz önünde tutularak önerilen Sayısal Filigranlama Teknikleri, Uyarlamalı Filigranlama Teknikleri *(Adaptive Watermarking)* olarak adlandırılırlar [93-95, 103-107, 127, 140, 141]. Bu teknikler hem algısal olarak özgün imgeye daha yakın filigranlı imgeler oluşturmada hem de filigranın saldırılara karşı daha gürbüz olmasında oldukça etkilidirler. Algısal bir model, imgenin genel algısal kalitesini etkilemeden imgenin belirli bir bölgesi içerisine gömülen bir filigranın neden olduğu bozulma miktarını tahmin eder [70]. Uyarlamalı Filigranlama'da, belirli bir bölge içerisine gömülen filigran bölgenin algısal kalitesinde bir değişime neden olmuyorsa, filigranın genlik değeri artırılarak daha gürbüz olması sağlanır. Ancak bölge içerisine gömülen filigran bölgenin algılanabilir kalitesinde bir değişime neden oluyorsa, gömülen bölgede filigranın genlik değeri azaltılarak daha az bir bozulmaya neden olması sağlanır [107]. Algısal bir model kullanılarak gömülen filigranın genliğinin ayarlanması algısal şekillendirme *(perceptual shaping)* olarak da tanımlanır [5, 70]. Algısal şekillendirmenin bir dezavantajı karmaşıklığı artırarak filigranın gömü maliyetini artırmasıdır.

Bilgili Filigranlama olarak adlandırılan teknikler ise, gömü bozulmasının İGS tarafından daha az algılanabilirliğini sağlamak için filigranı özgün imgeye uyarlayarak üretirler [5, 19, 20, 64]. Özgün imge incelenerek filigranın üretilmesi bilgili gömme olarak da adlandırılır [19].

3.5.4. EÖB Düzlemini Kullanan Filigranlama Teknikleri

Algısal bir bozulmaya neden olmadan filigranı özgün imge içerisine gömmek için imgenin EÖB düzleminin kullanılması önerilir [44, 115]. Bu öneri iki önemli avantaj sağlamaktadır. Birincisi, imgenin her bir pikseli içerisine bir bitlik bilginin gömülmesi ile oldukça yüksek bir veri yükü elde edilmesidir. İkincisi ise filigranı gömmek için imgenin sadece EÖB düzleminin kullanılmasıyla algılanamayan bir gömü bozulmasına neden olunmasıdır. Ancak filigranlı imgenin EÖB düzleminin değiştirilmesi veya filigranlı imgeye uygulanan basit imge işlemleri, imgenin EÖB düzlemini değiştirerek filigranın bozulmasına neden olur. Bu

yüzden EÖB içerisine filigranın gömülmesi genellikle Kırılgan Filigranlama Teknikleri'nde tercih edilir [34-54].

İmgenin uzaysal veya frekans alanı içerisine filigranın gömülmesi sonucunda, filigranın imge üzerindeki fark edilebilirliğine göre Sayısal Filigranlama Teknikleri görünür ve görünmez olarak iki sınıfa ayrılır.

3.5.5. Görünür Filigranlama Teknikleri

Görünür Filigranlama Teknikleri web üzerinde eğitim veren sitelerde veya sayısal kütüphaneler gibi ortamlarda yayımlanan sayısal imgeler gibi verilerin illegal olarak kopyalanması, çoğaltılmasını veya bu imgeler üzerindeki bireysel hakların ihlalini önlemek için kullanılırlar [142-150]. İmge üzerindeki görünür bir filigran genellikle imge ile aynı boyutlu ve telif hakkı sahibinin açıkça kimliğini gösteren yarı saydam bir görüntüdür. Görünür bir filigran şekil 3.14.'de verilmiştir. Sayısal görünür bir filigrandan aşağıdaki özelliklere sahip olması beklenilir:

- Filigran gömüldükten sonra özgün imgenin tüm detayları algılanabilir olmalıdır.
- Gömülen filigran imge üzerinde kolaylıkla fark edilecek şekilde fakat silik bir tarzda görünmelidir.
- Görünür filigranın imge üzerinden kesilerek silinmesini önlemek için filigran özgün imgenin büyük bir bölümü üzerinde bulunmalıdır.
- Gömülen filigranların imge üzerinden silinmesi çok yoğun bir çaba olmadıkça mümkün olmamalıdır veya çok zor olmalıdır.
- Filigran imge içerisine kolaylıkla gömülebilmelidir.



Şekil 3.14. Görünür Filigranlama

3.5.6. Gürbüz, Kırılgan ve Yarı Kırılgan Filigranlama Teknikleri

Gürbüz Filigranlama Teknikleri içerik izleme, telif hakkı koruma, kopya koruma ve yayın izleme uygulamalarında kullanılırlar. Bu uygulamalarda gürbüz filigran, ana verinin ayrılmaz bir parçası olarak tasarlanır. Filigranlı veri saldırıya uğradığında, gürbüz filigranlardan filigranlı veri kullanılamaz hale gelinceye kadar filigranlı veri içerisindeki varlığını sürdürmesi beklenir [5].

Kırılgan Filigranlama Teknikleri doğrulama uygulamalarında kullanılırlar [34-54]. Gürbüz filigranların aksine, kırılgan bir filigran filigranlı içerik üzerindeki en küçük değişiklikte yok olacak şekilde tasarlanır. Doğrulama uygulamalarında, kırılgan filigranların bu özelliğinden faydalanılır. Bu uygulamalarda, içeriğin herhangi bir değişime maruz kalmadığından emin olmak istenir.

Yarı Kırılgan Filigranlama Teknikleri'ndeki filigranlar hem gürbüz hem de kırılgan bir yapıya sahiptirler [6-13, 70]. Yarı kırılgan filigranlar gürbüz filigranlar gibi filigranlı içerik üzerinde yapılan değişimlere karşı bir dereceye kadar gürbüzlük sergiler. Buradaki gürbüzlük derecesi, uygulamada filigranlı içeriğin kullanımını etkilemeyecek değişiklikler ile sınırlıdır. Yarı kırılgan filigranlar, kırılgan filigranlar gibi içerikte yapılan önemli değişimlere karşı kırılgan bir özellik sergilerler. Yarı kırılgan filigranlar için değişim iki türlüdür. Birinci tür değişim "içeriği koruyan" değişimlerdir ve yarı kırılgan filigranlar bu değişime karşı gürbüzdürler. Örneğin, filigranlı imgenin görünür kalitesini etkilemeyecek bir JPEG sıkıştırma oranında sıkıştırılmasına karşı yarı kırılgan filigranlar gürbüzlüklerini korurlar. İkinci tür değişim "içeriği bozan" değişimlerdir ve yarı kırılgan filigranlar bu değişimlere karşı kırılgandırlar. Örneğin yarı kırılgan filigranlar, imge içerisine yeni bir obje eklenmesi, bir objenin değiştirilmesi veya silinmesi gibi bozulmalara karşı kırılgan bir yapıya sahiptirler.

3.6. Saldırılar

Sayısal Filigranlama Teknikleri'ne karşı bir çok saldırı türü önerilmiştir [5, 39, 42, 68, 71, 151-154]. Bu saldırıların amacı, ana verinin kalitesini etkilemeden, ana veri içerisindeki filigranı silmek, değiştirmek veya ana veri içerisine yeni bir filigran gömmektir. Saldırılar çok genel olarak kasıtlı veya kasıtsız olarak sınıflandırılabilir.

Kasıtlı bir saldırı, doğrudan doğruya filigranın gömülme amacını ortadan kaldırmak için yapılır. Kasıtlı saldırılarda, saldırganın filigranlama sistemi hakkında sahip olduğu bilgi oldukça önemlidir. En basit durumda, saldırgan algoritmayla ilgili bir bilgiye ve bir filigran bulucuya sahip değildir. Bu durumdaki bir saldırgan, çoğu filigranlama algoritmasının sıkıntı çektiği zayıflıkları göz önünde tutarak bir saldırı gerçekleştirmeye çalışır. Örneğin saldırgan filigranlı

bir içerikten filigranı silmek için gürültü azaltıcı filtreler, yüksek oranda sıkıştırma ve küçük geometrik bozulmalar uygulayabilir [5]. Bazı durumlarda saldırgan algoritmayı bilmemesine rağmen, birden fazla filigranlı içeriğe sahip olabilir. Bu durumda saldırgan danışıklı *(collusion)* saldırılar olarak adlandırılan saldırıyı gerçekleştirebilir. Diğer bir durum da ise saldırganın, filigranlama algoritmasıyla ilgili tüm bilgiye sahip olduğu düşünülür. Bu durumda saldırgan algoritmayı inceler ve bulma algoritmasının zayıflıklarını tespit ederek bir saldırı gerçekleştirebilir. Örneğin filigran bulucunun filigranı bulmasını engelleyecek bir maske geliştirebilir. En son varsayım ise saldırganın bir bulucuya sahip olduğu durumdur. Bu durumda saldırgan, bulma algoritması hakkında herhangi bir bilgiye sahip olmamasına rağmen, bulucuya sahip olmasından dolayı önemli bir avantaja sahiptir. Saldırgan, filigranlı imgeler üzerinde küçük değişiklikler yaparak bulucunun bir filigran bulup bulmadığını gözlemler ve bulma algoritmasının çalışması ile ilgili önemli bilgiler elde eder.

Kasıtsız bir saldırı ise genellikle filigranlı içeriğin yasal kullanıcısı tarafından yapılır. Örneğin daha fazla disk alanı sağlamak için, bir Bitmap imgesinin sıkıştırılarak JPEG formatında saklanması yasal kullanıcılar tarafından sıkça yapılan bir uygulamadır.

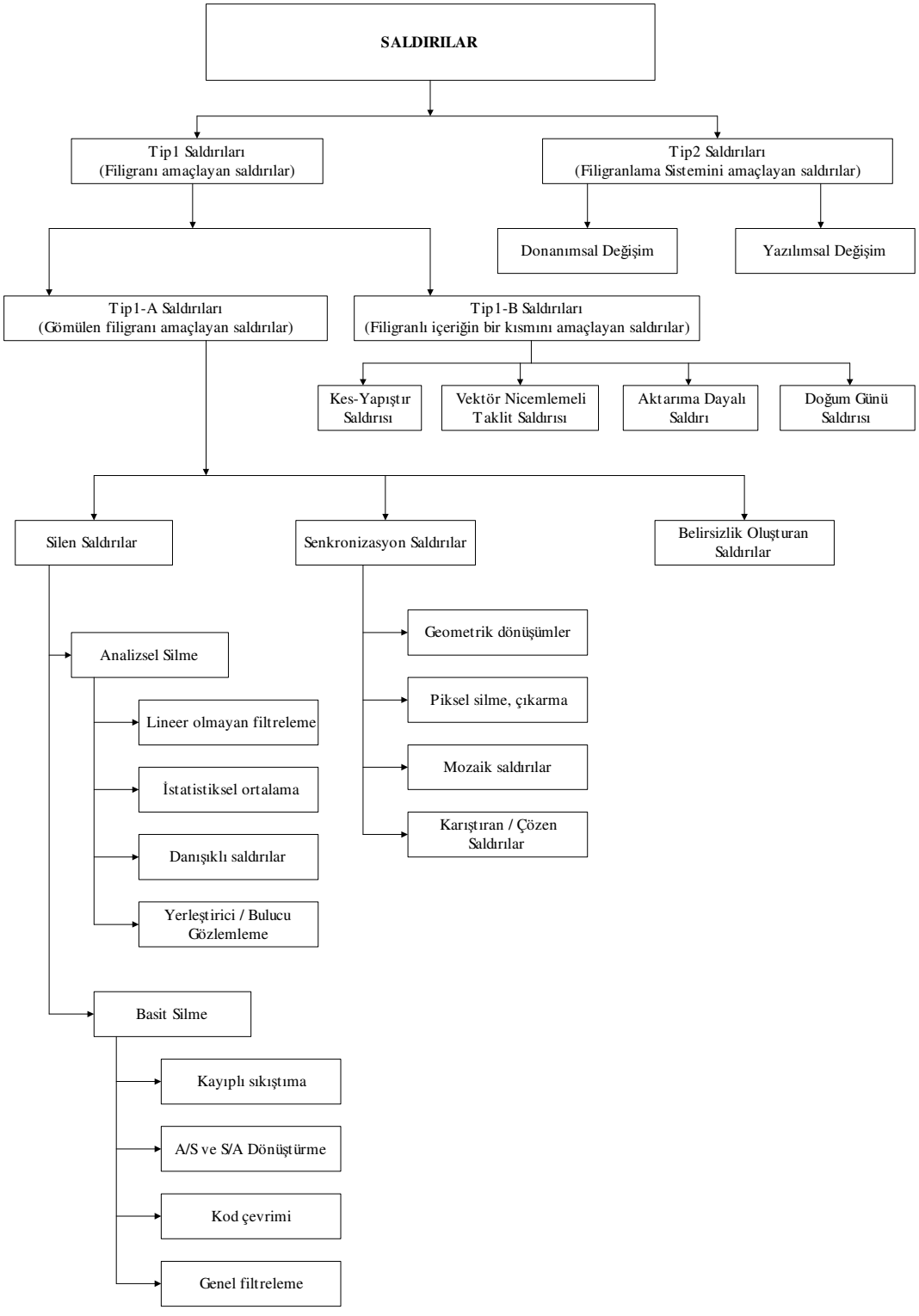
Literatürde saldırıları tanımlamak ve sınıflandırmak için farklı öneriler yapılmıştır [68, 71, 151-154]. Bu tez çalışmasında Setyawan [154] tarafından önerilen ve şekil 3.15.'de verilen sınıflandırma kullanılmıştır. Bu öneride saldırılar Tip1 ve Tip2 olarak sınıflandırılmıştır. Tip1 saldırıları, doğrudan doğruya filigranlı içerikteki filigran verisine odaklanır. Tip2 saldırıları ise filigranlama sisteminin kendisi ile ilgilenir. Tip2 saldırılarında, işletim sistemleri, programlama dilleri veya donanımla ilgili bilgiye ihtiyaç duyulur. Tip1 saldırıları tekrar iki kategoriye bölünmüştür. İlk kategorideki saldırılar Tip1-A saldırıları olarak adlandırılır ve amacı filigran bulucunun filigranı bulamamasını sağlamaktır. İkinci kategorideki saldırılar, gömülen filigranı bozmadan filigranlı içerik üzerinde bir değişim yapmayı amaçlar. Bu saldırılar Tip1-B saldırıları olarak tanımlanmıştır.

3.6.1. Tip1-A Saldırıları

Tip1-A saldırıları, saldırıların ayırt edici karakteristiklerinden dolayı tekrar üç alt kategoriye ayrılmıştır.

3.6.1.1. Silen Saldırılar

Silen saldırıların amacı ana veri içerisine gömülen filigranı silerek veya filigranın enerjisini aşırı derecede azaltarak, filigran bulucunun filigranı bulamamasını sağlamaktır.



Şekil 3.15. Saldırıların sınıflandırılması

Bu amaçtaki siliciler, basit siliciler ve analizsel siliciler olarak iki kategoride incelenmektedir [154]. Basit silicili saldırılarda, filigranı silmek için filigranlı verinin analizine ihtiyaç duyulmadığından dolayı bu saldırılar basit olarak tanımlanırlar. Basit silicili saldırılar doğrudan doğruya filigranlı veri üzerinde işlem yaparlar. Bu yüzden, hem ana verinin hem de filigranın görsel kalitesini etkilerler. Ancak filigran sinyalinin enerjisi ana veriyi temsil eden sinyalin enerjisine göre oldukça düşük olduğu için, saldırgan silme işleminde, ana verinin kalitesini etkilemeden önce filigranın silinmesini veya enerjisinin yeteri kadar azalmasını beklemektedir. Basit silicili saldırı işlemleri aşağıda verilmiştir:

- Kayıplı Sıkıştırma: JPEG gibi kayıplı sıkıştırma teknikleri önemsiz sayılan imge verisinin bir kısmını atarak sıkıştırma işlemini gerçekleştirirler. Bu atılan kısmın miktarı kullanılan kalite faktörüne bağımlıdır. Filigranlar genellikle imgelerin görsel kalitesinin etkilememek için bu önemsiz sayılan veri içerisine gömülürler. Bu yüzden sıkıştırma işleminde filigranlar, silinebilirler veya önemli miktardaki enerjilerini kaybederler.
- Analog-Sayısal ve Sayısal-Analog Dönüştürücüler: Sayısal bir imgenin basılmasında analog-sayısal bir dönüşüme; taranmasında ise sayısal-analog bir dönüşüme ihtiyaç duyulur. Ancak bu dönüştürme işlemlerinin sonucunda, hem filigran hem de filigranlı veri üzerinde bir bozulmaya neden olunur. Buradaki bozulma miktarı baskılama ve tarama işlemlerinde kullanılan cihazların karakteristiğine bağımlıdır.
- Kod Çevrimi: Sayısal imgeler bir imge formatından diğerine dönüştürülebilir. Ancak bu dönüşüm içerisinde, yeniden kodlamadan dolayı oluşan bozulma filigranı silebilir veya enerjisini önemli miktarda azaltabilir.
- Genel Filtreleme: Genel filtreleme teknikleri filigranlı bir içerik içerisinden filigranı silmek amacıyla kullanılabilir. Sözde rastlantısal gürültü modeliyle oluşturulmuş bir filigran yüksek frekanslı bir gürültüdür. Eğer filigranlı içerik filigran olarak bu gürültü modelini kullanıyorsa, alçak geçiren bir filtreleme filigranı silmek için oldukça kullanışlı olacaktır.

Analizsel silme saldırılarında saldırgan, filigranlı veriyi analiz ederek ana veri veya filigran hakkında bilgi elde etmeye çalışır. Bu bilgi daha sonra filigranı silmek için kullanılır. Bu saldırılar oldukça ayrıntılıdır ve genellikle kasıtlı olarak yapılır. Basit saldırılardan farklı olarak, filigranlı veri içerisinden veri silinirken, ana verinin kalitesi korunur. Bu saldırı türünde kullanılan işlemler aşağıda verilmiştir:

- Lineer Olmayan Filtreleme: Bir saldırgan lineer olmayan bir filtre kullanarak bir imge içerisine gömülen filigranı tahmin edebilir. Bu tahmin daha sonra filigranı silmek için kullanılabilir. Bu saldırının bir örneği WRS *(Watermark Removing Scheme)* saldırısıdır [155].

- İstatistiksel Ortalama: Bir saldırgan aynı filigranın gömüldüğü farklı imgelere sahipse, bu imgelerin istatistiksel ortalamalarını kullanarak filigranı tahmin edebilir. Bu saldırı özellikle, filigran içeriğe bağımlı değilse etkilidir.
- Danışıklı *(işbirlikçi-collusion)* Saldırı: Bu saldırıda, saldırgan grubundaki her bir üye farklı filigranı barından aynı imgelere sahiptir. Saldırganlar imgelerin ortalamasını alarak özgün imgenin bir kopyasını oluşturabilirler.
- Yerleştirici/Bulucu Gözlemleme: Bu saldırıda, saldırgan bir filigran bulucuya sahiptir. Saldırgan filigranlı içeriğin özelliklerini değiştirerek (imgenin parlaklık değerini değiştirmek gibi) filigran bulucunun cevabını gözlemler. Saldırganın amacı, filigranlı içerik üzerinde bulucunun filigranı bulamayacağı en küçük değişimi yapmaktır.

3.6.1.2. Senkronizasyon Saldırıları

Senkronizasyon saldırılarında amaç, filigranın senkronizasyonunu bozarak filigran bulucunun filigranı bulmasını engellemektir [154]. Bu saldırılarda, filigranın kendisi veya büyük bir bölümü hala ana veri içerisindedir. Basit silme saldırılarında olduğu gibi, saldırgan filigranlı verinin analizini yapmaz. Senkronize bir saldırı filigranlı veriye geometrik işlemler uygulanarak yapılmaktadır. Bu işlemler imgenin uzaysal alanı üzerinde yapıldığında uzaysal senkronizasyon saldırıları olarak adlandırılır. Bu saldırıda kullanılan bazı işlemler aşağıda verilmiştir:

- Geometrik Dönüşümler: Döndürme, taşıma, ölçeklendirme gibi dönüşümler filigranın senkronizasyonunu bozarak filigran bulucunun filigranı bulmasını engeller.
- Piksel Silme/Çıkarma: Bu saldırı imgeden piksellerin silinmesi veya çıkartılmasıyla yapılır. Örneğin, imgeden bir satır veya bir kolondaki piksellerin tümünün silinmesi filigran bulucunun senkronizasyonunu bozabilir. Eğer imge boyutları korunması istenirse, imgedeki bir kolon veya satır, imge içerisindeki başka bir kolon veya satır ile değiştirilir. Bu saldırı filigranlı verinin kalitesinde önemli bozulmalara neden olmaz.
- Mozaik Saldırılar: Mozaik saldırı tipinde, imge küçük parçalara bölünerek, bulucunun bu parçalar içerisinde filigranı bulamaması sağlanır. Asıl imge daha sonra bu küçük parçalar bir araya getirilerek yeniden oluşturulur [71].
- Karıştıran Saldırılar: Bu saldırıda, imgenin pikselleri karıştırılarak filigran bulucuya verilir. Bu karıştırma işlemi, basit bir değişim ile veya daha karmaşık bir şekilde sözde rastlantısal bir yapıda olabilir.

3.6.1.3. Belirsizlik Oluşturan Saldırılar

Belirsizlik oluşturan saldırı *(ambiguity attack)* türünde, saldırgan filigranlı veri içerisine ikinci bir filigran ekleyerek bulucunun filigranı bulmasını engeller [154]. Bu saldırının daha karmaşık bir türünde ise saldırgan filigranlı imge içerisinde kendi filigranının bulunduğunu ispatlar [156, 157]. Diğer bir belirsizlik oluşturan saldırı türü ise kopya saldırılarıdır [158]. Bu saldırıda, saldırgan filigranlı veri içerisindeki filigranı kopyalar ve başka ana veriler içerisine gömer. Ayrıca saldırgan kopya saldırısını gerçekleştirirken, filigranı gömerken kullanılan anahtara veya gömme algoritması ile ilgili herhangi bir bilgiye sahip değildir. Kopya saldırısı sonucunda bulucu, sahte filigranlı imgeyi ve bulunan filigranı doğrular.

3.6.2. Tip1-B Saldırıları

Bu kategorideki saldırıların amacı filigrana zarar vermeden filigranlı veri üzerinde değişimler yapmaktır. İmgelerin doğrulanması uygulamalarında, bu saldırı tipi sıkça kullanılmaktadır [39, 42]. Örneğin, saldırgan kendisine karşı mahkemede kanıt niteliği olan bir filigranlı imge üzerinde, suçu ifşa eden nesneyi filigranı bozmadan değiştirmeyi veya silmeyi isteyebilir.

3.6.3. Tip2 Saldırıları

Tip2 saldırılarında saldırgan, filigranlama sisteminin yazılım veya donanım bileşenleri üzerine saldırı gerçekleştirir.

3.6.3.1. Yazılım Değiştirme Saldırıları

Filigran gömücü ve bulucu yazılımsal olarak gerçekleştirilmişse ve bu yazılım dağıtılmışsa, saldırgan tarafından bu yazılımın değiştirilmesi mümkündür. Bu saldırıda saldırgan ilk olarak gömücü/bulucu yazılımını bulur ve yazılımı kaynak koda dönüştürür. Daha sonra kaynak kod içerisinde kodun filigranı üreten ve bulan kısmı belirlenir. Son olarak, bulunan bu kısım saldırganın amacı doğrultusunda değiştirilir.

3.6.3.2. Donanımsal Değiştirme Saldırıları

Bu saldırılar, filigranlama sistemini barından donanımsal bileşen üzerinde gerçekleştirilir. Örneğin saldırgan DVD oynatıcı cihaz üzerinde filigranlama sistemini barındıran devreyi kendi amaçları doğrultusunda değiştirebilir.

4. KIRILGAN FİLİGRANLAMA TEKNİKLERİ

Gürbüz filigranlardan farklı olarak kırılgan filigranların saldırılara karşı dayanıklı olmamaları istenir. Kırılgan filigranlar tam kırılgan, yarı kırılgan ve içerik bağımlı kırılgan filigranlar olarak sınıflandırılırlar. Tam kırılgan filigranlar, filigranlı içerikteki en küçük değişiklikte yok olacak şekilde tasarlanırlar [34-54]. Yarı kırılgan filigranlar, gürbüz ve tam kırılgan filigranların özelliklerini bir arada barındırırlar [6-13]. Bu filigranlar, içeriğin değiştirilmesi gibi kötü niyetli saldırılara karşı kırılgan, fakat sıkıştırma gibi içeriğin yapısını değiştirmeyen işlemlere karşı gürbüz bir yapıda oluşturulurlar. İçerik bağımlı kırılgan filigranlar ise öznitelik vektörlerinden oluşturulurlar [159-163]. Öznitelik vektörü, içeriğin değişmeyen özelliklerinden çıkartılan bir vektördür. Bu vektör filigranlama tekniğinin performansı üzerinde önemli bir etkiye sahiptir.

Bu tez çalışmasında tam kırılgan filigranlar üzerine odaklanılmış ve içerik olarak sayısal imgeler kullanılmıştır. Tam kırılgan filigranlar imge doğrulama uygulamalarında oldukça kullanılmaktadır. Bu uygulamalarda tam kırılgan bir filigranlama tekniğinden beklenilen önemli nitelikler aşağıda verilmiştir. Literatürde önerilmiş Kırılgan Filigranlama Teknikleri ve bu tez çalışmasında önerilen teknikler bu niteliklere göre değerlendirilecektir.

Bütünlük Doğrulama: Önerilen teknik, imge bütünlüğünü doğrulama niteliğine sahip olmalıdır. Bir bitlik bir değişim bile imgenin doğrulanamaması ile sonuçlanmalıdır.

Kaynak Doğrulama: Önerilen teknik, imgenin kaynağını doğrulayabilmelidir.

İnkar Edememe: İmgeyi filigranlayan kişi daha sonra filigranladığı imgeyi inkar edememelidir.

Değişim Bölgesi Belirleme: Sayısal bir imge genellikle büyük miktarda veriden oluşmaktadır. İmge üzerinde bir değişim yapıldığında, önerilen teknik değişim yapılan bölgeyi belirleyebilmelidir. Değişim bölgesinin belirlenmesinde önemli olan, bölgenin yüksek doğrulukta bir çözünürlükte belirlenebilmesidir. Değişim bölgesinin yüksek doğrulama çözünürlüğünde belirlenebilmesi, içeriği önemli olan imge uygulamalarında oldukça önemlidir. Örneğin mahkemede kanıt niteliği taşıyan imgelerde, değişim bölgesinin hassas olarak belirlenmesi istenir. Ayrıca önerilen teknik, değişim yapılan bölgenin dışında kalan alanları doğrulayabilme yeteneğine de sahip olmalıdır. Değişim bölgesi belirleme niteliğine sahip bir teknik tasarlandığında, tekniğin güvenliği ile değişim bölgesi belirleme çözünürlüğü arasındaki denge iyi sağlanmalıdır.

Algısal Benzerlik: Önerilen filigranlama tekniği, özgün imge ile filigranlı imge arasındaki algısal benzerliği korumalıdır. Doğrulama uygulamaları için önerilen Kırılgan Filigranlama

Teknikleri'nde, filigranlı imge ile özgün imge arasındaki algısal benzerliği sağlamak için filigran genellikle imgenin EÖB düzlemi içerisine gömülür.

Unutkan (Kör) Buluculu Sistem: Filigranlı imgenin doğrulanması esnasında, özgün içeriği kullanmayan sistemler "unutkan, kör" sistemler olarak adlandırılır. İmge doğrulama uygulamalarında, doğrulayıcı tarafından özgün imgeye erişme olanağı bulunmamaktadır. Bu yüzden doğrulama sistemleri için önerilen Kırılgan Filigranlama Teknikleri'nin unutkan buluculu bir özelliğe sahip olması istenir.

Güvenlik: Önerilen filigranlama algoritmasıyla ilgili tüm bilgiye sahip olan yetkilendirilmemiş üçüncü bir kişinin sahte bir filigranlı imge oluşturması mümkün olmamalıdır. Filigranlama algoritmalarının güvenliği Kriptografi'de olduğu gibi gizli anahtarların kullanımıyla sağlanmalıdır. Ayrıca önerilen teknik bilinen filigranlama saldırılarına karşı güvenilir olmalıdır. Doğrulama uygulamaları için önerilen filigranlama tekniklerinin daha çok asimetrik anahtar yapısına sahip olması istenir [38].

İmgedeki değişim bölgesini belirleyebilen ilk kırılgan filigranlama tekniklerinden biri Walton [44] tarafından önerilmiştir. Bu teknikte, sahte rasgele seçilen piksellerin EÖB düzlemi içerisine, bu piksellerin en önemli 7 bitinin sınama toplamının gömülmesi önerilmiştir. Bu teknik, bir pikselin yalnızca bir bitinde değişim olduğunda bozulmayı belirleyebildiği için, sınırlı bir değişim belirleme niteliğine sahiptir. Ayrıca bu teknikte pikseller üzerinde değişimler yapılarak, sahte bir filigranın geçerli olarak doğrulanmasının mümkün olması önemli bir güvenlik açığına neden olmaktadır.

Yeung ve Mintzer [45] tarafından, imgedeki her bir pikselin bütünlüğünü doğrulayabilecek basit ve hızlı bir kırılgan filigranlama tekniği önerilmiştir. Bu teknik imge içerisinde yapılan piksel değişimlerini belirleyebilmesine rağmen, imgenin boyutlarındaki değişimleri belirleyememektedir [38]. Daha sonra Yeung ve Mintzer tarafından önerilen teknik, JPEG sıkıştırma alanında imge doğrulanması için nicemlenmiş Ayrık Kosinüs Dönüşüm katsayılarında kullanılmıştır [46]. Ancak Yeung ve Mintzer tarafından önerilen teknikte, filigran olarak kullanılan logonun gizliliğinin sağlanamaması durumunda, tekniğin saldırılara karşı açık olduğu gösterilmiştir [47]. Ayrıca filigran olarak aynı logonun birden fazla imge içerisinde kullanılması durumunda, logo ve anahtar bilgisine ihtiyaç duyulmadan bir saldırı yapılabileceği de belirtilmiştir [48].

Değişim bölgesi belirleme yeteneğine sahip diğer bir teknik Wong [37] tarafından önerilmiştir. Bu teknikte doğrulanacak olan imgeler üst üste gelmeyen bloklara bölünerek, her bir bloğun EÖB düzlemi içerisine blok piksellerinden hesaplanan bir mesaj doğrulama kodu gömülmüştür. Mesaj Doğrulama Kodları'nda, simetrik anahtarlı bir yapı kullanılmaktadır. Dolayısıyla bu tekniğin güvenliği, kullanıcı anahtarının gizliliğinin korunmasıyla

sağlanmaktadır. Simetrik anahtarlı sistemlerde kullanıcı anahtarı hem özütün kriptolanmasında hem de mesaj doğrulama kodunun çözülmesinde kullanılmaktadır. Bu yüzden filigranlı bir imgenin doğrulanması sadece gizli anahtara sahip kişi tarafından yapılabilmektedir. Gizli anahtarın öğrenilmesi ise sistemin güvenliğini tehlikeye atmaktadır. Bu problemi çözmek için Wong [38] tarafından simetrik anahtarlı yapının asimetrik anahtarlı sürümü önerilmiştir. Bu sürümde Mesaj Doğrulama Kodları yerine asimetrik anahtar yapısına dayanan Sayısal İmzaların kullanılması önerilmiştir. Asimetrik anahtarlı sistemlerde özütün kriptolanmasında özel anahtar, Sayısal İmzaların çözülmesinde ise genel anahtar kullanılır. Sayısal imgenin doğrulanması genel anahtar kullanılarak yapılacağından sistemin güvenliği de tehlikeye atılmaz. Ayrıca filigran olarak Sayısal İmzaların kullanılmasıyla, Sayısal İmzaların sahip olduğu kaynak doğrulama ve inkar edememe niteliği de filigranın bünyesine katılır. Wong tarafından önerilen bu tekniklerde, değişim bölgesi seçilen blok büyüklüğüne bağımlı olarak belirlenmektedir.

Hem Yeung ve Mintzer hem de Wong tarafından önerilen tekniklerde filigranlama işlemi diğer bloklardan bağımsız olarak yapıldığı için bu teknikler "bloksal bağımsız filigranlama teknikleri" olarak adlandırılır. Bloksal bağımsız filigranlama tekniklerinin, VNTS'ye karşı korunmasız olduğu gösterilmiştir [39]. Bu saldırıda filigran olarak kullanılan logoya ve kullanıcı anahtarına ihtiyaç duyulmadan sahte imgelerin özgün bir filigran ile filigranlanması mümkündür. Yeung ve Mintzer'in tekniği daha küçük blok büyüklüğüne (1x1 piksel) sahip olduğu için, VNTS'ye karşı daha açıktır.

VNTS'yi önlemek için blok boyutlarının büyütülmesi, blok özütünün hesaplanmasında blok indislerinin eklenmesi, daha karmaşık logoların kullanılması önerilmiştir [39]. Ancak bu öneriler saldırı yapılmasını önleyememiş, sadece saldırı yapılmasını zorlaştırmıştır. Etkili bir çözüm ise bloklar arasında bağımlılığın kurulmasıyla sağlanmıştır [39]. Bloksal bağımlılıkta her bir blok içerisine gömülecek özütün hesaplanmasında, hem bloğun pikselleri hem de bloğa komşu seçilen diğer bloğun pikselleri birlikte kullanılır. Ancak bloksal bağımlılıkta, bir bloğa ait bir pikselin bozulması her iki bloğun birden doğrulanamamasına neden olur. Diğer bir çözüm ise her bir blok içerisine o bloğu kapsayan daha büyük bir bloğun özütünün gömülmesi ile elde edilmiştir [49]. Ancak bu çözüm saldırı yapılmasını bir dereceye kadar önlerken, değişim bölgesi belirleme doğruluğunu kaybetmiştir [50]. Alternatif bir çözüm olarak blok özütleri içerisine her bir imgeye ait benzersiz bir indisin gömülmesi önerilmiştir [51]. Bu öneri VNTS'nin yapılmasını tamamıyla önler. Ancak aynı imge indisinin doğrulama yordamında da kullanılması gerekir. İmge veritabanlarında yapılacak doğrulamalarda imge indislerinin yönetilmesi büyük bir yük getirecektir. Bu problemi çözmek için imge indisinin imge piksellerinden hesaplanması önerilmiştir [40, 60, 61]. Ancak imge üzerinde bir değişim yapılması sonucunda yanlış imge indisinin çıkarılması, önerilen tekniğin değişim bölgesi

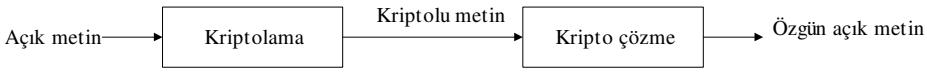
belirleme niteliğini yok edecektir. Literatürde VNTS'nin yanı sıra Kırılgan Filigranlama Teknikleri üzerinde etkili olan Doğum Günü Saldırısı [2], Kes-Yapıştır Saldırısı [42] ve Aktarıma Dayalı Saldırı [42] önerilmiştir.

Literatürde hem saldırılara karşı güvenilir hem de değişim bölgesi belirleme nitelliğine sahip bir kırılgan filigranlama tekniği Çelik ve diğ. [52] tarafından önerilmiştir. Bu teknikte imge sıradüzensel bir yapı içerisinde alt bloklara bölünür. Sıradüzenin en üst seviyesindeki bloğa (imgenin kendisi) ait sayısal imza saldırılara karşı tam bir güvenlik sağlarken, en alt seviyedeki bloklara ait sayısal imzalar değişim bölgesinin belirlenmesinde kullanılırlar. Sıradüzensel Filigranlama Tekniğinin değişim bölgesi belirleme doğruluğu imge çözünürlüğüne bağımlıdır. Eğer imge sıradüzenin üst seviyelerinde bir alt seviyeye bölünemezse, tekniğin değişim bölgesi belirleme doğruluğu oldukça azalmaktadır. Sıradüzensel yapının en alt seviyesindeki bloklar hem kendi bloklarına ait sayısal imzaları hem de sıradüzensel seviyeye bağımlı olarak üst seviyelerdeki bloklara ait sayısal imzaların bir parçasını taşımaktadırlar. Dolayısıyla seçilen sayısal imza algoritmasının bit uzunluğu en alt seviyedeki blok büyüklüğünü belirlemektedir. Daha yüksek bir güvenlik seviyesi için daha uzun bit çıkışına sahip bir sayısal imza algoritmasının kullanılması, Sıradüzensel Filigranlama Tekniğinin değişim bölgesi belirleme doğruluğunu azaltmaktadır.

Bu tezde önerilen kırılgan filigranlama tekniklerini daha iyi açıklayabilmek ve değerlendirebilmek için bu bölümde: Özüt fonksiyonları, simetrik ve asimetrik anahtar yapılı kriptolama, Mesaj Doğrulama Kodları ve Sayısal İmzalar açıklanmıştır. Ayrıca bu bölümde, tezde önerilen tekniklerin temelini oluşturan Wong'un tekniğinin açıklanması ve uygulanması, bu tekniğe karşı yapılan saldırılar ve alınan önlemler incelenmiştir.

4.1. Kriptografi

Kriptografi, mesajların güvenliğini koruma bilimidir [1-4]. Mesaj kelimesi metin, sayısal veri gibi herhangi bir bilgiyi temsil etmektedir. Kriptografi'de gizlenilmek istenen mesaj, açık metin *(plaintext, cleartext)* olarak tanımlanır. Kriptolama, mesajı gizlemek için mesajın görünümünün değiştirilmesi işlemidir. Bir mesajın kriptolanması ile oluşturulan veri kriptolu metin *(ciphertext)* olarak tanımlanır. Kriptolu bir mesajı açık mesaja dönüştürme işlemi ise kripto çözme olarak bilinir. Kriptolama ve kripto çözme işleminin blok yapısı şekil 4.1. ile verilmiştir.



Şekil 4.1. Kriptolama ve kripto çözme işlemi

Kriptografik bir algoritma, kriptolama ve kripto çözme için kullanılan matematiksel bir fonksiyondur. Modern Kriptografi'de bir algoritmanın güvenliği, gizli anahtarların kullanımıyla sağlanmaktadır. Bu anahtarlar simetrik veya asimetrik yapıda olabilir.

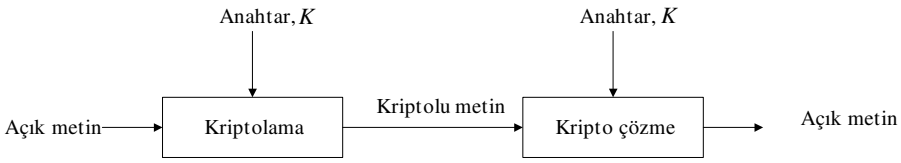
4.1.1. Simetrik Anahtarlı Kriptografi

Simetrik anahtarlı algoritmalarda, kriptolama anahtarı kripto çözme anahtarından veya kripto çözme anahtarı, kriptolama anahtarından hesaplanabilir [2]. Simetrik algoritmaların çoğunda kriptolama ve kripto çözme için aynı anahtar kullanılır. Bu algoritmalar tek anahtarlı algoritmalar olarak da bilinir. Simetrik anahtarlı algoritmalarda mesajın göndericisi ve alıcısı, haberleşme işlemine başlamadan önce bir anahtar üzerinde anlaşmaları gerekir. Simetrik bir algoritmanın güvenliği anahtarın gizliliği korunarak sağlanır. Simetrik bir algoritmada kriptolama işlemi eşitlik 4.1. ve kripto çözme işlemi eşitlik 4.2. vasıtasıyla gerçekleştirilir.

$$E_K(m) = c \tag{4.1}$$

$$D_K(c) = m \tag{4.2}$$

Burada E kriptolama algoritmasını, K kriptolama ve kripto çözme anahtarını, m mesajı, c kriptolu metni ve D kripto çözme algoritmasını tanımlamaktadır. Simetrik algoritmalarda anahtarın istenilmeyen üçüncü bir kişi tarafından bilinmesi mesajın güvenirliliğini yok eder. Simetrik anahtarlı algoritmalardaki diğer bir problem ise ikiden fazla kişinin haberleşmek istemesiyle ortaya çıkmaktadır. Bu durumda güvenilir bir ortamda haberleşmek isteyen tüm kişiler kriptolama ve kripto çözmek için aynı anahtarı kullanacağından ortamda mesajın kimin tarafından gönderildiği belirlenemeyecektir. Simetrik anahtarlı kriptografi yapısı şekil 4.2. ile verilmiştir.



Şekil 4.2. Simetrik anahtarlı kriptografi

4.1.2. Asimetrik Anahtarlı Kriptografi

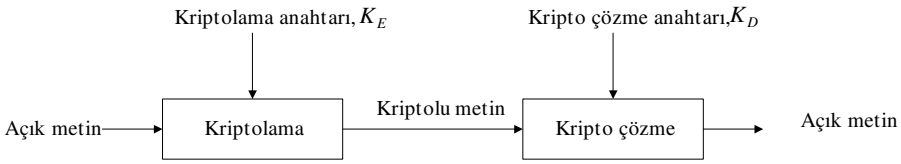
Asimetrik anahtarlı kriptografi, simetrik anahtarlı kriptografi tarafından çözülemeyen problemlere çözüm sağlamaktadır [1, 2]. Asimetrik anahtarlı kriptografide, kriptolama için kullanılan anahtar, kripto çözme için kullanılan anahtardan farklıdır. Mesajı güvenilir bir şekilde iletmek için bir çift anahtar kullanımına ihtiyaç duyulur. Mesaj, K_E kriptolama anahtarı kullanılarak kriptolanır. Kriptolama işlemi eşitlik 4.3. ile tanımlanır.

$$E_{K_E}(m) = c \tag{4.3}$$

Kriptolu metni çözmek için ise K_D kripto çözme anahtarı kullanılır. Kripto çözme işlemi eşitlik 4.4 ile tanımlanır.

$$D_{K_D}(c) = m \tag{4.4}$$

Asimetrik anahtarlı kriptografide, her bir kriptolama anahtarına karşılık gelen bir kripto çözme anahtarı mevcuttur. Bu sistemler ideal olarak herhangi bir anahtardan diğerinin elde edilmesine izin vermeyecek şekilde tasarlanır. Ancak RSA (Rivest, Shamir and Adleman) gibi popüler asimetrik algoritmalarda bu yapı tek yönlü olarak sağlanabilmektedir. Tek yön ifadesi herhangi bir anahtardan diğerinin elde edilebileceğini, fakat bunun tersinin olamayacağını açıklamaktadır. Kriptografi'de elde edilebilen anahtar genel *(public)* anahtar olarak ve elde edilemeyen anahtar özel *(private)* anahtar olarak adlandırılır. Genel anahtar herhangi bir risk oluşturmayacağı için dağıtılabilir. Uygulamaya bağımlı olarak kriptolama veya kripto çözme anahtarı genel anahtar olabilir. Sayısal İmza Algoritmaları'nda, kriptolama anahtarı özel anahtar yapısında ve kripto çözme anahtarı genel anahtar yapısındadır. Şekil 4.3. asimetrik anahtar yapısındaki bir kriptolama sistemini göstermektedir.



Şekil 4.3. Asimetrik anahtarlı kriptografi

4.1.3. Tek Yönlü Özüt Fonksiyonları

Bir özüt fonksiyonu, değişken uzunluktaki bir diziyi sabit uzunluktaki bir diziye eşleştiren bir fonksiyondur [1]. Sabit uzunluktaki dizi, özüt olarak tanımlanır. Basit bir özüt fonksiyonu, bir karakter dizisindeki tüm byte'ların XOR'lanmasıyla oluşturulabilir. Burada karakter dizisinin uzunluğu ne olursa olsun, çıkıştaki özüt değeri bir byte uzunluğundadır. Özüt fonksiyonları çoktan teke bir dönüşüm gerçekleştirirler.

Tek yönlü bir özüt fonksiyonunda, değişken uzunluktaki giriş dizisinden sabit uzunluklu bir özüt değeri oluşturmak oldukça kolaydır. Ancak özüt değerinden ters bir dönüşüm ile giriş dizisini elde etmek mümkün değildir. Özüt fonksiyonu eşitlik 4.5. ile tanımlanmaktadır.

$$h = H(m) \tag{4.5}$$

Burada h özüt değerini ve H özüt fonksiyonunu belirtmektedir. Tek yönlü özüt fonksiyonunun diğer önemli bir özelliği, iki ayrı giriş dizisinden hiçbir zaman aynı özüt değerini üretmemesidir [1]. Bu özellik eşitlik 4.6. ile verilmiştir.

$$H(m) \neq H(m') \tag{4.6}$$

Sıkça kullanılan tek yönlü özüt fonksiyonları arasında MD5 (Message Digest 5) ve SHA (Secure Hash Algorithm) algoritmaları verilebilir.

4.1.4. Mesaj Doğrulama Kodları

Mesaj Doğrulama Kodu, gizli anahtar ilave edilmiş tek yönlü özüt fonksiyonudur [2]. Özüt değeri, giriş dizisi ve anahtarın bir fonksiyonudur. Mesaj Doğrulama Kodu eşitlik 4.7. ile tanımlanabilir.

$$h = H(m, K) \tag{4.7}$$

Mesaj Doğrulama Kodları bir mesajın bütünlüğünü doğrulamak için kullanılmaktadır. Doğrulama uygulamalarında ilk olarak gönderici ve alıcı gizli bir anahtar üzerinde anlaşırlar. Daha sonra gönderici mesaj ile birlikte mesaj doğrulama kodunu gönderir. Alıcı anahtarı ve mesajı kullanarak yeniden mesaj doğrulama kodunu hesaplar. Gönderilen mesaj doğrulama kodu ile hesaplanan mesaj doğrulama kodunun eşleşmesi mesajın doğrulanmasını sağlar. Ancak simetrik anahtarlı kriptografide olduğu gibi, haberleşme ortamında birden fazla kişi varsa mesajın kimin tarafından gönderildiği belirlenemez.

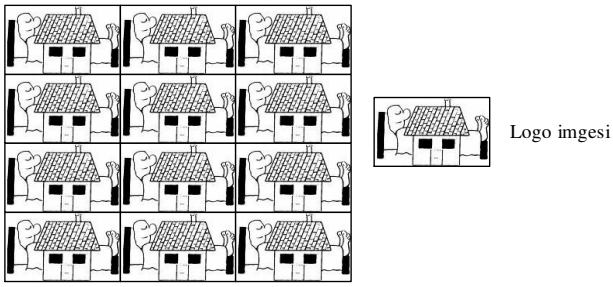
4.1.5. Sayısal İmzalar

Mesaj Doğrulama Kodları gibi Sayısal İmzalar'da mesajların bütünlüğünü doğrulamak için kullanılır [1-4]. Ancak Mesaj Doğrulama Kodları'ndan farklı olarak Sayısal İmzalar asimetrik anahtar yapısını kullanırlar. Asimetrik anahtar yapısı, Sayısal İmza'ların mesajın kaynağını belirleyebilmesine ve mesajı gönderen kişinin göndermediğini inkar edememesini sağlamaktadır.

4.2. Wong'un Tekniği

Literatürde değişim bölgesi belirleme niteliğine sahip bir Kırılgan Filigranlama Tekniği Wong [38] tarafından önerilmiştir. Bu teknikte doğrulanacak olan imgeler üst üste gelmeyen bloklara bölünerek, her bir blok içerisine bloğun kendisine ait bir sayısal imza gömülmektedir. Bu teknikte sayısal imzanın kullanılmasıyla, sayısal imzaların sahip olduğu kaynak doğrulama, inkar edememe ve yüksek seviyeli kriptografik güvenlik filigranın bünyesine katılmıştır. Wong'un tekniği hem bu tezde önerilen kırılgan filigranlama tekniklerinin hem de teknik yazında önerilmiş bir çok kırılgan filigranlama tekniğinin temelini oluşturmaktadır [39-43, 52].

Wong, blok tabanlı bir kırılgan filigranlama tekniği önermiştir. Bu teknikte $M * N$ boyutlu bir I imgesi içerisine, imge ile aynı boyutlu ikili bir imge (b_{M*N}) filigran olarak gömülmektedir. b_{M*N} filigran imgesi, daha küçük boyutlu ikili imgesel logonun şekil 4.4.'deki gibi döşenmesiyle oluşturulur.

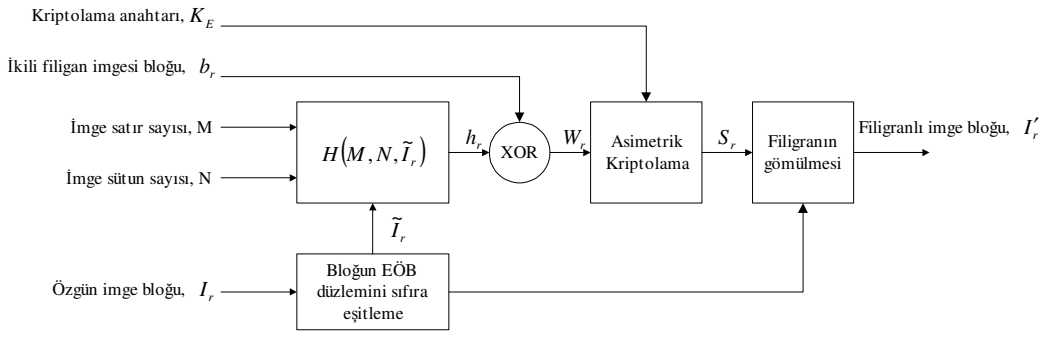


Şekil 4.4. Logo imgesinden imge ile aynı boyutlu ikili filigran imgesinin oluşturulması

Wong tarafından önerilen filigranlama yordamı şekil 4.5. ile verilmiştir. Filigranlama yordamında ilk olarak I özgün imgesi ve b_{M*N} filigranı üst üste gelmeyen $J*L$ piksellik $\{I_1, I_2, ..., I_n\}$ ve $\{b_1, b_2, ..., b_n\}$ bloklarına bölünür. $J*L$ piksellik blok büyüklüğü, seçilen sayısal imzanın uzunluğuna eşit veya büyük olmalıdır. Şekil 4.5.'de I_r ve b_r, $J*L$ piksellik imge ve filigran bloğunu temsil etmektedir. Her bir imge bloğunun özütü hesaplanmadan önce, blokların EÖB düzlemi sıfır yapılarak $\tilde{I}_r$ blokları oluşturulur. Her bir bloğa ait özüt değeri eşitlik 4.8. kullanılarak hesaplanır.

$$h_r = H\left(M, N, \tilde{I}_r\right) \tag{4.8}$$

İkili filigran imgesindeki her bir b_r bloğu ile bu bloğa karşılık gelen h_r özüt değeri XOR fonksiyonu kullanılarak eşitlik 4.9.'daki gibi birleştirilir. Daha sonra her bir bloğa ait sayısal imza, imge sahibinin özel anahtarı kullanılarak eşitlik 4.10.'daki gibi oluşturulur.



Şekil 4.5. Wong'un tekniğinin filigranlama yordamı

$$W_r = b_r \oplus h_r \tag{4.9}$$

$$S_r = E(W_r, K_E) \tag{4.10}$$

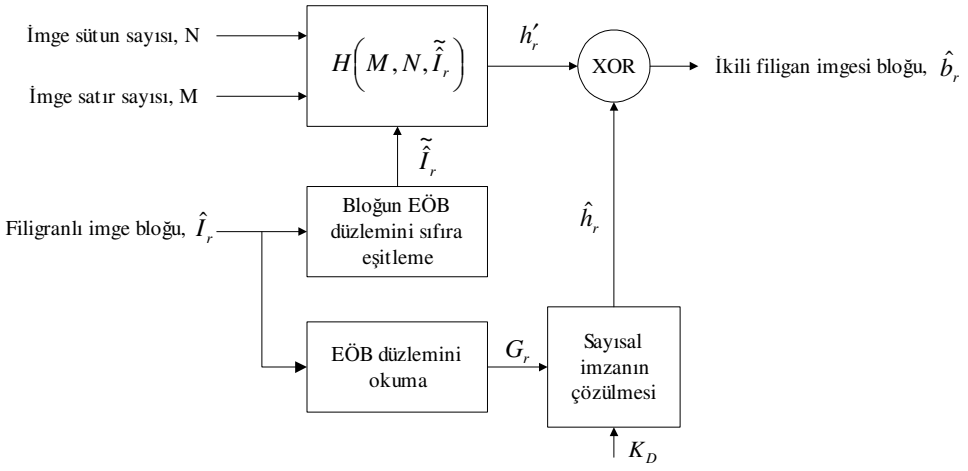
Son olarak her bir I_r bloğunu filiganlamak için, S_r sayısal imzası bloğun en az önemli bit düzlemi içerisine gömülür. Bu işlemler imge üzerindeki tüm bloklara diğer bloklardan bağımsız olarak uygulanarak filiganlı imge I' oluşturulur.

Filigranın çıkarılması ve teyit edilmesi yordamı şekil 4.6.'da gösterilmiştir. Bu yordamda muhtemelen bozulmuş filigranlı imge $\hat{I}$, filigranlama yordamında olduğu gibi üst üste gelmeyen $\{\hat{I}_1, \hat{I}_2, ...\hat{I}_r\}$ bloklarına bölünür. Daha sonra her bir $\hat{I}_r$ bloğu, $\tilde{\tilde{I}}_r$ ve G_r olarak iki parçaya ayrıştırılır. Burada $\tilde{\tilde{I}}_r$ EÖB düzlemi sıfır yapılan parçayı; G_r ise bloğun EÖB düzlemindeki veriyi barındıran parçayı tanımlamaktadır.

Blokları doğrulayabilmek için, blok özütlerinin yeniden hesaplanması ve gömülen özüt değerleri ile karşılaştırılması gerekir. Her bir bloğun özütü eşitlik 4.11. kullanılarak yeniden hesaplanır. G_r parçasının taşıdığı sayısal imza ise, genel anahtarlı kripto çözme algoritması kullanılarak eşitlik 4.12.'deki gibi çözülür.

$$h_r' = H\left(M, N, \tilde{\tilde{I}}_r\right) \tag{4.11}$$

$$\hat{h}_r = D(G_r, K_D) \tag{4.12}$$



Şekil 4.6. Wong'un tekniğinin doğrulama yordamı.

Eşitlik 4.11. kullanılarak hesaplanan özüt h_r' ile sayısal imzanın çözülmesiyle elde edilen $\hat{h}_r$ özütü XOR işlemine tabi tutularak ikili filigran imgesi elde edilir. Bu işlem eşitlik 4.13. ile verilmiştir.

$$\hat{b}_r = h_r' \oplus \hat{h}_r \tag{4.13}$$

İmge filigranlandıktan sonra imgenin piksel değerlerinin veya imge boyutunun değiştirilmesi durumunda, gömülen ikili filigran imgesinin değişim bölgesine karşılık gelen bloklarında bozulma oluşur. Wong'un tekniğinde herhangi bir bloktaki bir pikselin bir bitinin bile değiştirilmesi o bloğun tümünün doğrulanamamasına neden olur. Ancak bu teknikte, aynı filigran imgesini taşıyan filigranlı imge bloklarının yer değiştirmesi veya bir bloğun kopyalanarak bir başka bloğun yerine yapıştırılabilmesi mümkündür.

4.3. Kırılgan Filigranlama Saldırıları

Kırılgan filigranlara karşı yapılan saldırılar, gürbüz filigranlara karşı yapılan saldırılardan farklıdır. Kırılgan filigranlara karşı yapılan saldırıların amacı, filigrana zarar vermeden içeriğin değiştirilmesini sağlamaktır. Bu değiştirme işlemi iki şekilde olabilir. İlkinde, filigrana zarar vermeden filigranlı verinin belirli bölgeleri değiştirilerek özgün olduğu ileri sürülür. İkincisinde ise, filigran içermeyen sahte bir imge legal bir filigran ile filigranlanarak sahte imgenin özgünlüğü ispat edilmek istenir.

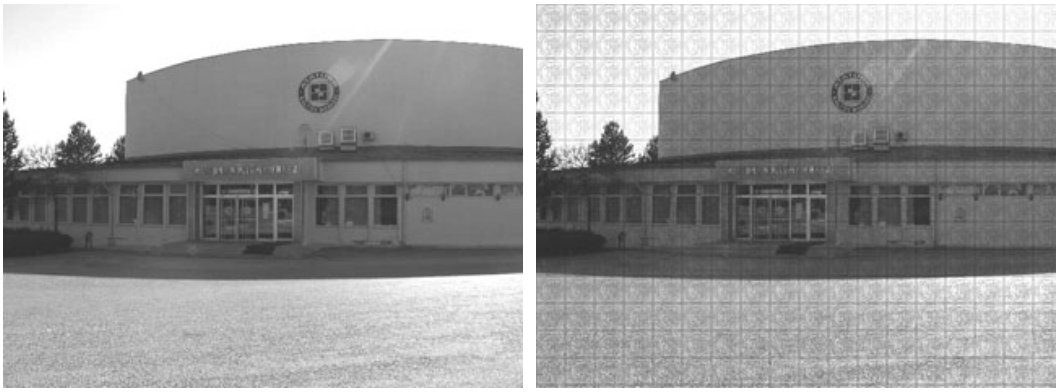
4.3.1. Kes-Yapıştır Saldırısı

Kes-yapıştır saldırısı bağımsız blok yapısıyla tasarlanan Kırılgan Filigranlama Teknikleri üzerinde etkilidir [39, 42]. Bu saldırıda, bloklar arasındaki bağımsızlıktan faydalanılarak imge içerisindeki bir blok kesilerek diğer bir bloğun yerine yapıştırılır. Ancak bu değişimin yapılabilmesi için blokların aynı özelliklere sahip olması gerekir. Örneğin Wong'un tekniğinde her bir bloğa gömülen sayısal imza, bloğun piksel değerleri, imgenin boyutları ve bloğa karşılık gelen ikili filigran imgesi kullanılarak oluşturulmaktadır. Bu yüzden değiştirilen blokların aynı imge boyutlarına ve aynı filigran modeline sahip olması gereklidir.

Kes-yapıştır saldırısı bir imge üzerinde şöyle açıklanabilir. Şekil 4.7.(a)'da verilen özgün imge, Wong'un tekniği ile filigranlanarak şekil 4.7.(b)'deki filigranlı imge elde edilmiştir. Aslında şekil 4.7. (b) üzerindeki filigran modeli görünmezdir, fakat saldırıyı daha iyi ifade edebilmek için görünür bir yapıda tasarlanmıştır. Şekil 4.7.(b) üzerinde Wong'un tekniği

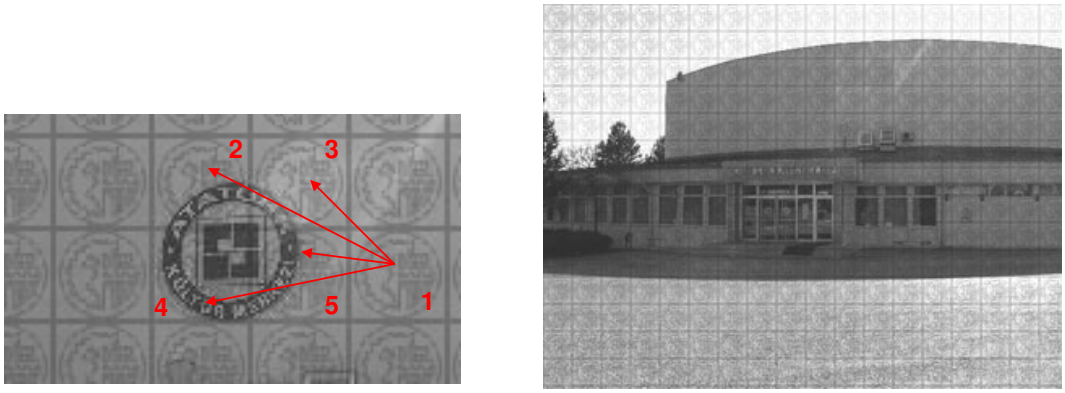
tarafından belirlenemeyecek bir değişim yapabilmek için aynı filigran modelini taşıyan blokların birbirleriyle değiştirilmesi gerekir. Şekil 4.7.(b)'de verilen filigranlı imgedeki binanın üzerinde bulunan logonun Wong'un tekniği tarafından belirlenmeden silinmesi istenirse, şekil 4.8. (a)'da "1" ile gösterilen bloğun kesilerek '2', '3', '4' ve '5' ile gösterilen blokların yerine yapıştırılır. Şekil 4.8. (b) saldırı yapılmış filigranlı imgeyi göstermektedir.

Bu saldırıda maskelenmek istenen bloklar aynı filigran modeline sahip bir bloğun kesilip yapıştırılmasıyla silinir. Saldırı yapılmış bloklar aynı filigran modeline ve aynı imge boyut değerlerine sahip olduğunda, blok bağımsız filigranlama teknikleri saldırıyı belirleyememektedir. Dolayısıyla bu teknikler saldırı yapılmış imgeyi özgünlüğünü doğrulayarak önemli bir güvenlik açığına neden olmaktadırlar.



(a) **(b)**

Şekil 4.7. Wong'un Tekniği ile filigranlama **(a)** Özgün imge **(b)** Filigranlı imge



(a) **(b)**

Şekil 4.8. Kes yapıştır saldırısı **(a)** Saldırı yapılan bölge **(b)** Saldırı yapılmış filigranlı imge

4.3.2. Vektör Nicemlemeli Taklit Saldırısı

Kırılgan Filigranlama Teknikleri, imge üzerinde yapılan değişimleri belirleyebilsin diye bloksal yapıda tasarlanırlar [37, 38, 44-46]. Ancak bu yapı içerisindeki bloksal bağımsızlık saldırı yapılmasına imkan sağlamaktadır. Kırılgan Filigranlama Teknikleri'ne karşı bloksal bağımsızlıktan faydalanılarak yapılan bir saldırı Vektör Nicemlemeli Taklit Saldırısı'dır [39]. Bu saldırıda, filigran sahibinin rızasına ve kriptolama anahtarına ihtiyaç duyulmadan özgün bir filigran sahte imgeler içerisine gömülebilir. Böylece sahte bir imge özgün bir filigran ile filigranlanabilir ve filigranın sahte imgenin aslıyla aynılığını kanıtlamasıyla önemli bir saldırı gerçekleştirilir. VNTS'yi gerçekleştirebilmek için bir saldırganın yeteri sayıda filigranlı imge bloğuna sahip olması gerekir. Saldırgan özgün bir filigran barındıran sahte bir imge oluşturmak için, sahte imge içerisindeki her bir bloğu filigranlı blok veritabanındaki benzer bir bloğa nicemler. Sahte imge içerisindeki her bir bloğun nicemlenmesi sonucunda, görsel olarak sahte imgeye benzeyen bir filigranlı imge oluşturulur. VNTS, şekil 4.9.'da gösterilmiştir.

Bu saldırıyı daha iyi açıklayabilmek için *bloksal bağımsızlık* ve K *-eşitliği* kavramlarını tanımlamak gerekir. Burada özgün yazımda kullanılan terimler dizgesi kullanılmıştır [39]. Blok tabanlı bir filigranlama tekniği, I imgesini üst üste gelmeyen $\{I_1, I_2, ..., I_n\}$ bloklarına böler ve W_r filigranını I_r içerisine K_r anahtarını kullanarak gömer. Eğer her bir filigranlı blok I_r'; sadece I_r özgün imge bloğuna, W_r filigranına ve K_r anahtarına bağımlıysa, blok tabanlı yaklaşımın "bloksal bağımsız" olduğu söylenir. Bloksal bağımsız bir filigranlama tekniği için filigran ekleme $\varepsilon_K(\cdot)$ ve çıkarım fonksiyonları $\mathcal{D}_K(\cdot)$ eşitlik 4.14. ve 4.15. ile tanımlanır.

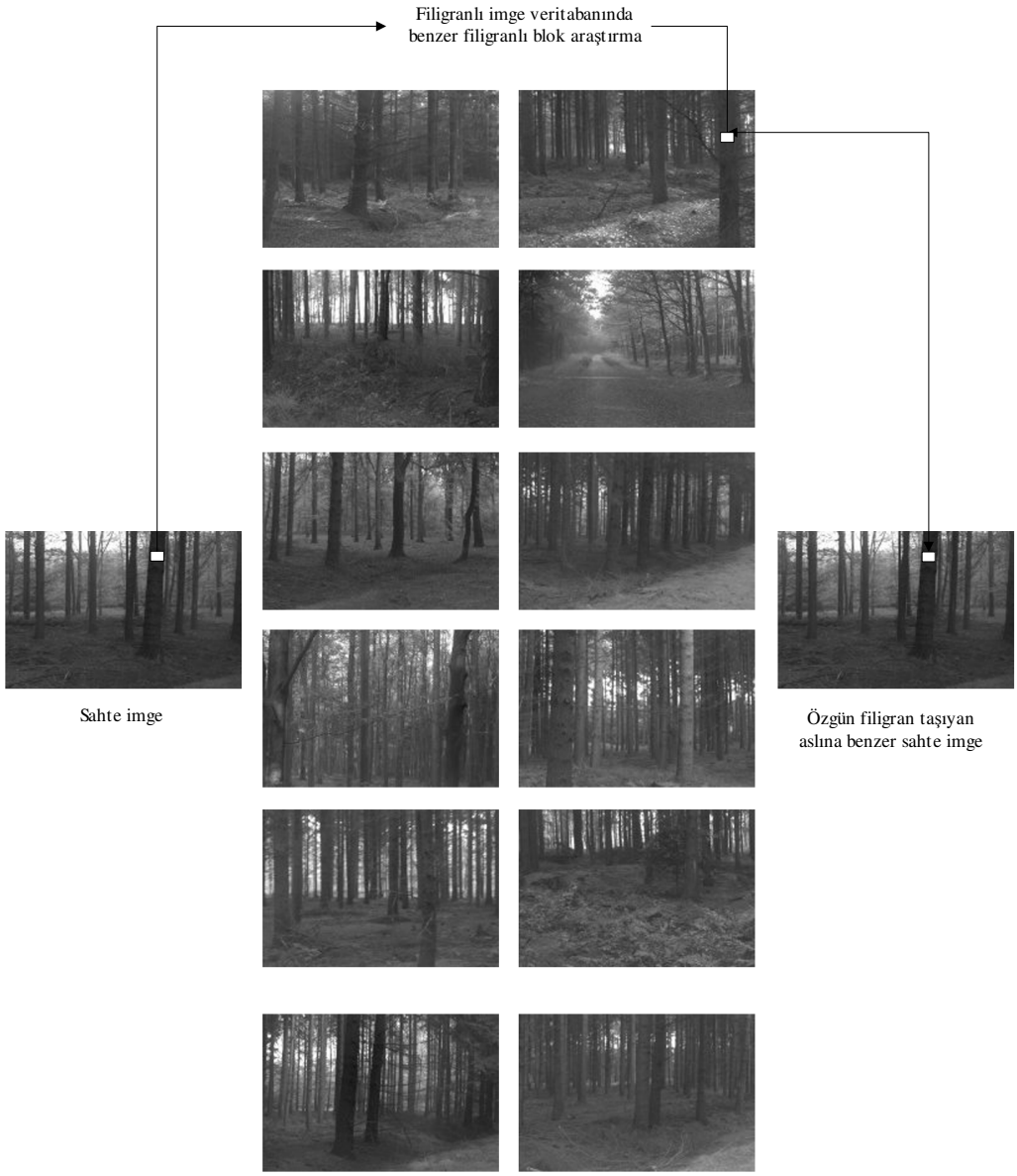
$$I' = \varepsilon_K(I, W) = \varepsilon_{K_1}(I_1, W_1) \| \varepsilon_{K_2}(I_2, W_2) \| ... \| \varepsilon_{K_n}(I_n, W_n) \tag{4.14}$$

$$\hat{W} = \mathcal{D}_K(\hat{I}) = \mathcal{D}_{K_1}(\hat{I}_1) \| \mathcal{D}_{K_2}(\hat{I}_2) \| ... \| \mathcal{D}_{K_n}(\hat{I}_n) = \hat{W}_1 \| \hat{W}_2 \| ... \| \hat{W}_n \tag{4.15}$$

Burada $K_1, K_2, ..., K_n$ tek bir K anahtarından elde edilen yerleştirme anahtarlarını, $W = W_1 \| W_2 \| .. \| W_n$ filigran modelini ve $\|$, birbirine bağlanmayı tanımlamaktadır.

Verilen bir K anahtarı I_i ve I_j imge bloklarından aynı filigranı çıkarıyorsa, bu iki imge bloğunun *K-eşitliği*'ne sahip olduğu söylenir. *K-eşitliği* eşitlik 4.16. ile tanımlanır.

$$\mathcal{D}_K(I_i) = \mathcal{D}_k(I_j) = W \tag{4.16}$$



Şekil 4.9. Vektör Nicemlemeli Taklit Saldırısı

Bu özellik, verilen bir K anahtarı ile imge içerisindeki tüm blokların kümesini $\{C_1, C_2, ..., C_m\}$ eşit sınıflar içerisine ayrıştırır. Eşit sınıf içerisindeki herhangi bir bloğa, K anahtarı kullanılarak filigran çıkarım işleminin uygulanması aynı filigranın bulunması ile sonuçlanır.

Vektör nicemlemeli taklit saldırısında saldırgan, bloksal bağımsız filigranlama tekniklerinin *K-eşitliği* özelliğini kullanarak saldırıyı gerçekleştirir. Saldırının yapılabilmesi için, bir K anahtarı ve W filigranı ile filigranlanmış yeterli miktarda I' imgesine sahip olunması gerekir. Filigranlı imge bloklarını farklı C_K eşitlik sınıfları içerisine tasnif etmek için

logo imgesi gibi kısmi bir filigran bilgisinin kullanılması genellikle yeterlidir. Oluşturulan C_K eşitlik sınıfları vektör nicemleme kod çizelgesi olarak da adlandırılır. Bu kod çizelgesi kullanılarak bir imge bloğu I_i', eşit sınıftaki başka bir imge bloğu V_i' ile değiştirilebilir veya daha ileri gidilerek özgün filigran barındıran sahte bir V' imgesi oluşturulabilir. Saldırgan, hem özgün bir filigranı barındıran hem de özgün V imgesine yeteri kadar benzeyen sahte bir V' filigranlı imgesini oluşturabilmek için aşağıdaki prosedürü uygular:

```
for i=1:n
   I_i' bloğunun ait olduğu K-eşitlik sınıfını belirleme
   V_i' ∈ C_K ve V_i' ≅ V_i sağlayan bir yaklaşım bulma
   V_i bloğunu V_i' bloğu ile değiştirme
   V' = V_1' || V_2' ||...,|| V_n' filigranlı sahte imgeyi oluşturma
```

Yukarıdaki prosedürde, W filigranını içeren bir $I' = \{I_1', I_2', ..., I_n'\}$ imgesi verilirse, $1 \le i \le n$ için I_i''nin K_i eşitliğinin V_i' ve $V' \cong V$ olduğu $V = \{V_1', V_2', ..., V_n'\}$ sahte imgesinin oluşturulabileceği ifade edilmiştir. Bu saldırıda C_k eşitlik sınıfına ait bir V_i''nin V_i'ye yakınlaştırılması imge bloğuna vektör nicemlemesi uygulanarak gerçekleştirilir. Burada kullanılan kod çizelgesi, I' içerisinde C_k'ya ait olduğu bilinen filigranlı blokların kümesiyle oluşturulur. Saldırgan C_k kümesindeki I' bloklarını biliyorsa, sadece küme içerisinde V_i''ne en yakın eşleştirmeyi bulması yeterli olacaktır.

4.3.3. Doğum Günü Saldırısı

Bloksal bağımsız Kırılgan Filigranlama Teknikleri'nin, Doğum Günü Saldırısı'na *(Birthday attack)* karşı açık olduğu gösterilmiştir [42]. Bu saldırı, "bir popülasyon içerisinde en az iki kişinin aynı gün doğma olasılığının 0.5'den daha büyük olması için minimum populasyon boyutu ne olmalıdır ?" sorusu temel alınarak geliştirilmiştir [2]. Doğum günü paradoksuna göre, her bir blok için l bit uzunluğunda bir dizi üreten bir özüt fonksiyonu kullanıldığında, $2^{l/2}$ tane filigranlı imge bloğu mevcutsa iki imge bloğunun özüt değerinin aynı olma olasılığı 0.5'den daha büyüktür. Bu saldırı aynı anahtar ve logo ile filigranlanmış yeteri sayıda filigranlı imgeden, özgün filigranı barındıran yeni bir filigranlı imge oluşturmak için kullanılır. Wong'un tekniğinde kullanılan 64 bit uzunluğundaki MAC doğrulama kodları için, bir saldırganın 2^{32} adet filigranlı imge bloğuna sahip olması aynı özütü veren blok çiftlerini bulması için yeterlidir.

4.3.4. Aktarıma Dayalı Saldırı

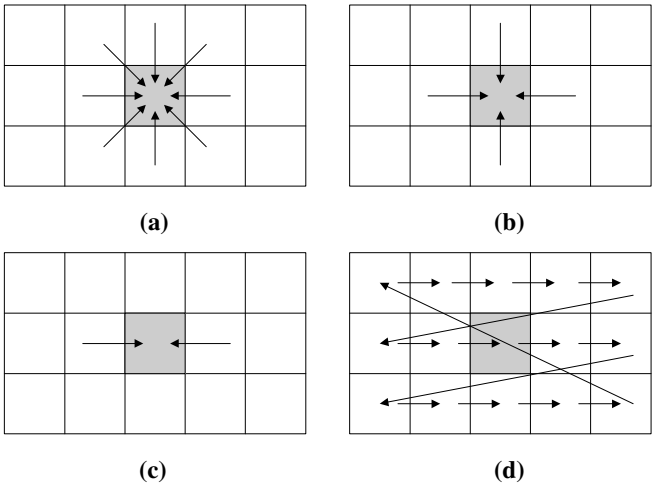
Barreto ve diğ [42], Aktarıma Dayalı Saldırı olarak adlandırılan başka bir saldırı önermişlerdir. Bu saldırı VNTS'ye karşı dayanıklı olan bazı Kırılgan Filigranlama Teknikleri üzerinde de etkilidir [42]. VNTS'ye karşı güvenirlik sağlayabilmek için bloklar arasında bağımlılığın kullanılması önerilmektedir [39]. Literatürde bloksal bağımlılığın kurulması için farklı öneriler sunulmuştur [39, 42, 52]. Bu önerilerden biri imge içerisindeki her bir bloğu bir önceki blok ile bağımlı hale getirmektir. Bölüm 4.4.5**.** içerisinde ayrıntılı olarak anlatılan bu teknik HBC1 (Hash block Chaining Version 1) olarak adlandırılmaktadır. Aktarıma Dayalı Saldırıyı HBC1 tekniği ile filigranlanmış imgeler üzerinde açıklayalım. I' ve I'' HBC1 tekniği kullanılarak oluşturulmuş filigranlı imgeler olsun. $I'_A \to I'_B$, I'_B bloğunun I'_A bloğuna bağımlı olduğunu ifade etmektedir. Yani I'_B blok özütü hesaplanırken I'_B pikselleri ile birlikte I'_A blok pikselleri de kullanılmıştır. I' ve I'' imgelerinin aşağıdaki blok yapısına sahip olduğunu farz edelim:

$$\cdots \to I'_A \to I'_D \to I'_B \to I'_C \to \cdots,$$
$$\cdots \to I''_A \to I''_E \to I''_B \to I''_C \to \cdots$$

Burada I'_A bloğu ile I''_A bloğunun, I'_B bloğu ile I''_B bloğunun ve I'_C bloğu ile I''_C bloğunun içeriği aynıdır. Ancak I'_D bloğu ile I''_E bloğunun içerikleri farklıdır. O halde I' imgesi içerisindeki (I'_D, I'_B) blok çifti I''imgesi içerisindeki (I''_E, I''_B) blok çifti ile HBC1 tekniğinin belirleyemeyeceği şekilde değiştirilebilir. Üstelik bloklar arasındaki bağımlılık sayısının artırılması da Aktarıma Dayalı Saldırı'yı önleyememektedir. Örneğin, blok başına iki bağımlılığın kurulması durumda, I' imgesi içerisindeki üçlü blok (I'_B, I'_E, I'_C), I'' imgesi içerisindeki (I''_B, I''_F, I''_C) üçlü blok ile değiştirilebilir.

$$\cdots \leftrightarrow I'_A \leftrightarrow I'_B \leftrightarrow I'_E \leftrightarrow I'_C \leftrightarrow I'_D \leftrightarrow \cdots,$$
$$\cdots \leftrightarrow I''_A \leftrightarrow I''_B \leftrightarrow I''_F \leftrightarrow I''_C \leftrightarrow I''_D \leftrightarrow \cdots$$

Benzer saldırılar şekil 4.10.'da gösterilen blok başına 4 ve 8 bağımlılığın kullanıldığı durumlar için de geçerlidir.



Şekil 4.10. Blok bağımlılıkları **(a)** 8 bağımlılık **(b)** 4 bağımlılık **(c)** 2 bağımlılık **(d)** 1 bağımlılık

4.4. Saldırılara Karşı Alınan Önlemler

Bu bölüm içerisinde saldırılara karşı önlem almak amacıyla önerilen teknikler incelenmiştir. Bu önerilerin bir kısmı önerilen tekniklerin saldırılara karşı geliştirilmiş sürümleridir [40, 47-51]. Ancak bu geliştirilmiş sürümler ya güvenliği tam olarak sağlayamamaktadır ya da güvenliği sağlayarak değişim bölgesi belirleme yeteneklerini kaybetmektedirler. Çözüm olarak bağımlı blok yapısının kullanılması önerilmiştir [39, 42]. Ancak bağımlı blok yapısında, bir blok içerisindeki bir pikselin değerinin değiştirilmesi değiştirilen bloğun ve bu bloğa bağımlı tüm blokların doğrulanamamasına neden olmaktadır. Literatürde hem saldırılara karşı güvenilir hem de değişim bölgesi belirleyebilme özelliğine sahip diğer bir teknik ise Sıradüzensel Filigranlama Tekniği'dir [52]. Ancak Sıradüzensel Filigranlama Tekniğinde, imge çözünürlüğü sıradüzen içerisinde filigran bilgisini taşıyan en alt seviyeye kadar bölünebilirse değişim bölgesi doğru olarak belirlenebilmektedir. Saldırılara karşı önerilen çözümler bölüm 4.4.1 ile 4.4.7. arasında ayrıntılı olarak incelenmiştir.

4.4.1. Blok Boyutlarını Büyütme

VNTS kullanılarak oluşturulan sahte imgelerin aslına benzerliklerini etkileyen iki önemli etken vardır. Bunlar, kod çizelgesinde kullanılan imge bloklarının boyutu ve sayısıdır [39, 52]. Daha küçük boyutlu bloklar, büyük boyutlu bloklara göre daha kolay ve daha doğru olarak özgününe yaklaştırılabilir. Benzer şekilde kod çizelgesindeki blok sayısının artırılması, özgün filigranı taşıyan sahte imgelerin asıllarına daha benzer olmalarını sağlar. Filigranlama

işleminde kabul edilebilir bir sahtecilik olasılığı, blok boyutlarının büyütülmesiyle azaltılabilir. Daha büyük boyutlu blokların kullanılması kod çizelgesindeki blok sayısını azaltarak, özgün filigran taşıyan sahte imgenin aslına benzerlik oranını düşürür. Ancak bu önlem sahte bir imgenin oluşturulmasını tamamıyla önleyemez. Eğer saldırgan filigranlı imgelerin olduğu yeterli büyüklükteki bir kod çizelgesine sahip ise, kabul edilebilir sahte imgeleri yapması hala mümkündür. Ayrıca daha büyük boyutlu blok seçimi, filigranın değişim bölgesi belirleme doğruluğunu azaltacaktır.

4.4.2. Sayısal İmza İçerisine Blok İndisi Ekleme

Vektör nicemlemeli taklit saldırısını önlemek amacıyla Wong'un tekniğindeki sayısal imzanın oluşturulmasında blok indisinin de hesaplamaya katılması önerilmektedir [51]. Geliştirilmiş yeni sayısal imza oluşturma işlemi eşitlik 4.17., 4.18. ve 4.19. ile verilmiştir.

$$h_r = H\left(M, N, \tilde{I}_r, r\right) \tag{4.17}$$

$$W_r = b_r \oplus h_r \tag{4.18}$$

$$S_r = E\left(W_r, K_E\right) \tag{4.19}$$

Burada r, imge içerisindeki bloğun konumunu göstermektedir. Sayısal imza hesaplanırken blok indisinin kullanılması eşitlik sınıf sayısını artırır. Böylece, imge içerisinde aynı filigranı içeren farklı konumlardaki bloklar ayrı eşitlik sınıflarına ait olurlar. Her bir sınıf için kod çizelgesi aynı indisli ve aynı filigrana sahip bloklar ile sınırlandırıldığı için, VNTS'nin yapılması oldukça zorlaştırılmıştır. Ancak, filigranlı imgelere ait yeterli büyüklükteki bir kod çizelgesi ile bir saldırı gerçekleştirmek hala mümkündür [39].

4.4.3. Sayısal İmza İçerisine İmge İndisi Ekleme

Saldırıları önlemek için sayısal imza içerisine imge indislerinin eklenmesi önerilmiştir [41, 50, 51]. Bu öneri eşitlik sınıflarının sayısını artırır ve kod çizelgesi oluşturmayı sınırlar. Yeterince büyük indis değerlerinin kullanımı sahte imge oluşumunu pratik olarak engelleyebilir. Sayısal imza içerisine imge indisleri eşitlik 4.20., 4.21. ve 4.22. kullanılarak eklenebilir.

$$h_r = H\left(M, N, \tilde{I}_r, r, I_{ID}\right) \tag{4.20}$$

$$W_r = b_r \oplus h_r \tag{4.21}$$

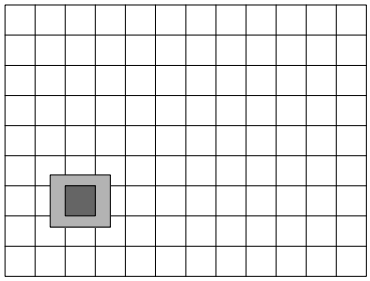
$$S_r = E(W_r, K_E) \tag{4.22}$$

Burada I_{ID}, imgeye ait indisi belirtmektedir. Dikkat edilirse böyle bir indisin kullanılması filigranın çıkarımı esnasında da gereklidir. Bir veritabanı içerisinde tüm imgeler için ayrı indislerin yönetilmesi bazı uygulamalar için olanaklı iken, uygulamaların çoğu için bu yapı büyük bir yük getirir. Bu kısıtlama düşünülerek imgeye ait indisin imge içerisinden çıkarılması önerilmiştir [40, 41]. Örneğin tüm imgenin özüt değerinden çıkartılan bir indisin kullanılması düşünülmüştür. Bu öneri indisin saklanması ve yönetilmesi açısından uygun bir çözüm sağlamasına rağmen, filigranın değişim bölgesi belirleme niteliğini yok edecektir. Çünkü imge içerisinde herhangi bir pikselin bir bitinin bile değişmesi ile özüt ve dolayısıyla imge indisi de değişecektir.

Alternatif bir yaklaşım olarak imge indislerinin imge içerisinde birden fazla bölgeye gömülmesi önerilmiştir [41]. Böylece imgeye ait indisin birden fazla bölgede olmasından faydalanılarak filigran çıkarımı esnasında indisin bulunma şansı artırılmıştır. Ancak bu teknik, tüm olası durumlar için indisin doğru olarak çıkarılacağını garanti etmez. Ayrıca indisin birden fazla bölgeye yerleştirilmesi, hem normal olarak filigran tarafından kullanılacak yerin işgal edilmesine hem de filigranın değişim bölgesi belirleme doğruluğunun azalmasına yol açmaktadır.

4.4.4. Bağımlı Blok Yapısının Kullanılması

VNTS'yi önlemek için bölüm 4.4.1. ve 4.4.2. içerisinde verilen çözümler saldırı yapılmasını engelleyemezler. Bu çözümler sadece saldırı yapılmasını zorlaştırırlar. Bölüm 4.4.3. içerisinde önerilen çözüm ise saldırı yapılmasını tamamıyla önleyebilir. Ancak bu çözümün filigranlı imge üzerinde yapılan değişimleri belirleyememesi önemli bir eksikliktir. VNTS'yi engellemek için alternatif bir yaklaşım bağımlı blok yapısının kullanılmasıdır [39]. Bu yapının kullanılmasıyla bir saldırganın filigranlı imge bloklarını kullanarak eşitlik sınıfları oluşturmasına müsaade edilmez ve dolayısıyla özgün bir filigran taşıyan sahte bir imge meydana getirmesi engellenir. Örneğin Wong'un tekniğinde her bir blok için hesaplanılan özütün, doğrulanacak olan bloğun piksellerinin yanı sıra bu bloğa komşu pikseller de kullanılarak hesaplanması önerilmiştir [49]. Bu teknikte küçük boyutlu bir blok içerisine bu bloğu kaplayan daha büyük bir bloğun özütü eklenilmektedir. Örneğin 8*8 piksellik bir blok içerisine bu bloğu kapsayan 16*16 piksellik bir bloğun özütü gömülmektedir. Bu blok yapısı şekil 4.11. içerisinde gösterilmiştir. Bu şekil içerisinde koyu olan alan filigran gömülecek bölgeyi, daha açık renkli alan özütü hesaplamak için kullanılacak bölgeyi göstermektedir.



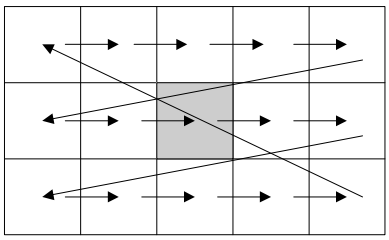
Şekil 4.11. Bloksal bağımlılığın kurulduğu kırılgan filigranlama tekniği

Bu teknik vektör nicemlemeli taklit saldırısını önlemesine rağmen, değişim bölgesini doğru olarak belirlemede belirsizliklere sahiptir [50, 52].

4.4.5. Hash Block Chaining Version 1

Barreto ve diğ [42], HBC1 *(Hash Block Chaining Version 1, Özüt Blok Zincirleme Sürüm 1)* olarak adlandırdıkları bloksal bağımlılığın kullanıldığı diğer bir kırılgan filigranlama tekniği önermişlerdir. Bu teknikte kullanılan bloklar arasındaki bağımlılık düzeni şekil 4.12. ile verilmiştir. HBC1 Filigranlama Tekniği'nde her bir blok özütü hem doğrulanacak olan bloğun içeriğine, hem de şekil 4.12.'de gösterildiği gibi kendisinden önceki bloğun içeriğine bağımlıdır. Her bir blok özütü eşitlik 4.23. kullanılarak hesaplanır.

$$h_r = H\left(M, N, \tilde{I}_r, \tilde{I}_{r-1}, r\right) \tag{4.23}$$



Şekil 4.12. HBC1 tekniğinde kullanılan bloksal bağımlılık

Şekil 4.12.'de blok başına bir bağımlılık kullanılmıştır. HBC1 Filigranlama Tekniği'ndeki bloklar arasında, şekil 4.10.'da gösterildiği gibi birden fazla bağımlılık kurulabilir. Birden fazla bağımlılığın kurulması durumunda, eşitlik 4.23. içerisine tüm bağımlı blokların içeriği dahil edilir. Ancak HBC1 Filigranlama Tekniği'nde mümkün olduğu kadar az blok bağımlılığının

kurulması istenilir. Blok bağımlılık sayısının artması, tekniğin değişim bölgesi doğrulama hassasiyetini azaltır. Örneğin blok başına bir bağımlılık kullanıldığında, bir bloğun bir pikselinin bozulması iki bloğun doğrulanamamasıyla sonuçlanırken, blok başına 2 bağımlılık kullanıldığında, bir bloğun bir pikselinin bozulması 3 bloğun birden doğrulanamamasına neden olur. Bu yüzden HBC1 tekniğinde blok başına bir bağımlılık kullanılmıştır. HBC1 ve bazı blok bağımlı tekniklerin bölüm 4.3.4. içerisinde verilen Aktarıma Dayalı Saldırı'ya karşı güvenilir olmadıkları bilinmektedir [42]. Bloksal bağımlılığa dayanan teknikleri Aktarıma Dayalı Saldırı'ya karşı güvenilir bir yapıya dönüştürmek için deterministik olmayan sayısal imzaların kullanılması önerilmiştir [42].

4.4.6. Hash Block Chaining Version 2

HBC2 *(Hash Block Chaining Version 2, Özüt Blok Zincirleme Sürüm 2)*, HBC1 Filigranlama Tekniği ile aynı bloksal bağımlılığı kullanır [42]. HBC2 Filigranlama Tekniği'nde blok imzalarının oluşturulmasında deterministik olmayan sayısal imzalar kullanılarak Aktarıma Dayalı Saldırı ve Geliştirilmiş Doğum Günü Saldırıları engellenir. Deterministik olmayan Sayısal İmzalar hem özüt fonksiyonuna hem de rasgele seçilen bir parametreye bağımlıdırlar. DSA *(Digital Signature Algorithm)* ve Schnorr sayısal imzaları deterministik olmayan imzalardır [1, 2]. Deterministik bir sayısal imza kullanıldığında aynı iki imgeye ait imzalar birbirinden farklıdır. Bu özellik HBC2 Filigranlama Tekniğini Aktarıma Dayalı ve Geliştirilmiş Doğum Günü Saldırıları'na karşı güvenilir yapmaktadır. HBC2 Filigranlama Tekniği'nde her bir blok özütü eşitlik 4.24. kullanılarak hesaplanır.

$$h_r = H\left(M, N, \tilde{I}_r, \tilde{I}_{r-1}, r, S_{r-1}\right) \tag{4.24}$$

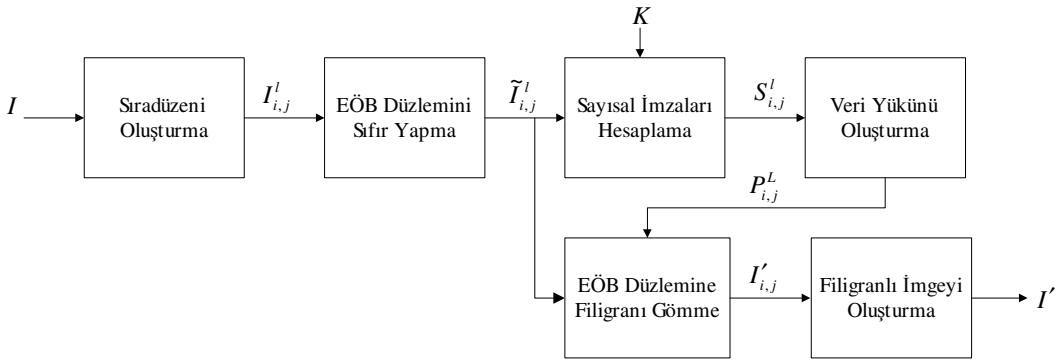
Burada S_{r-1}, $\tilde{I}_{r-1}$ bloğunun deterministik olmayan sayısal imzasıdır. Eşitlik 4.24.'de verildiği gibi HBC2 Filigranlama Tekniği'ndeki her bir blok özütü; doğrulanacak olan blok içeriği, komşu bloğun içeriği ve komşu bloğunun deterministik olmayan sayısal imzası kullanılarak hesaplanır.

4.4.7. Sıradüzensel Filigranlama Tekniği

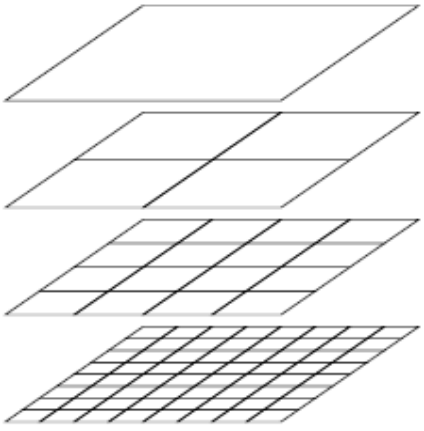
Çelik ve diğ. [52], imgelerin aslıyla aynılığını doğrulamak için sıradüzensel blok yapısına dayanan bir kırılgan filigranlama tekniği önermişlerdir. Sıradüzensel Filigranlama Tekniği hem saldırılara karşı güvenilir bir yapıya hem de değişim bölgesi belirleme özelliğine sahiptir. Bu teknikte filigranın imgeler içerisine gömülmesi ve çıkarılması sıradüzensel bir yapı

kullanılarak gerçekleştirilir. İmgeyi üst üste gelmeyen bloklara bölme işlemi çok seviyeli bir sıradüzenin oluşmasını sağlar. Sıradüzen içerisindeki her bir seviye bir önceki seviyedeki ayrık blokların birleştirilmesiyle oluşturulur. Sıradüzenin en üst seviyesindeki bloğa (imgenin kendisi) ait sayısal imza saldırılara karşı tam bir güvenlik sağlarken, en alt seviyedeki bloklara ait sayısal imzalar değişim bölgesinin belirlenmesinde kullanılırlar. Sıradüzensel Filigranlama Tekniği'nin filigranlama yordamı, şekil 4.13.'de gösterilmiştir.

Sıradüzensel Filigranlama Tekniği'nde filigranlanmak istenen $M * N$ boyutlu bir I imgesi ilk olarak sıradüzensel bir yapı içerisinde bloklara bölünür. Sıradüzenin en alt seviyesindeki üst üste gelemeyen her bir blok $I_{i,j}^{l}$ ile temsil edilir. Burada i, j, bloğun uzaysal konumunu ve l, bloğun ait olduğu sıradüzensel seviyeyi göstermektedir.



Şekil 4.13. Sıradüzensel Filigranlama Tekniğinin filigranlama yordamı



Şekil 4.14. Sıradüzensel 4 seviyeli blok yapısı

Sıradüzen içerisindeki toplam seviye ise L ile tanımlanmaktadır. Sıradüzen içerisindeki ardışık seviyelerin her biri, bir önceki seviyedeki $2x2$ ayrık bloğun birleştirilmesiyle oluşturulur. Sıradüzensel seviyelerdeki bloklar eşitlik 4.25. kullanılarak oluşturulur.

$$\texttt{for } l = L-1:2$$

$$\begin{bmatrix} I_{2i,2j}^{l+1} & \| & I_{2i,2j+1}^{l+1} \\ I_{2i+1,2j}^{l+1} & \| & I_{2i+1,2j+1}^{l+1} \end{bmatrix} = I_{i,j}^{l} \tag{4.25}$$

Sıradüzenin en üst seviyesinde sadece $I_{0,0}^{1} = I$ bloğu bulunmaktadır. 4 seviyeli bir sıradüzen yapısı şekil 4.14.'de gösterilmiştir. Bu teknikte imgenin filigranlanması için her bir seviyedeki bloklara ait sayısal imzalarının hesaplanması gerekir. Bloklara ait sayısal imzalar hesaplanmadan önce Wong'un tekniğinde olduğu gibi her bir bloğun EÖB düzlemi sıfıra eşitlenir. EÖB düzlemi sıfıra eşitlenmiş olan bloklar $\tilde{I}_{i,j}^{l}$ ile gösterilmektedir. Blok imzaları eşitlik 4.26. ve 4.27. kullanılarak hesaplanır.

$$\texttt{for } l = 1:L$$

$$h_{i,j}^{l} = H\left(\tilde{I}_{i,j}^{l} \,\|\, [top]\right) \tag{4.26}$$

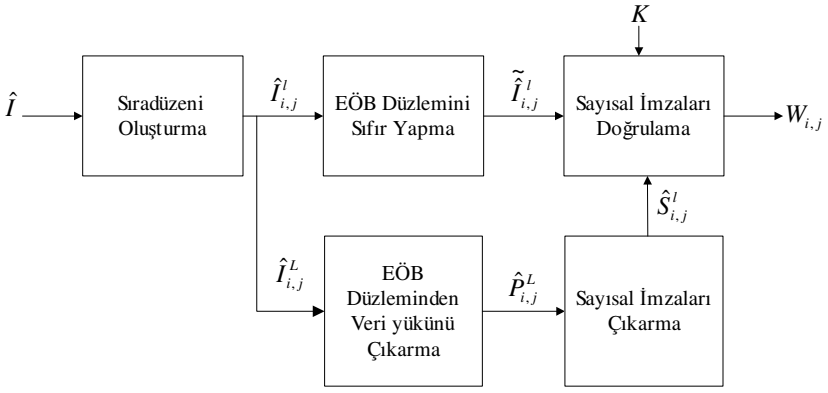
$$S_{i,j}^{l} = E\left(h_{i,j}^{l}, K_E\right) \tag{4.27}$$

Eşitlik 4.26. içerisindeki "top" değeri sadece $l = 1$ olduğu zaman kullanılır. Sıradüzenin farklı seviyelerindeki bloklara ait sayısal imzalar aynı EÖB düzlemine gömüleceğinden dolayı daha yüksek seviyelerdeki bloklara ait sayısal imzalar daha düşük seviyedeki bloklara paylaştırılır. Sıradüzenin en alt seviyesindeki bloklar, hem kendi bloklarına ait sayısal imzaları hem de üst seviyelerdeki blokların sayısal imzasını taşırlar. Her bir bloğun taşıdığı veri yükü yaklaşık olarak eşitlik 4.28.'deki gibi hesaplanır.

$$P \cong \frac{4|S|}{3} \tag{4.28}$$

Burada $|S|$ sayısal imzanın uzunluğunu göstermektedir. Sıradüzensel yapının en altındaki blok büyüklüğü veri yüküne eşit veya daha büyük olmalıdır.

Sıradüzensel Filigranlama Tekniği'nin doğrulama yordamı şekil 4.15.'de verilmiştir. Doğrulama yordamında ilk olarak doğrulanacak olan imge ($\hat{I}$), sıradüzensel yapı içerisinde alt bloklara bölünür. Her bir $\hat{I}_{i,j}^{l}$ alt bloğu içerisinden veri yükleri ($\hat{P}_{i,j}^{L}$) çıkartılarak her bir seviyedeki bloklara ait sayısal imzalar ($\hat{S}_{i,j}^{l}$) oluşturulur. Daha sonra her bir sayısal imza çözülerek elde edilen özüt değeri, bloğun yeniden hesaplanılan özüt değeri ile karşılaştırılır. Özüt değerlerinin eşleşmesi bloğun aslıyla aynılığını doğrularken, eşleşmeyen blok özüt değerleri bloğun değiştirildiğini gösterir.



Şekil 4.15. Sıradüzensel Filigranlama Tekniğinin doğrulama yordamı

5. SIRADÜZENSEL AĞAÇ YAPILI KIRILGAN FİLİGRANLAMA TEKNİĞİ

İmge üzerinde yapılan değişimleri belirleyebilmek için Kırılgan Filigranlama Teknikleri bloksal yapıda tasarlanırlar. Değişim bölgesinin belirlenmesinde önemli olan, değiştirilen bölgenin yüksek doğrulukta bir çözünürlükte tespit edilebilmesidir. Bloksal yapıda tasarlanan kırılgan filigranlama tekniklerinde, değişim bölgesi doğrulama çözünürlüğü seçilen blok büyüklüğüne göre belirlenmektedir. Daha büyük blokların seçilmesi değişim yapılan bölgenin tespit edilmesindeki doğrulama hassasiyetini azaltacaktır. Blok büyüklüğünün seçimi ise her bir blok içerisine gömülecek veri yüküne ve bloktaki her bir pikselin kaç bitinin kullanılacağına bağımlıdır. Kırılgan Filigranlama Tekniklerinde gömü bozulmasını minimuma indirmek için genellikle imgenin EÖB düzlemi kullanılır [37, 38, 40, 42-44, 51, 52]. Dolayısıyla her bir piksel içerisine sadece bir bitlik bilgi gömülmektedir. Sayısal İmzaların kullanıldığı Kırılgan Filigranlama Teknikleri'nde veri yükü, kullanılan sayısal imzanın uzunluğuna bağımlı olarak hesaplanır [37, 38, 40, 42, 43, 51, 52]. Bu yüzden Kırılgan Filigranlama Teknikleri'nde seçilen sayısal imzanın uzunluğu veri yükünü ve dolayısıyla seçilen blok büyüklüğünü belirlemektedir. Ayrıca seçilen sayısal imza algoritması, kırılgan filigranlama tekniğinin kriptografik güvenlik seviyesini de belirlemektedir.

Bu tez çalışmasında imgelerin aslıyla aynılığını kanıtlamak için iki yeni kırılgan filigranlama tekniği önerilmiştir. Önerilen ilk teknik Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniği olarak adlandırılmıştır. Bu teknik, Sıradüzensel Filigranlama Tekniği temel alınarak geliştirilmiştir. Sıradüzensel Filigranlama Tekniği'nin değişim bölgesi belirleme doğruluğu, imgenin sıradüzen içerisinde alt seviyelere bölünebilmesine bağımlıdır. Eğer imge sıradüzenin üst seviyelerinde bir alt seviyeye bölünemezse, Sıradüzensel Filigranlama Tekniğinin değişim bölgesi belirleme doğruluğu oldukça azalmaktadır. Örneğin 1200*1600 piksellik uzaysal çözünürlüğü sahip bir imge, sıradüzen içerisinde 5. seviyeye kadar bölünebilir. Bu seviyedeki her bir blok büyüklüğü 75*100 pikselden oluşur. Dolayısıyla bir bitlik bir bozulma 7500 pikselin doğrulanamamasına neden olur. Yine 767*1025 piksellik uzaysal çözünürlüğe sahip olan bir imge sıradüzenin en üst seviyesinde bir alt seviyeye bölünemediğinden imgenin Sıradüzensel Filigranlama Tekniği ile filigranlanması sadece en üst seviyede yapılabilir. Ancak en üst seviyede yapılan filigranlamada Sıradüzensel Filigranlama Tekniğinin değişim bölgesi belirleme niteliği yok olmaktadır. Dolayısıyla Sıradüzensel Filigranlama Tekniği, imge çözünürlüğüne bağımlı olarak yüksek doğrulukta bir değişim bölgesi belirleme niteliğine sahiptir. Ayrıca Sıradüzensel Filigranlama Tekniği'nde daha uzun sayısal imzaların kullanılması değişim bölgesi belirleme doğruluğunu azaltmaktadır. Örneğin

Sıradüzensel Filigranlama Tekniğinde blok doğrulaması için 64 bitlik MAC'in *(Message Authentication Code)* kullanılması durumunda, yaklaşık olarak 86 bit barındırabilecek bir blok büyüklüğüne ihtiyaç duyulurken, 64 bitlik MAC yerine daha güvenilir 320 bitlik DSA *(Digital Signature Algorithm)* sayısal imzasının kullanılmasıyla yaklaşık olarak 427 bit barındırabilen bir blok büyüklüğüne ihtiyaç duyulmaktadır. Sıradüzensel Filigranlama Tekniğinde DSA sayısal imzasının kullanılması durumunda, filigranlanmış imgenin güvenlik seviyesi artırılırken değişim bölgesi belirleme doğruluğu azalmaktadır. Bu durum sayısal imzaları kullanan diğer kırılgan filigranlama teknikleri içinde geçerlidir [37, 38, 42, 43, 51, 52].

Önerilen teknikte ise imgenin bloklara bölünmesinde sıradüzensel yapı yerine sıradüzensel ağaç yapısının kullanılması önerilmiştir. Sıradüzensel ağaç yapısında tavan ve taban fonksiyonları kullanılarak imgenin sıradüzen içerisinde sürekli olarak alt seviyelere bölünmesi garanti altına alınmıştır. İmge çözünürlüğünden bağımsız olarak imgenin istenilen alt seviyeye kadar bölünebilmesi, önerilen tekniğin tüm imge çözünürlüklerinde yüksek doğrulukta değişim bölgesi belirleme niteliğine sahip olmasını sağlamıştır. Ayrıca önerilen teknikte daha yüksek kriptografik güvenlik seviyesine ve daha uzun bit uzunluğuna sahip bir sayısal imza algoritmasının en alttaki blok büyüklüğünü minimum seviyede etkilemesi için iki ayrı sayısal imza algoritmasının birlikte kullanılması önerilmiştir. Bu öneride alt seviyedeki blok imzalarının hesaplanmasında daha kısa bit çıkışına sahip sayısal imza algoritmasının kullanılması, daha üst seviyelerde ise daha güvenilir ve daha uzun bit çıkışına sahip sayısal imza algoritmasının kullanılması öngörülmüştür. Seviyeler içerisinde hangi sayısal imzanın kullanılacağı ise geliştirilen bir algoritma ile belirlenmiştir. Bu algoritma ile önerilen tekniğin kriptografik güvenlik seviyesi artırılırken, değişim bölgesi belirleme doğruluğunun azalmaması temin edilmiştir.

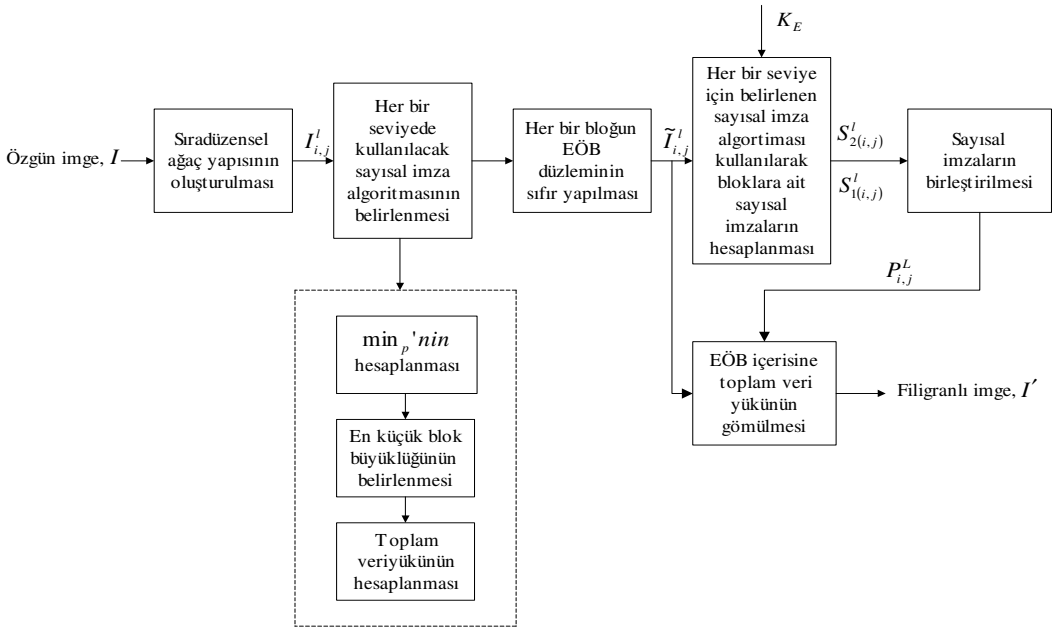
Önerilen teknik, Sıradüzensel Filigranlama Tekniğine göre üç önemli avantaj sağlamaktadır: 1) Önerilen teknik, imgelerin uzaysal çözünürlüğünden bağımsız olarak filigranlama işlemini gerçekleştirdiğinden dolayı tüm imge çözünürlüklerinde çalışmaktadır. 2) Önerilen sıradüzensel ağaç yapısıyla imgelerin sürekli olarak alt seviyelere bölünmesi garanti altına alınarak, önerilen tekniğin daima yüksek doğrulukta değişim bölgesi belirleme niteliğine sahip olması sağlanmıştır. 3) Değişim bölgesi belirleme doğruluğunu etkilemeden, daha yüksek kriptografik güvenlik seviyesine sahip Sayısal İmzaların kullanılması sağlanmıştır.

Önerilen Sıradüzensel Ağaç yapılı Kırılgan Filigranlama Tekniği'nin hem yüksek bir güvenlik seviyesine hem de yüksek doğrulukta bir değişim bölgesi belirleme niteliğine sahip olduğu yapılan deneysel çalışmaların sonuçlarıyla belirlenmiştir.

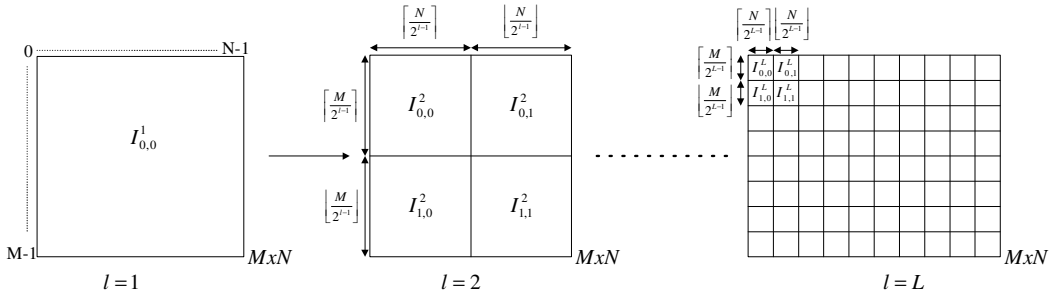
5.1. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniğinin Filigranlama Yordamı

Önerilen teknikte imgenin filigranlanması 4 aşamada gerçekleştirilmiştir. İlk aşama, imgenin sıradüzensel ağaç yapısı kullanılarak üst üste gelmeyen bloklara bölünmesi işleminden oluşmaktadır. İkinci aşamada her bir seviyede kullanılacak sayısal imzanın belirlenmesi yapılır. Üçüncü aşamada, her bir seviye için seçilen sayısal imza kullanılarak blok imzaları hesaplanır. Son aşamada ise her bir seviyedeki sayısal imzalar birleştirilerek sıradüzenin en alt seviyesindeki blokların EÖB düzlemi içerisine gömülür. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniğinin filigranlama yordamı şekil 5.1. içerisinde gösterilmiştir.

Önerilen kırılgan filigranlama tekniği ilk olarak sıradüzensel ağaç yapısını kullanarak imgeyi çok seviyeli bir sıradüzen içerisinde üst üste gelmeyen bloklara böler. Bölme işlemine minimum veri yükünü taşıyacak en küçük blok büyüklüğüne varılana kadar devam edilir. İmgenin sıradüzensel ağaç yapısı kullanılarak bloklara bölünmesi şekil 5.2.'de gösterilmiştir. Sıradüzensel yapıda olduğu gibi sıradüzensel ağaç yapısında da en üst seviyedeki blok tüm imge piksellerinden oluşurken, alt seviyelerdeki her bir blok bir alt seviyedeki 2x2 ayrık bloğun birleşmesinden oluşur. Ancak sıradüzensel yapıdan farklı olarak, sıradüzensel ağaç yapısında her bir seviyenin bir alt seviyeye bölünmesinde taban $(\lceil \cdot \rceil)$ ve tavan $(\lfloor \cdot \rfloor)$ fonksiyonları kullanılır. Bu fonksiyonlar iki önemli fayda sağlamaktadır. Birincisi, önerilen tekniğin tüm imge çözünürlüklerinde çalışmasını garanti eder.



Şekil 5.1. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniği'nin filigranlama yordamı



Şekil 5.2. Sıradüzensel ağaç yapısı kullanılarak imgenin bloklara bölünmesi

İkincisi ise, sıradüzenin her bir seviyesindeki blokların bir alt seviyedeki bloklara bölünmesini garanti ederek, veri yükünü taşıyacak minimum boyutlu blokların oluşmasını sağlar. Sıradüzensel ağaç yapısı kullanılarak daha küçük boyutlu blokların seçilebilmesi, önerilen tekniğin değişim bölgesi doğrulama hassasiyetini önemli miktarda artırmaktadır. *MxN* boyutlu bir I imgesinin sıradüzensel ağaç yapısı kullanılarak bölünmesiyle oluşan her bir $I_{i,j}^{l}$ bloğu eşitlik 5.1. ile verilmiştir.

$$\texttt{for}\ \ l = 1 : L$$
$$I_{i,j}^{l} = \left\{ I_{i,j}^{l+1}, I_{i,j+1}^{l+1}, I_{i+1,j}^{l+1}, I_{i+1,j+1}^{l+1} \right\} \tag{5.1}$$

Burada L sıradüzensel ağaç yapısındaki toplam seviyeyi; l her bir alt seviyeyi; (i, j) bloğun uzaysal pozisyonunu göstermektedir. Ayrıca en üst seviyede $I_{0,0}^{1} = I$ 'dır. Alt seviyedeki her bir blok büyüklüğü ise eşitlik 5.2. ile hesaplanır.

$$\texttt{for}\ \ l = 2 : L$$
$$\begin{aligned} I_{i,j}^{l} &= \left\lceil \frac{M}{2^{l-1}} \right\rceil x \left\lceil \frac{N}{2^{l-1}} \right\rceil \\ I_{i,j+1}^{l} &= \left\lceil \frac{M}{2^{l-1}} \right\rceil x \left\lfloor \frac{N}{2^{l-1}} \right\rfloor \\ I_{i+1,j}^{l} &= \left\lfloor \frac{M}{2^{l-1}} \right\rfloor x \left\lceil \frac{N}{2^{l-1}} \right\rceil \\ I_{i+1,j+1}^{l} &= \left\lfloor \frac{M}{2^{l-1}} \right\rfloor x \left\lfloor \frac{N}{2^{l-1}} \right\rfloor \end{aligned} \tag{5.2}$$

Önerilen filigranlama tekniğinde Sıradüzensel Filigranlama Tekniğinden farklı olarak, iki ayrı sayısal imza algoritmasının kullanılmıştır. Bu işlem daha uzun bit çıkışına ve daha yüksek kriptografik güvenlik seviyesine sahip sayısal imzaların kullanılması durumunda, önerilen tekniğin değişim bölgesi belirleme niteliğini yüksek doğrulukta devam ettirmesini sağlamak amacıyla yapılmıştır.

Sıradüzensel ağaç yapısındaki seviyeler içerisinde hangi sayısal imza algoritmasının kullanılacağı üç aşamada belirlenmektedir: İlk aşamada sıradüzensel ağaç yapısının en alt seviyesindeki her bir bloğun taşıyacağı minimum veri yükü hesaplanır. Minimum veri yükü sayısal imza algoritmalarının bit uzunluğuna bağımlı olarak eşitlik 5.3.'deki gibi hesaplanır.

$$\min_p = \left\lceil \frac{|S_2|}{2^{2(L-1)}} \right\rceil + \sum_{l=0}^{L-2} \left\lceil \frac{|S_1|}{2^{2l}} \right\rceil \tag{5.3}$$

Burada $|\cdot|$ sayısal imzanın uzunluğunu; $\min_p$, minimum veri yükünü tanımlamaktadır. Reel sayılı bir veri yükü olamayacağı için oluşacak ondalıklı değer, kendisinden büyük en küçük tam sayıya yuvarlanmıştır.

İkinci aşamada, sıradüzensel ağaç yapısının en alt seviyesinde minimum veri yükünü taşıyacak en küçük blok büyüklüğü belirlenir.

Üçüncü aşamada ise sıradüzensel ağaç yapısının en alt seviyesindeki bloklarda S_1 sayısal imzası, geriye kalan tüm üst seviyelerdeki bloklarda S_2 sayısal imzası kullanılarak toplam veri yükü eşitlik 5.4. ile hesaplanır.

$$P_{i,j}^{L} = |S_1| + \sum_{l=1}^{L-1} \left\lceil \frac{|S|}{2^{2l}} \right\rceil \tag{5.4}$$

Burada $P_{i,j}^{L}$, en alt seviyede her bir bloğun taşıyacağı toplam veri yükünü; $|S|$, sıradüzensel ağaç yapısındaki seviyelerdeki kullanıma göre S_1 veya S_2 sayısal imzasına ait uzunluğu tanımlamaktadır. Eğer hesaplanılan veri yükü, ikinci aşamada seçilen blok büyüklüğü tarafından taşınamazsa, en alt iki seviyedeki bloklarda S_1 sayısal imzası, geriye kalan üst seviyelerdeki tüm bloklarda S_2 sayısal imzası kullanılarak veri yükü yeniden hesaplanır. Bu işlem en alt seviyedeki blok, toplam veri yükünü taşıyana kadar devam ettirilir.

Sıradüzensel ağaç yapısı kullanılarak bloklar oluşturulduktan ve her bir seviyede kullanılacak sayısal imza algoritmaları belirlendikten sonra, her bir $I_{i,j}^{l}$ bloğundaki piksellerin

EÖB düzlemi sıfır yapılarak $\tilde{I}_{i,j}^{l}$ blokları oluşturulur. Sıradüzensel ağaç yapısındaki bloklara ait sayısal imzalar eşitlik 5.5., 5.6. ve 5.7. kullanılarak hesaplanır.

for $l = 1:L$

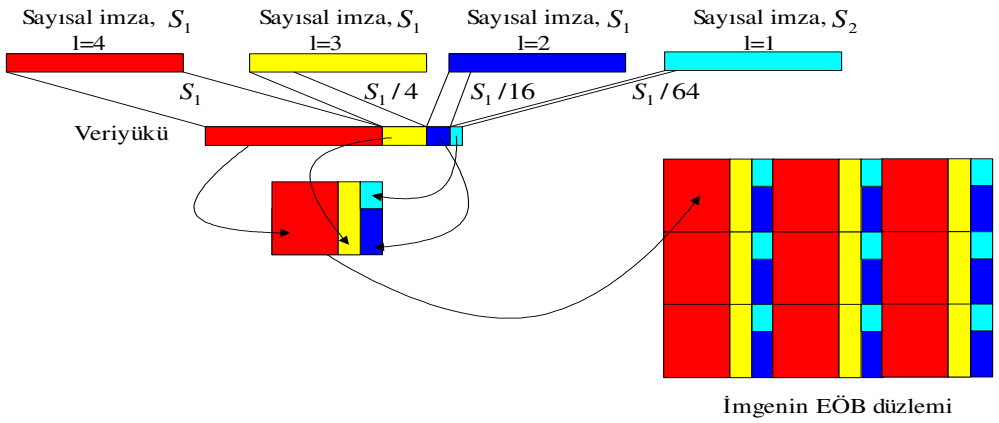
$$h_{i,j}^{l} = H\left(\tilde{I}_{i,j}^{l}\right) \tag{5.5}$$

$$S_{1(i,j)}^{l} = E\left(h_{i,j}^{l}, K\right) \tag{5.6}$$

$$S_{2(i,j)}^{l} = E\left(h_{i,j}^{l}, K\right) \tag{5.7}$$

Eşitlik 5.5.'de her bir bloğa ait özüt değerinin hesaplanmasında tüm blok pikselleri kullanılmaktadır. Sıradüzensel ağaç yapısının her bir seviyesindeki bloklara ait sayısal imzalar imgenin EÖB düzlemi içerisine gömülür. Ancak farklı seviyelerdeki bloklara ait sayısal imzalar aynı bit düzlemine gömüleceği için, sayısal imzaların üst üste gelmesini önleyen bir algoritmaya ihtiyaç duyulmaktadır. Sıradüzensel Filigranlama Tekniğinde, üst seviyelere ait sayısal imzaların daha alt seviyedeki bloklar içerisine bölüştürülmesi ve en alt seviyede biriken veri yükünün sıradüzenin en alt seviyesindeki blokların EÖB düzlemine gömülmesi önerilmiştir [52]. Dolayısıyla en alt seviyedeki her bir blok, hem bloğun kendisine ait sayısal imzayı hem de üst seviyelerdeki sayısal imzaların bir parçasını taşımaktadır. Bu yaklaşım önerilen kırılgan filigranlama tekniği içerisinde de kullanılmıştır. Ancak, önerilen filigranlama tekniğinde Sıradüzensel Filigranlama Tekniği'nden farklı olarak iki ayrı sayısal imza kullanıldığı için, en alt seviyede biriken veri yükü daha azdır. Örneğin 4 seviyeli bir sıradüzensel ağaç yapısı içerisinde, en alt üç seviyedeki blok imzalarının S_1 sayısal imza algoritmasıyla ve en üst seviyedeki blok imzasının S_2 sayısal imza algoritmasıyla hesaplanması durumunda, en alt seviyedeki bloğun taşıyacağı veri yükü $S_1 + \left\lceil \frac{|S_1|}{4} \right\rceil + \left\lceil \frac{|S_1|}{16} \right\rceil + \left\lceil \frac{|S_2|}{64} \right\rceil$ bit uzunluğundan oluşmaktadır. Bu oluşumda en alt seviyedeki her bir blok, tüm imgeye ait sayısal imzanın $(1/64)$'ünü, üçüncü seviyedeki bir bloğa ait sayısal imzanın $(1/16)$'sını, ikinci seviyedeki bir bloğa ait sayısal imzanın $(1/4)$'ünü, en alt seviyedeki bloğa ait sayısal imzanın tümünü taşımaktadır. Dört seviyeden oluşan sıradüzensel ağaç yapısının en alt seviyesindeki her bir bloğun taşıdığı toplam veri yükü şekil 5.3.'de gösterilmiştir.

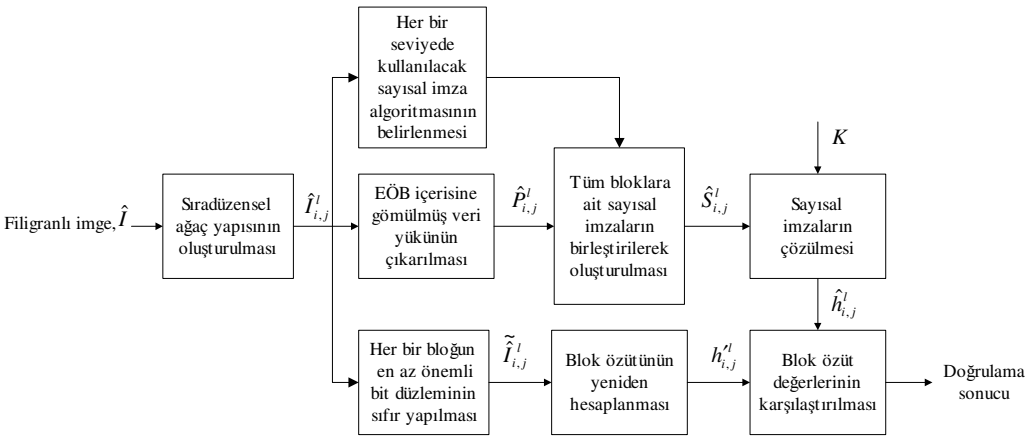
Önerilen kırılgan filigranlama tekniğinde imgenin filigranlanması, sıradüzensel ağaç yapısının en alt seviyesindeki her bir blok içerisine toplam veri yükünün gömülmesiyle tamamlanır.



Şekil 5.3. 4 seviyeli sıradüzensel ağaç yapısında sayısal imzaların birleştirilmesiyle oluşturulan veri yükü

5.2. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniğinin Doğrulama Yordamı

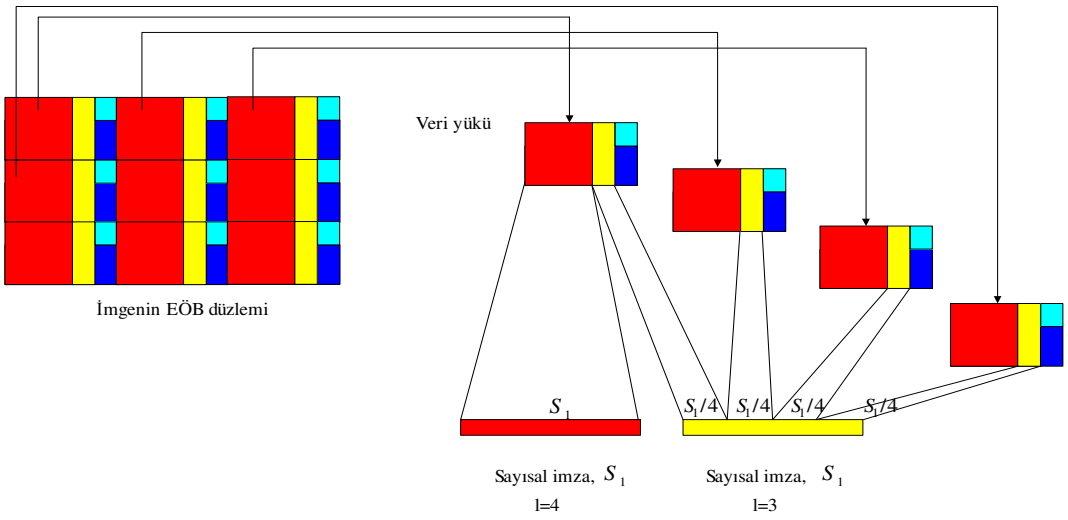
Önerilen kırılgan filigranlama tekniğinin doğrulama yordamı üç temel adımdan oluşmaktadır. İlk olarak filigranlama yordamında olduğu gibi imgenin sıradüzensel ağaç yapısı kullanılarak alt bloklara bölünmesi yapılır. İkinci adımda her bir seviyede kullanılan sayısal imza algoritması belirlenir ve en alt seviyedeki her bir bloğun EÖB düzlemine gömülmüş veri yükü çıkartılarak sayısal imzalar oluşturulur. Son adımda ise bloklara ait sayısal imzalar çözülerek blok özüt değerleri elde edilir ve bu değerler yeniden hesaplanan blok özüt değerleri ile karşılaştırılır. Özüt değerlerinin eşleşmesi bloğun aslıyla aynılığını doğrularken, eşleşmeyen blok özüt değerleri bloğun değiştirildiğini gösterir. Önerilen kırılgan filigranlama tekniğine ait doğrulama yordamı şekil 5.4.'de verilmiştir.



Şekil 5.4. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniği'nin doğrulama yordamı

Muhtemelen bozulmuş filigranlı imge ($\hat{I}$) içerisinden her bir seviyedeki bloklara ait sayısal imzaları çıkarabilmek için, en alt seviyedeki her bir blok içerisine dağılmış sayısal imza parçalarını birleştirmek gerekir. Ancak parçaların birleştirilebilmesi için, sıradüzensel ağaç yapısının seviyelerinde hangi sayısal imzanın kullanıldığının bilinmesi gerekir. Bu yüzden ilk olarak filigranlama yordamında olduğu gibi sıradüzensel ağaç yapısındaki seviyelerde hangi sayısal imzanın kullanıldığı belirlenir. Daha sonra sıradüzensel ağaç yapısının en alt seviyesindeki her bir blok içerisine gömülmüş veri yükü çıkarılır. Çıkarılan veri yükü, en alt seviyedeki bloğa ait sayısal imzanın tümünü ve seviyeye bağımlı olarak üst seviyelerdeki bloklara ait sayısal imzaların bir kısmını taşımaktadır.

Sıradüzenin üst seviyelerindeki bloklara ait sayısal imzalar, en alt seviyelerdeki bloklar üzerine dağılmış sayısal imza parçaları birleştirilerek oluşturulur. Örneğin 4 seviyeli bir sıradüzensel ağaç yapısı içerisinde, en alt üç seviyedeki blok imzalarının S_1 sayısal imza algoritmasıyla ve en üst seviyedeki blok imzasının S_2 sayısal imza algoritmasıyla hesaplandığını farz edelim. Bu durumunda, en alt seviyedeki blok yükü $S_1 + \left\lceil \frac{|S_1|}{4} \right\rceil + \left\lceil \frac{|S_1|}{16} \right\rceil + \left\lceil \frac{|S_2|}{64} \right\rceil$ bit uzunluğundan oluşur. Böylece sıradüzenin ikinci seviyesindeki bir bloğa ait sayısal imzayı oluşturmak için, sıradüzenin en alt seviyesindeki 4 blok içerisindeki sayısal imzaların birleştirilmesi gerekir. Benzer şekilde sıradüzenin üçüncü seviyesindeki bir bloğa ait sayısal imzayı oluşturmak için sıradüzenin en alt seviyesindeki 16 blok içerisindeki sayısal imzaların birleştirilmesi gerekir. Sıradüzenin her bir seviyesindeki bloklara ait sayısal imzaların birleştirilmesi şekil 5.5. ile verilmiştir.



Şekil 5.5. Sıradüzensel ağaç yapısındaki 4. ve 3. seviyedeki blok imzalarının birleştirilerek oluşturulması

Sıradüzensel ağaç yapısının her bir seviyesindeki bloklara ait sayısal imzalar birleştirildikten sonra, bu imzalar eşitlik 5.8. kullanılarak çözülür.

$$\texttt{for}\ \ l = 1:L$$
$$\hat{h}_{i,j}^{l} = D\left(\hat{S}_{i,j}^{l}, K\right) \tag{5.8}$$

Burada $\hat{h}_{i,j}^{l}$, filigranlı imge içerisinden çıkartılan sayısal imzanın çözülmesiyle elde edilen özüt değerini göstermektedir. $\hat{S}_{i,j}^{l}$ ise her bir seviyede kullanılan sayısal imzaya bağlı olarak $\hat{S}_1$ veya $\hat{S}_2$ sayısal imzalarını tanımlamaktadır.

Sıradüzensel ağaç yapısındaki her bir bloğun aslıyla aynılığını doğrulamak için, eşitlik 5.8. ile hesaplanan özüt değeri ile her bir blok için yeniden hesaplanacak özüt değerinin karşılaştırılması gerekir. Her bir blok özüt değerini yeniden hesaplamak için, $\hat{I}_{i,j}^{l}$ filigranlı imge bloklarının EÖB düzlemi sıfır yapılarak $\tilde{\hat{I}}_{i,j}^{l}$ blokları oluşturulur ve bu blokların özüt değeri eşitlik 5.9. kullanılarak hesaplanır.

$$\texttt{for}\ \ l = 1:L$$
$$h_{i,j}^{\prime l} = H\left(\tilde{\hat{I}}_{i,j}^{l}\right) \tag{5.9}$$

Yeniden hesaplanılan özüt değeri $h_{i,j}^{\prime l}$ ile $\hat{h}_{i,j}^{l}$ özüt değerinin eşleşmesi bloğun aslıyla aynılığını doğrularken, blokların eşleşmemesi blok içerisinde bir değişim olduğunu gösterir. Tüm imge blokları üzerinde bu doğrulama işlemi uygulanarak, imgenin aslıyla aynılığının doğrulanması yapılır.

5.3. Deneysel Çalışmalar

Önerilen tekniğin etkinliğini değerlendirebilmek amacıyla 5 farklı imge çözünürlüğünde (128*192, 240*320, 480*640, 768*1024, 1200*1600) ve 19 farklı test imgesi üzerinde deneysel çalışmalar yapılmıştır. Yapılan deneysel çalışmalarda sayısal imza algoritmaları olarak MD5 *(Message Digest 5)* algoritmasına dayanan 64 bitlik MAC ve SHA-1 *(Secure Hash Algorithm)* algoritmasına dayanan 320 bitlik DSA sayısal imzaları kullanılmıştır. Bu bölüm içerisinde gerek önerilen tekniğe gerekse karşılaştırma yapılan Sıradüzensel Filigranlama Tekniğine ait filigranlama ve doğrulama algoritmaları MATLAB ortamında yazılan programlar ile gerçekleştirilmiştir.

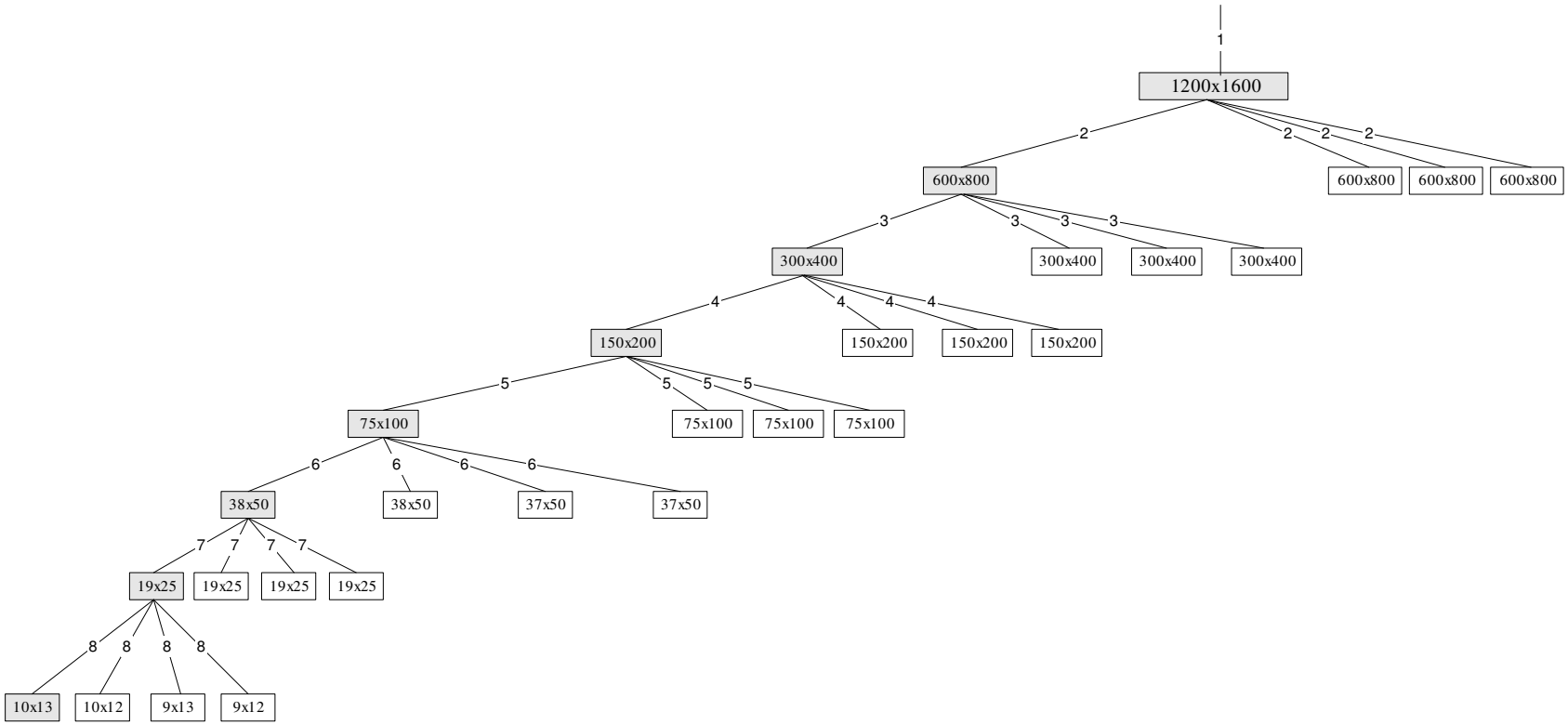
5.3.1. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniğinin Değişim Bölgesi Belirleme Doğruluğu

Önerilen teknik ile şekil 5.6.'da gösterilen 1200*1600 çözünürlüğe sahip test imgesinin filigranlanması ayrıntılı olarak aşağıda verilmiştir. İlk olarak imge sıradüzensel ağaç yapısı kullanılarak minimum veri yükünü taşıyacak en küçük blok büyüklüğüne kadar bölünür. Minimum veri yükü ($\min_p$) eşitlik 5.3. kullanılarak 89 bit olarak bulunur. İmgenin sıradüzensel ağaç yapısında eşitlik 5.1. ve 5.2. kullanılarak bölünmesi şekil 5.7.'de gösterilmiştir. Şekil 5.7.'de karmaşıklığı önlemek için her bir seviyedeki en soldaki düğüm göz önünde tutularak bölmeleme yapılmıştır. Seviyelerdeki diğer düğümler de benzer şekilde bloklara ayrılmaktadır. Şekil 5.7.'de görüldüğü gibi sıradüzensel ağaç yapısının en alt seviyesinde (8. seviye) $\min_p$'yi taşıyacak en küçük blok büyüklüğü 9*12 piksel boyutundadır. Bir alt seviyedeki bloklar $\min_p$'yi taşıyamayacağı için tekrar bir alt seviyeye bölmeleme yapılamaz.

Bir sonraki aşamada sıradüzensel ağaç yapısındaki her bir seviyede hangi sayısal imza algoritmasının kullanılacağı belirlenir. İlk olarak sıradüzenin en alt seviyesinde S_1 sayısal imzası (MAC), kalan tüm seviyelerde S_2 (DSA) sayısal imzası kullanılarak veri yükü eşitlik 5.4. ile hesaplanır. En alt seviyedeki her bir bloğun taşıyacağı veri yükü $P_{i,j}^L = 174$ bit olarak bulunur. Ancak bu veri yükünü en alt seviyedeki en küçük blok (9*12) taşıyamayacağı için, sıradüzensel ağaç yapısının en alt iki seviyesinde S_1 sayısal imzası ve kalan tüm seviyelerde S_2 sayısal imzası kullanılarak veri yükü tekrar hesaplanır ve $P_{i,j}^L = 110$ bit olarak bulunur.



Şekil 5.6. 1200*1600 piksellik özgün imge



Şekil 5.7. Sıradüzensel ağaç yapısı kullanılarak 1200x1600 piksellik bir imgenin bölünmesi

Oluşan bu yani veri yükü de en alt seviyedeki en küçük blok (9*12) tarafından taşınamayacağı için, en alt üç seviyede S_1, kalan tüm seviyelerde S_2 sayısal imzası kullanılarak veri yükü tekrar hesaplanır ve $P_{i,j}^{L} = 94$ bit olarak bulunur. Bu veri yükü sıradüzensel ağaç yapısının en alt seviyesindeki en küçük blok tarafından taşınabileceği için 8, 7 ve 6. seviyelerde S_1 sayısal imza algoritması kalan tüm seviyelerde (5, 4, 3, 2, 1) S_2 sayısal imza algoritması kullanılır.

Sıradüzensel ağaç yapısı oluşturulduktan ve her bir seviyede kullanılacak sayısal imza algoritması belirlendikten sonra, blokların EÖB düzlemleri sıfır yapılır ve eşitlik 5.5. kullanılarak aşağıdaki özütler hesaplanır. Özütlerin her biri şekil 5.7.'deki en sol taraftaki bloklara aittir. Şekil 5.7.'de bu bloklar koyu renk ile gösterilmiştir.

$h_{0,0}^{8} =$ 8fb4d0af962e66ec96ec2a22af5365c7

$h_{0,0}^{7} =$ 46145d7f58505ff5f2646497249859ee

$h_{0,0}^{6} =$ ddf58845352714b54efb1b11a001bef6

$h_{0,0}^{5} =$ 281da10ec942758d7cd4a3449cac9cf688812d40

$h_{0,0}^{4} =$ 3ff931f5839f027c9120b0457f0b3ae65ea99368

$h_{0,0}^{3} =$ e0fd2cf2cd324dd31578835b585fd72ff696ecbd

$h_{0,0}^{2} =$ 48251758c8fef497e01a8b6203554ac1f504a38b

$h_{0,0}^{1} =$ 81d9393aa1e5f528f8522b5d3f97deb503200c1d

Daha sonra eşitlik 5.6. ve 5.7. vasıtasıyla sayısal imzalar hesaplanır. Şekil 5.7.'de sıradüzensel ağaç yapısının 8. seviyesindeki 10x13 piksellik bloğun taşıyacağı veri yükünü oluşturan sayısal imzalar aşağıda verilmiştir.

$S_{1(0,0)}^{8} =$ eff660460199e606

$S_{1(0,0)}^{7} =$ 01ccb29b9e610abe

$S_{1(0,0)}^{6} =$ 72fe3fe553b00dbb

$S_{2(0,0)}^{5} =$ dd5a46351f7369ba9354ed625f6e4aa43ba2bb626ec6e8941e7e8afa300f3336a06c0b74f3e66bd8

$S_{2(0,0)}^{4} =$ 2bd6096b6ea9a88b41572662293b2b42076ad80f7734254ab0105e8e127d9b4d6a3d5b15ae4b22b3

$S_{2(0,0)}^{3} =$ 2d5f16e3a311bc59a36abc05df4435a35e40268e816f9cc3d13aca2296b6434b9bb3ea00b811ade7

$S_{2(0,0)}^{2} =$ fcc5bd6842815089ec5cacc73868f8647c027a04 c5306463121f03a6c24be2544117a7d35d5cfb23

$S_{2(0,0)}^{1} =$ 23f681636e053fa79706103aa173d81c33442a520344b8863e7df1a6eec66929a78d6133ae697c07

Sayısal imzaların her biri şekil 5.7.'deki en sol taraftaki bloklara aittir. Şekil 5.7.'de bu bloklar koyu renk ile gösterilmiştir. Her bir seviyedeki sayısal imzanın kaç bitinin 10x13 piksellik bloğun veri yükünde kullanılacağı sıradüzensel seviyeye bağımlı olarak eşitlik 5.4. ile hesaplanır. Bu hesaplamaya göre: $S^8_{1(0,0)}$'in 64 biti; $S^7_{1(0,0)}$'nin 16 biti; $S^6_{1(0,0)}$'nın 4 biti; $S^5_{2(0,0)}$'in 5 biti; $S^4_{2(0,0)}$'ün 2 biti; $S^3_{2(0,0)}$'ün 1 biti; $S^2_{2(0,0)}$'nin 1 biti; $S^1_{2(0,0)}$'in 1 biti birleştirilerek $P^8_{0,0}$ aşağıdaki gibi oluşturulur.

$P^8_{0,0} = 1110111111110110011000000100011000000000110011001111001100000011000000000111001100011111011000010$

Son olarak oluşturulan veri yükü 10*13 piksellik bloğun EÖB düzlemine gömülür. Yukarıdaki işlemler en alt seviyedeki tüm bloklar için uygulanarak filigranlı imge elde edilir. Filigranlı imge şekil 5.8.'de gösterilmiştir.

Önerilen tekniğin 1200*1600 piksellik imgelerin doğrulanmasındaki performansını değerlendirebilmek için, şekil 5.8.'de verilen filigranlı imge üzerinde değişimler yapılmıştır. Şekil 5.9. değiştirilmiş filigranlı imgeyi göstermektedir. Şekil 5.9. incelendiğinde, sahile bir "martı" eklenmiş, katamaran üzerindeki "7" rakamı ve "logo" silinmiştir. Değiştirilmiş bölgeler şekil 5.9.'da oklarla gösterilmiştir.



Şekil 5.8. Önerilen teknik ile filigranlanmış imge (çözünürlük=1200*1600 piksel)



Şekil 5.9. Değiştirilmiş filigranlı imge (çözünürlük=1200*1600 piksel)

Değiştirilmiş filigranlı imgeyi doğrulamak için önerilen tekniğin doğrulama yordamı çalıştırılır. Doğrulama yordamında ilk olarak imge filigranlama yordamında olduğu gibi sıradüzensel ağaç yapısı kullanılarak üst üste gelmeyen bloklara bölünür. Daha sonra her bir seviyede kullanılan sayısal imza algoritmaları belirlenir ve en alt seviyedeki her bir bloğun EÖB düzlemine gömülmüş veri yükü çıkartılarak her bir seviyedeki bloklara ait sayısal imzalar oluşturulur. Son olarak her bir seviyedeki bloklara ait sayısal imzalar eşitlik 5.8. kullanılarak çözülür ve elde edilen özüt değerleri eşitlik 5.9. kullanılarak yeniden hesaplanan blok özüt değerleri karşılaştırılır. Özüt değerlerinin eşleşmesi bloğu doğrularken, özüt değerlerinin eşleşmemesi bloğun değiştirildiğini gösterir.

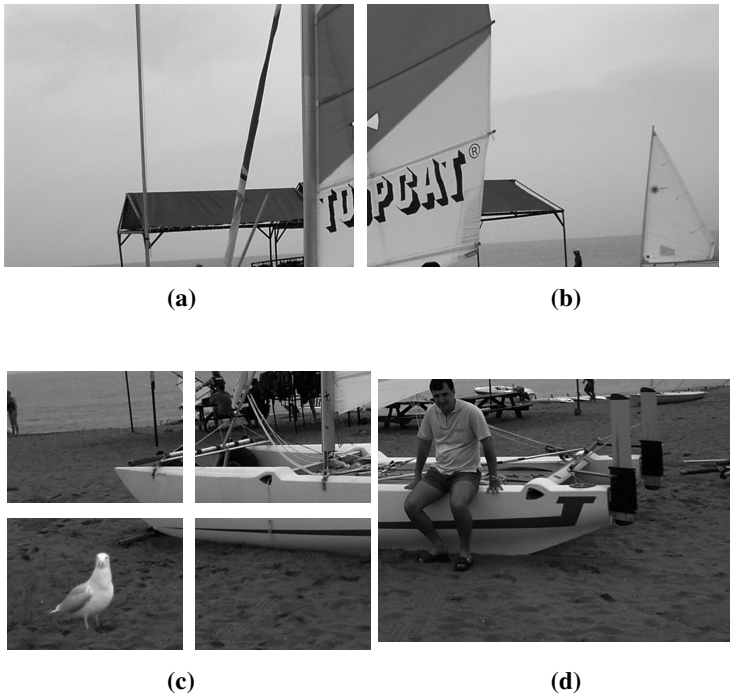
Blok özüt değerleri karşılaştırılırken ilk olarak tüm imgeye ait özüt değerleri ($\hat{h}^1_{0,0}$, $h'^1_{0,0}$) karşılaştırılır. Şekil 5.9.'daki filigranlı imgenin 1.seviyesindeki özüt değerleri aşağıda verilmiştir.

$$\hat{h}^1_{0,0} = 047c1010b728bf3769e4df976dc57900afc9f39f$$

$$h'^1_{0,0} = ecaaa32a36fcfce8339ad4f4730027fd6af08d39$$

Yukarıda verilen özüt değerlerinin eşleşmemesi imgede bir bozulma olduğunu gösterir. Bu yüzden, sıradüzensel ağaç yapısının 2. seviyesindeki blokların doğrulanması yapılır. 2. seviyede, 4 blok ($I^2_{0,0}$, $I^2_{0,1}$, $I^2_{1,0}$, $I^2_{1,1}$) mevcuttur ve her bir bloğun doğrulanması için bloğun

yeniden hesaplanan özüt değerleri ile gömülen özüt değerleri karşılaştırılır. Bu karşılaştırmanın sonucunda $I_{0,0}^2$, $I_{0,1}^2$ ve $I_{1,1}^2$ bloklarının özüt değerleri eşleşirken, $I_{1,0}^2$ bloğunun özüt değerleri eşleşmemiştir. $I_{0,0}^2$, $I_{0,1}^2$ ve $I_{1,1}^2$ bloklarının doğrulanması, bu blokların alt seviyelerindeki blokların da doğrulanmasını sağlamaktadır. Ancak $I_{1,0}^2$ bloğunun doğrulanamamasından dolayı, daha alt seviyede hangi blokların bozulduğunu belirleyebilmek için, sıradüzensel ağaç yapısı içerisinde 8. seviyeye kadar blokları doğrulama işlemine devam edilir. Şekil 5.10. içerisinde 2. seviyedeki blokların doğrulanması gösterilmiştir. 2. seviyede doğrulanamayan $I_{1,0}^2$ bloğu tekrar bir alt seviye içerisinde 4 bloğa bölünür ve yukarıda olduğu gibi her bir bloğun doğrulanması yapılır. 3. seviyede, 4 blok ($I_{0,0}^3$, $I_{0,1}^3$, $I_{1,0}^3$, $I_{1,1}^3$) mevcuttur ve her bir bloğun doğrulanması için bloğun yeniden hesaplanan özüt değerleri ile gömülen özüt değerleri karşılaştırılır. Bu karşılaştırma sonucunda tüm blokların özüt değerleri eşleşmemiştir. Bu yüzden her bir blok tekrar 4 alt bloğa bölünmüştür.



(a) **(b)**

(c) **(d)**

Şekil 5.10. 2. Seviyede doğrulama işlemi **(a)** $I_{0,0}^2$ bloğu **(b)** $I_{0,1}^2$ bloğu **(c)** $I_{1,0}^2$ bloğu **(d)** $I_{1,1}^2$ bloğu

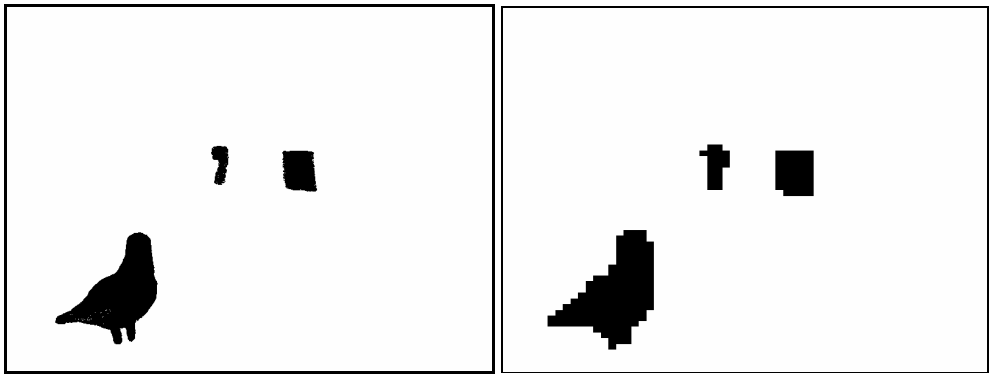
Bu bölmeleme işlemi 8. seviyedeki en küçük blok büyüklüğüne kadar devam ettirilmiş ve şekil 5.11.'deki doğrulama sonucu elde edilmiştir. Önerilen tekniğin doğrulama algoritmasının çıkışı

şekil 5.11'de gösterilmiştir. Şekil 5.11.'de doğrulanamayan bölgeler siyah ile gösterilmiştir. Doğrulanamayan piksel sayısı ise 24.226 olarak bulunmuştur.

Önerilen tekniğin doğrulama sonucuyla, imge üzerinde yapılan gerçek değişimi karşılaştırabilmek için, şekil 5.12.'de gerçek değişim ve önerilen tekniğin doğrulama sonucu gösterilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Gerçekte değiştirilen piksel sayısı ise 18.000'dir.



Şekil 5.11. Önerilen tekniğin doğrulama sonucu (çözünürlük=1200*1600 piksel)



(a) **(b)**

Şekil 5.12. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması (çözünürlük=1200*1600 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu

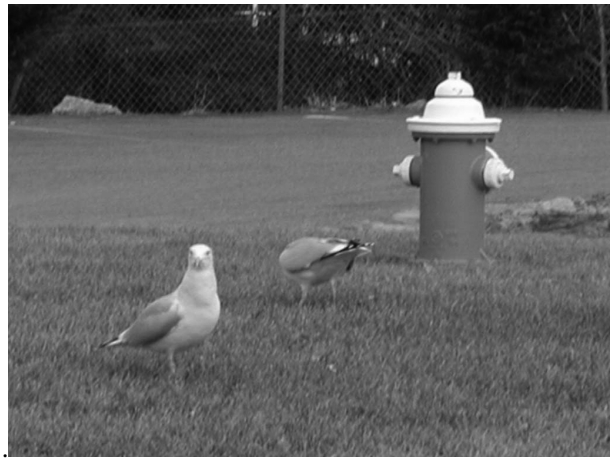
Önerilen tekniğin değişim bölgesi belirleme doğruluğunu değerlendirebilmek için, ikinci çalışma 768*1024 piksellik çözünürlükteki bir test imgesi üzerinde yapılmıştır. İmge ilk

olarak sıradüzensel ağaç yapısı kullanılarak 7. seviyede 12*16 piksellik blok büyüklüğüne kadar bölünmüştür. En alt seviyedeki en küçük bloğun taşıyacağı minimum veri yükü 88 bit olduğundan 7. seviyeye kadar bölmeleme yapılmıştır. Sıradüzensel ağaç yapısının en alt seviyesindeki bloklarda S_1 sayısal imza algoritması, kalan tüm seviyelerde S_2 sayısal imza algoritması kullanılarak toplam veri yükü $P_{i,j}^{L} = 173$ bit olarak bulunur. Bu veri yükü en alt seviyedeki her bir blok tarafından taşınabileceği için, 7. seviyede S_1 sayısal imza algoritması, kalan tüm seviyelerde (6, 5, 4, 3, 2, 1) S_2 sayısal imza algoritması kullanılarak sayısal imzalar hesaplanır ve bloğun EÖB düzlemine gömülür. Şekil 5.13. önerilen teknik ile filigranlanmış 768*1024 piksellik imgeyi göstermektedir.

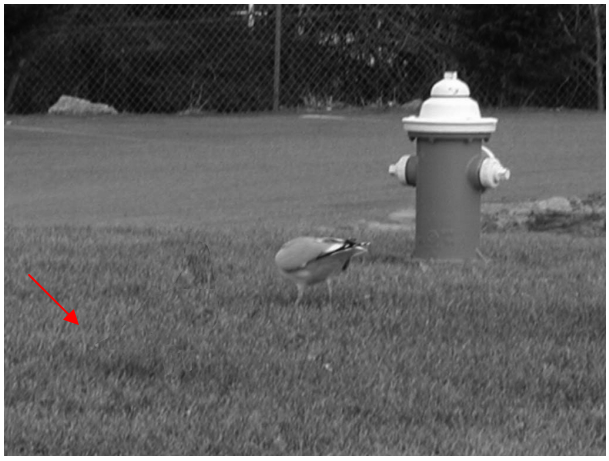
Önerilen tekniğin 768*1204 piksellik imgelerin doğrulanmasındaki performansını değerlendirebilmek için şekil 5.13.'de verilen filigranlı imge üzerinde öndeki martı silinerek değişim yapılmış ve şekil 5.14.'deki imge oluşturulmuştur.

Değiştirilmiş filigranlı imgenin doğrulanması için önerilen tekniğin doğrulama algoritması çalıştırılmış ve alınan sonuç şekil 5.15.'de gösterilmiştir. Şekil 5.15.'de doğrulanamayan bölgeler siyah ile gösterilmiştir. Doğrulanamayan piksel sayısı ise 26.496 olarak bulunmuştur.

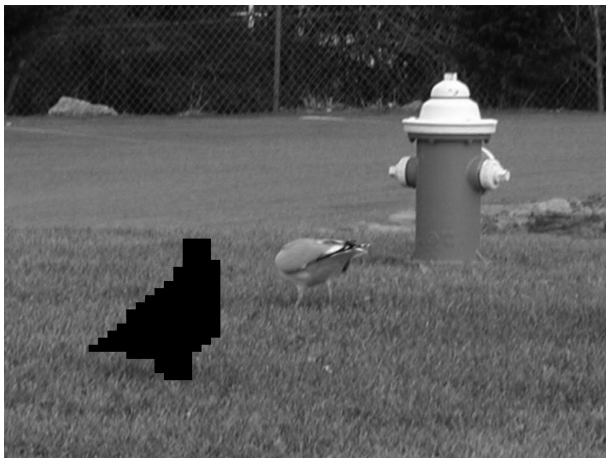
Önerilen tekniğin doğrulama sonucuyla, imge üzerinde yapılan gerçek değişimi karşılaştırabilmek için, şekil 5.16.'da gerçek değişim ve önerilen tekniğin doğrulama sonucu verilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Gerçekte değiştirilen piksel sayısı ise 20.459 olarak bulunmuştur.



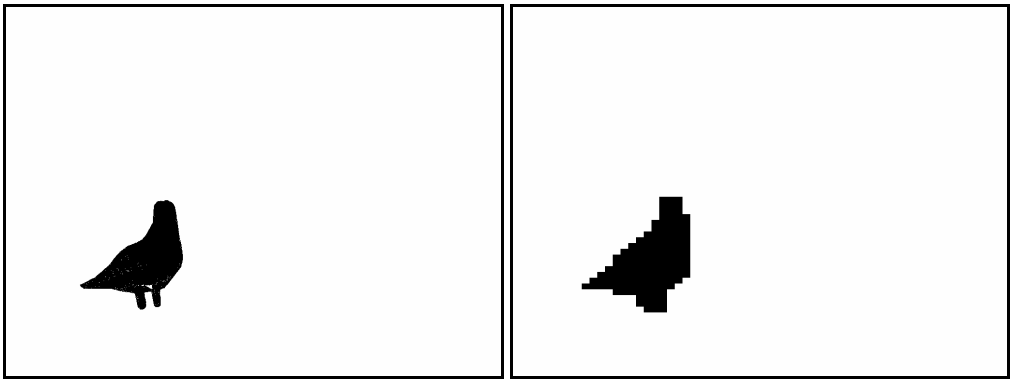
Şekil 5.13. Önerilen teknik ile filigranlanmış imge (çözünürlük=768*1024 piksel)



Şekil 5.14. Değiştirilmiş filigranlı imge (çözünürlük=768*1024 piksel)



Şekil 5.15. Önerilen tekniğin doğrulama sonucu (çözünürlük=768*1024 piksel)



Şekil 5.16. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması (çözünürlük=768*1024 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu

Önerilen tekniğin değişim bölgesi belirleme doğruluğunu değerlendirebilmek için, üçüncü çalışma 480*640 piksellik çözünürlükteki bir test imgesi üzerinde yapılmıştır. İmge ilk olarak sıradüzensel ağaç yapısı kullanılarak 6. seviyede 15*20 piksellik blok büyüklüğüne kadar bölünmüştür. En alt seviyedeki en küçük bloğun taşıyacağı minimum veri yükü 87 bit olduğundan 6. seviyeye kadar bölmeleme yapılmıştır. Sıradüzensel ağaç yapısının en alt seviyesindeki bloklarda S_1 sayısal imza algoritması, kalan tüm seviyelerde S_2 sayısal imza algoritması kullanılarak toplam veri yükü $P_{i,j}^{L} = 172$ bit olarak bulunur. Bu veri yükü en alt seviyedeki her bir blok tarafından taşınabileceği için, 6. seviyede S_1 sayısal imza algoritması, kalan tüm seviyelerde (5, 4, 3, 2, 1) S_2 sayısal imza algoritması kullanılarak sayısal imzalar hesaplanır ve bloğun EÖB düzlemine gömülür. Şekil 5.17. Önerilen teknik ile filigranlanmış 480*640 piksellik imgeyi göstermektedir. Önerilen tekniğin 480*640 piksellik imgelerin doğrulanmasındaki performansını değerlendirebilmek için şekil 5.17.'de verilen filigranlı imge üzerinde değişimler yapılmış ve şekil 5.18.'deki imge oluşturulmuştur. Şekil 5.18.'de değişim yapılan bölgeler oklarla gösterilmiştir.

Değiştirilmiş filigranlı imgenin doğrulanması için önerilen tekniğin doğrulama algoritması çalıştırılmış ve alınan sonuç şekil 5.19.'da gösterilmiştir. Şekil 5.19.'da doğrulanamayan bölgeler siyah ile gösterilmiştir. Doğrulanamayan piksel sayısı ise 12.600 olarak bulunmuştur.

Önerilen tekniğin doğrulama sonucuyla, imge üzerinde yapılan gerçek değişimi karşılaştırabilmek için, şekil 5.20.'de gerçek değişim ve önerilen tekniğin doğrulama sonucu verilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Gerçekte değiştirilen piksel sayısı ise 4.878 olarak bulunmuştur.



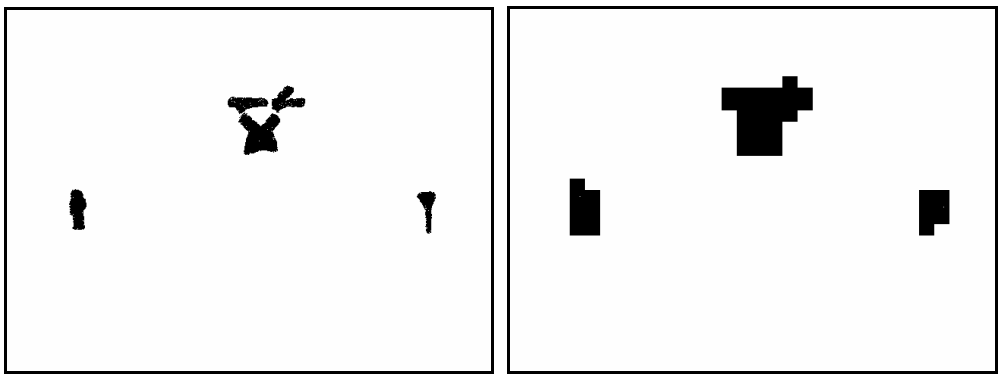
Şekil 5.17. Önerilen teknik ile filigranlanmış imge (çözünürlük=480*640 piksel)



Şekil 5.18. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)



Şekil 5.19. Önerilen tekniğin doğrulama sonucu (çözünürlük=480*640 piksel)



Şekil 5.20. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması (çözünürlük=480*640 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu

5.3.2. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniğinin Saldırılara Karşı Güvenirliği

Önerilen tekniğin; Kes Yapıştır, Vektör Nicemlemeli Taklit Saldırısı (VNTS), Doğum Günü Saldırısı ve Aktarıma Dayalı Saldırı'ya karşı güvenirliği aşağıda yapılan uygulamalar ile gösterilmiştir.

5.3.2.1. Kes Yapıştır Saldırısına Karşı Güvenirlik

Önerilen tekniğin Kes Yapıştır Saldırısı'na karşı güvenirliğini değerlendirebilmek için, 240*320 piksel çözünürlükteki bir test imgesi üzerinde çalışma yapılmıştır. İmge ilk olarak sıradüzensel ağaç yapısı kullanılarak 5. seviyede 15*20 piksellik blok büyüklüğüne kadar bölünmüştür. En alt seviyedeki en küçük bloğun taşıyacağı minimum veri yükü 87 bit olduğundan 5. seviyeye kadar bölmeleme yapılmıştır. Sıradüzensel ağaç yapısının en alt seviyesindeki bloklarda S_1 sayısal imza algoritması, kalan tüm seviyelerde S_2 sayısal imza algoritması kullanılarak toplam veri yükü $P_{i,j}^{L} = 171$ bit olarak bulunur. Bu veri yükü en alt seviyedeki her bir blok tarafından taşınabileceği için, 5. seviyede S_1 sayısal imza algoritması, kalan tüm seviyelerde (4, 3, 2, 1) S_2 sayısal imza algoritması kullanılarak sayısal imzalar hesaplanır ve bloğun EÖB düzlemine gömülür. Şekil 5.21., önerilen teknik ile filigranlanmış 240*320 piksellik imgeyi göstermektedir. Önerilen tekniğin Kes Yapıştır Saldırısı'na karşı güvenirliğini değerlendirebilmek için şekil 5.21.'de verilen filigranlı imge üzerinde kes-yapıştır saldırısı yapılmıştır. Bu saldırı, imge içerisindeki 66-75 satırları ve 221-240 sütunları (66:75, 221:240) arasındaki 15*20 piksellik bloğun kesilerek maskelenmek istenen blokların (36:45, 181:200; 36:45, 201:220; 46:60, 181:200; 46:60, 201:220, 61:75, 181:200; 61:75, 201:220) üzerine yapıştırılmasıyla yapılmıştır. Saldırı yapılmış imge şekil 5.22.'de gösterilmiştir.



Şekil 5.21. Önerilen teknik ile filigranlanmış imge (çözünürlük=240*320 piksel)



Şekil 5.22. Önerilen teknik ile filigranlanmış imgenin kes yapıştır saldırısı ile değiştirilmesi



(a) **(b)**

Şekil 5.23. Önerilen tekniğin doğrulama sonucu (çözünürlük 240*320 piksel)

(a) 5. seviyedeki doğrulama **(b)** 4. seviyedeki doğrulama

5.3.2.2. Vektör Nicemlemeli Taklit Saldırısına Karşı Güvenirlik

Önerilen tekniğin VNTS'ye karşı güvenirliğini değerlendirebilmek için, 128*192 piksel çözünürlükte ve VNTS kullanılarak oluşturulmuş bir test imgesi üzerinde çalışma yapılmıştır. Oluşturulan sahte test imgesindeki 8*12 piksellik her bir blok, önerilen teknik ile filigranlanmış imge veri tabanındaki benzer bir 8*12 piksellik bloğa nicemlenerek oluşturulmuştur. Oluşturulan imge şekil 5.24'de gösterilmiştir.

Saldırı yapılmış filigranlı imgenin doğrulanması için önerilen tekniğin doğrulama algoritması çalıştırılmıştır. Ancak buradaki doğrulama kes yapıştır saldırısında olduğu gibi en alt seviyedeki (5. seviye) bloklar tarafından yapılamamaktadır. Doğrulama algoritması, değişim yapılan bölgeyi 4. seviyedeki bloklar vasıtasıyla belirlemektedir.



Şekil 5.24. Vektör Nicemlemeli Taklit Saldırısıyla oluşturulmuş 128*192 piksellik imge



Şekil 5.25. VNTS'ye karşı önerilen tekniğin doğrulama sonucu **(a)** 5. seviyedeki doğrulama **(b)** 4. seviyedeki doğrulama

VNTS'de sahte imgenin 8*12 piksellik her bir bloğu, önerilen teknik ile filigranlanmış bir blok olduğu için en alt seviyede tüm bloklar doğrulanır. Ancak en alt seviyedeki 4 bloğun birleşmesiyle oluşan bir üst seviyedeki blokların özütleri eşleşmeyeceği için 16*24 piksellik bloklar doğrulanmayarak VNTS engellenir. Eğer sahte imge 16*24 piksellik blokların nicemlenmesiyle oluşturulursa, bu durumda da bir üst seviyedeki 32*48 piksellik bloklar VNTS'nin yapılmasını engelleyecektir. Önerilen tekniğin doğrulama algoritmasının 5. seviyedeki çıkışı, şekil 5.25.(a)'da ve 4. seviyedeki çıkışı şekil 5.25.(b)'de verilmiştir.

5.3.2.3. Doğum Günü Saldırısına Karşı Güvenirlik

Önerilen tekniğin Doğum Günü Saldırısı'na karşı güvenirliği S_2 sayısal imza algoritmasında kullanılan özüt fonksiyonunun uzunluğuna bağımlıdır. Yukarıda yapılan deneysel çalışmalarda özüt fonksiyonu olarak SHA algoritması kullanıldığından dolayı, herhangi bir blok üzerinde Doğum Günü Saldırısının yapılabilmesi için 2^{80} adet bloğa ihtiyaç duyulmaktadır [2]. Bir bloğun Doğum Günü Saldırısıyla değiştirilmesi durumunda, bu değişim sıradüzensel ağaç yapısının bir üst seviyesinde değiştirilen bölgeyi kapsayan blok tarafından bulunacağı için bu saldırının yapılması mümkün değildir.

5.3.2.4. Aktarıma Dayalı Saldırıya Karşı Güvenirlik

Önerilen teknik üzerinde Aktarıma Dayalı Saldırı'nın yapılabilmesi ise sıradüzensel ağaç yapısından dolayı mümkün değildir. Çünkü herhangi bir seviyedeki bloklar üzerinde bu saldırı yapıldığı takdirde, Kes-Yapıştır ve VNTS'de olduğu gibi bir üst seviyedeki bloklar tarafından değiştirilen bölgeler belirlenecektir.

5.3.3. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniği ile Sıradüzensel Filigranlama Tekniğinin Karşılaştırılması

Önerilen teknik ile Sıradüzensel Filigranlama Tekniği; saldırılara karşı güvenirlik, kriptografik güvenlik, değişim bölgesi belirleme, imge çözünürlüğünden bağımsız olarak filigranlama nitelikleri göz önünde tutularak karşılaştırılmıştır. İlk olarak aynı kriptografik güvenlik seviyesinde önerilen teknik ile Sıradüzensel Filigranlama Tekniğinin değişim bölgesi belirleme doğruluğu karşılaştırılmıştır.

Önerilen teknik ile Sıradüzensel Filigranlama Tekniği aynı kriptografik güvenlik seviyesinde değişim bölgesi belirleme doğruluğunu karşılaştırabilmek için Sıradüzensel Filigranlama Tekniğinde de SHA-1 algoritmasına dayanan 320 bitlik DSA sayısal imza algoritması kullanılmıştır. Sıradüzensel Filigranlama Tekniğinin değişim bölgesi belirleme doğruluğunu değerlendirebilmek için bölüm 5.3.1.'de verilen 1200*1600, 768*1024 ve 480*640 piksellik çözünürlüğe sahip aynı test imgeleri kullanılmıştır.

İlk olarak şekil 5.6.'da verilen 1200*1600 piksellik imgenin Sıradüzensel Filigranlama Tekniği kullanılarak filigranlanması yapılmıştır. Sıradüzensel Filigranlama Tekniğinde imge sıradüzen içerisinde üst üste gelmeyen bloklara bölünür. Sıradüzensel yapı kullanılarak imge sıradüzenin 5. seviyesine kadar bölünebilmektedir. Sıradüzenin 5. seviyesindeki blok boyutları 75*100 piksel büyüklüğündedir. Sıradüzensel Filigranlama Tekniğinde her bir blok içerisinde taşınacak veri yükü eşitlik 5.28 kullanılarak $P_{0,0}^{L} \cong 427$ bit olarak bulunur. Sıradüzensel Filigranlama Tekniği ile filigranlanmış 1200*1600 piksellik imge şekil 5.26.'da verilmiştir. Önerilen teknik ile Sıradüzensel Filigranlama Tekniğinin değişim bölgesi doğrulama niteliklerini karşılaştırabilmek için şekil 5.8.'e uygulanan aynı değişimler şekil 5.26.'da verilen imgeye de uygulanmıştır. Değiştirilmiş filigranlı imge şekil 5.27.'de verilmiştir.



Şekil 5.26. Sıradüzensel Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=1200*1600 piksel)



Şekil 5.27. Değiştirilmiş filigranlı imge (çözünürlük=1200*1600 piksel)

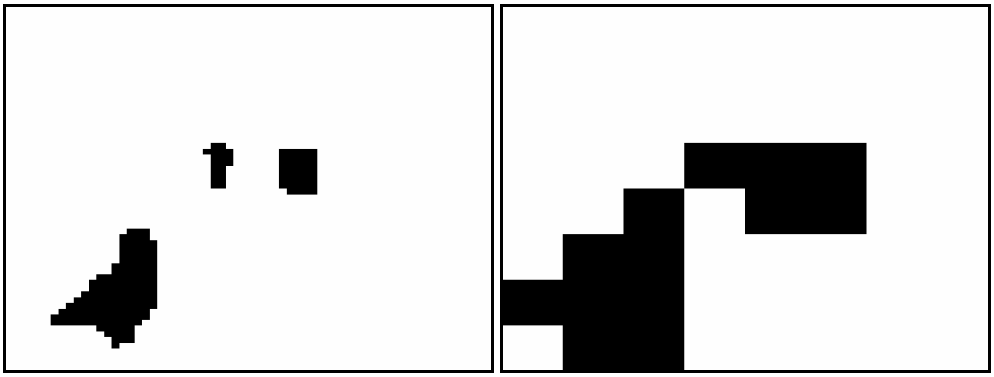
Değiştirilmiş filigranlı imgenin doğrulanması için Sıradüzensel Filigranlama Tekniğinin doğrulama algoritması çalıştırılmış ve alınan sonuç şekil 5.28'de gösterilmiştir. Şekil 5.28.'de doğrulanamayan bölgeler siyah ile gösterilmiştir. Sıradüzensel Filigranlama Tekniği tarafından doğrulanamayan piksel sayısı 97.500 olarak bulunmuştur.

Önerilen tekniğin doğrulama sonucuyla Sıradüzensel Filigranlama Tekniğinin doğrulama sonuçlarını karşılaştırabilmek için şekil 5.29.'da her iki tekniğin doğrulama sonuçları birlikte verilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Aynı kriptografik güvenlik seviyesinde önerilen teknik tarafından

doğrulanamayan piksel sayısı 24.226 iken, Sıradüzensel Filigranlama Tekniği tarafından doğrulanamayan piksel 97.500 olarak bulunmuştur.



Şekil 5.28. Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=1200*1600 piksel)

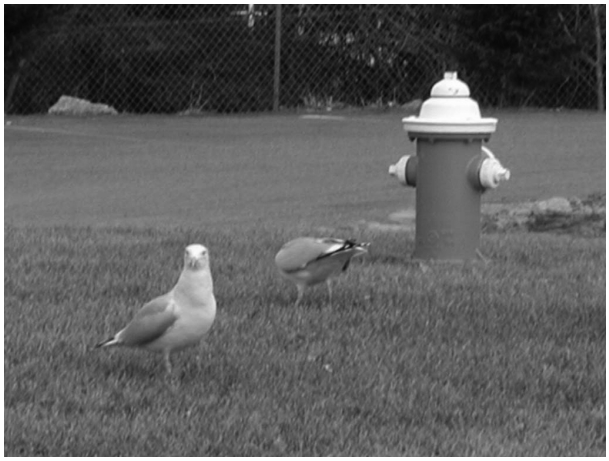


Şekil 5.29. Tekniklerin doğrulama sonuçlarının karşılaştırılması (çözünürlük=1200*1600 piksel)
(a) Önerilen tekniğin doğrulama sonucu **(b)** Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu

İkinci deneysel çalışma 768*1024 piksellik test imgesi üzerinde yapılmıştır. Önerilen teknik ile aynı kriptografik güvenlik seviyesini sağlamak için bu deneysel çalışmada da 320 bitlik DSA sayısal imza algoritması kullanılmıştır. Sıradüzensel Filigranlama Tekniğinin veri yükü $P_{0,0}^{L} \cong 427$ bit'dir. Sıradüzensel yapıda 427 bitlik veri yükünü taşıyabilecek blok büyüklüğü 6. seviyede 24*32 piksel büyülüğündedir. 768*1024 piksellik çözünürlükte Sıradüzensel Filigranlama Tekniği, veri yükünü taşıyacak minimum blok büyüklüğüne kadar

bölünebilmektedir. Sıradüzensel Filigranlama Tekniği ile filigranlanmış 768*1024 piksellik imge şekil 5.30.'da verilmiştir. Önerilen teknik ile Sıradüzensel Filigranlama Tekniğinin değişim bölgesi doğrulama niteliklerini karşılaştırabilmek için şekil 5.14.'e uygulanan aynı değişimler şekil 5.30.'da verilen imgeye de uygulanmıştır. Değiştirilmiş filigranlı imge şekil 5.31.'de verilmiştir.

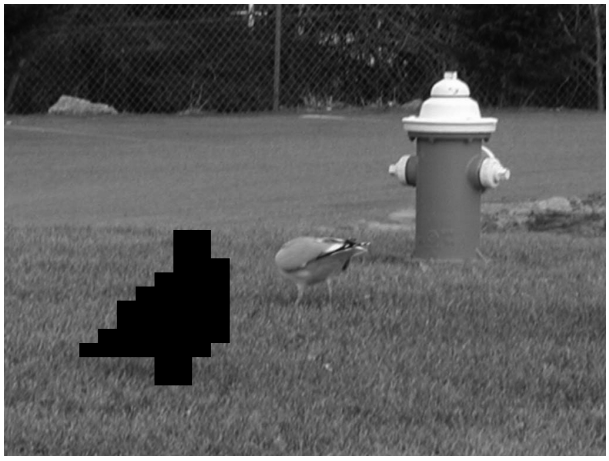
Değiştirilmiş filigranlı imgenin doğrulanması için Sıradüzensel Filigranlama Tekniğinin doğrulama algoritması çalıştırılmış ve alınan sonuç şekil 5.32.'de gösterilmiştir. Şekil 5.32.'de doğrulanamayan bölgeler siyah ile gösterilmiştir. Sıradüzensel Filigranlama Tekniği tarafından doğrulanamayan piksel sayısı 35.328 olarak bulunmuştur.



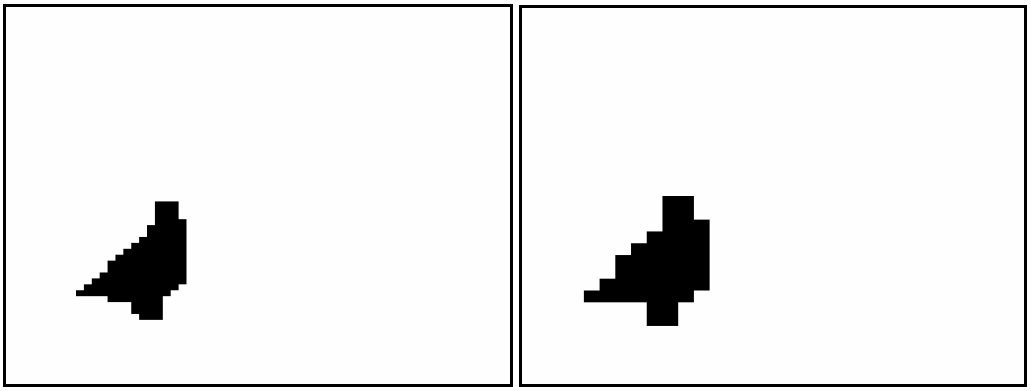
Şekil 5.30. Sıradüzensel Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=768*1024 piksel)



Şekil 5.31. Değiştirilmiş filigranlı imge (çözünürlük=768*1024 piksel)



Şekil 5.32. Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=768*1024 piksel)



Şekil 5.33. Tekniklerin doğrulama sonuçlarının karşılaştırılması (çözünürlük=768*1024 piksel)
(a) Önerilen tekniğin doğrulama sonucu **(b)** Sıradüzensel Filigranlama Tekniğin doğrulama sonucu

Önerilen tekniğin doğrulama sonucuyla Sıradüzensel Filigranlama Tekniğinin doğrulama sonuçlarını karşılaştırabilmek için şekil 5.33.'de her iki tekniğin doğrulama sonuçları birlikte verilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Aynı kriptografik güvenlik seviyesinde önerilen teknik tarafından doğrulanamayan piksel sayısı 26.496 iken, Sıradüzensel Filigranlama Tekniği tarafından doğrulanamayan piksel 35.328 olarak bulunmuştur. Sıradüzensel Filigranlama Tekniği 768*1024 piksellik çözünürlükte veri yükünü taşıyacak minimum blok büyüklüğüne kadar bölünebilmesine rağmen, değişim bölgesini önerilen teknikten çok daha az hassasiyette belirlemektedir.

Üçüncü deneysel çalışma 480*640 piksellik test imgesi üzerinde yapılmıştır. Önerilen teknik ile aynı kriptografik güvenlik seviyesini sağlamak için bu deneysel çalışmada da 320

bitlik DSA sayısal imza algoritması kullanılmıştır. Sıradüzensel Filigranlama Tekniğinin veri yükü $P_{0,0}^{L} \cong 427$ bit'dir. Sıradüzensel yapıda 427 bitlik veri yükünü taşıyabilecek blok büyüklüğü 5. seviyede 30*40 piksel büyülüğündedir. 480*640 piksellik çözünürlükte Sıradüzensel Filigranlama Tekniği, veri yükünü taşıyacak minimum blok büyüklüğüne kadar bölünebilmektedir. Sıradüzensel Filigranlama Tekniği ile filigranlanmış 480*640 piksellik imge şekil 5.34.'de verilmiştir.

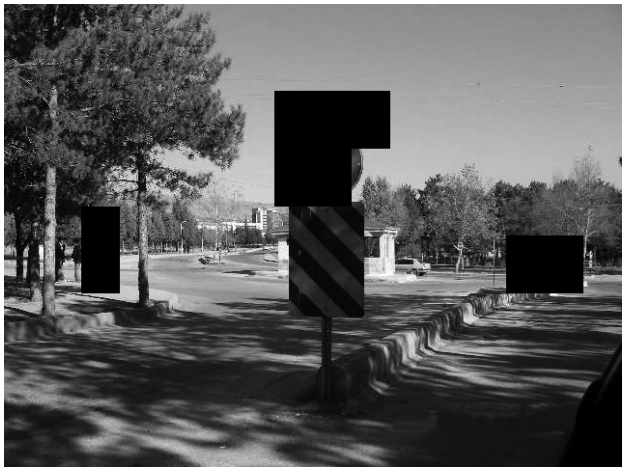
Önerilen teknik ile Sıradüzensel Filigranlama Tekniğinin değişim bölgesi doğrulama niteliklerini karşılaştırabilmek için şekil 5.18.'e uygulanan aynı değişimler şekil 5.34.'de verilen imgeye de uygulanmıştır. Değiştirilmiş filigranlı imge şekil 5.35.'de verilmiştir.



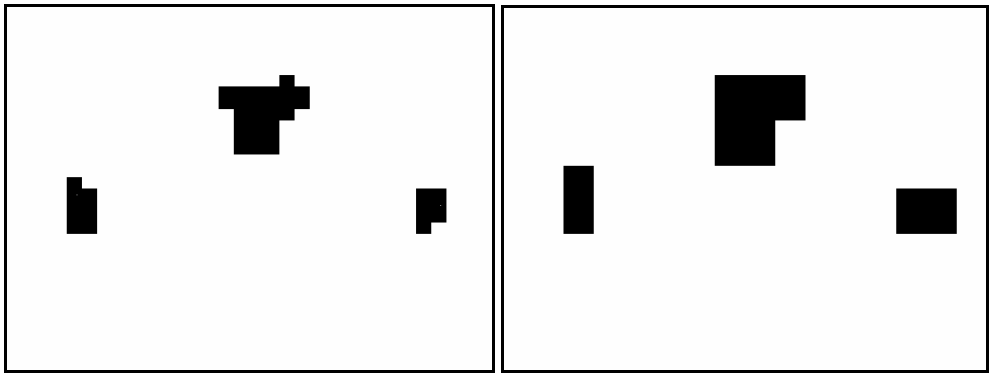
Şekil 5.34. Sıradüzensel Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=480*640 piksel)



Şekil 5.35. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)



Şekil 5.36. Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=480*640 piksel)



(a) **(b)**

Şekil 5.37. Tekniklerin doğrulama sonuçlarının karşılaştırılması (çözünürlük=480*640 piksel)

(a) Önerilen tekniğin doğrulama sonucu **(b)** Sıradüzensel Filigranlama Tekniğin doğrulama sonucu

Değiştirilmiş filigranlı imgenin doğrulanması için Sıradüzensel Filigranlama Tekniğinin doğrulama algoritması çalıştırılmış ve alınan sonuç şekil 5.36.'da gösterilmiştir. Şekil 5.36.'da doğrulanamayan bölgeler siyah ile gösterilmiştir. Sıradüzensel Filigranlama Tekniği tarafından doğrulanamayan piksel sayısı 20.400 olarak bulunmuştur.

Önerilen tekniğin doğrulama sonucuyla Sıradüzensel Filigranlama Tekniğinin doğrulama sonuçlarını karşılaştırabilmek için şekil 5.37.'de her iki tekniğin doğrulama sonuçları birlikte verilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Aynı kriptografik güvenlik seviyesinde önerilen teknik tarafından

doğrulanamayan piksel sayısı 12.600 iken, Sıradüzensel Filigranlama Tekniği tarafından doğrulanamayan piksel 20.400 olarak bulunmuştur.

Dördüncü deneysel çalışmada önerilen teknik ile Sıradüzensel Filigranlama Tekniği daha güvenilir bir sayısal imza algoritması kullanılarak karşılaştırılmıştır. RSA sayısal imza algoritmasında güvenlik, kullanılan anahtar uzunluğuna bağımlıdır. Daha uzun anahtarların kullanılması ise sayısal imzanın uzunluğunu artırır. RSA sayısal imza algoritmasında anahtar uzunluğu sayısal imzanın uzunluğuna eşittir [2]. Bu deneysel çalışmada sayısal imza algoritması olarak 2048 bitlik RSA sayısal imza algoritması seçilerek önerilen teknik ile Sıradüzensel Filigranlama Tekniği karşılaştırılmıştır. Karşılaştırma şekil 5.38.'de gösterilen 480*640 bitlik test imgesi üzerinde yapılmıştır.

İlk olarak 480x640 bitlik imge önerilen teknik ile filigranlanmıştır. Önerilen teknikte iki ayrı sayısal imza algoritması kullanıldığı için S_2 sayısal imza algoritması olarak, 2048 bitlik RSA sayısal imza algoritması ve S_1 sayısal imza algoritması olarak da MAC kullanılmıştır. Önerilen teknikte S_2 ve S_1 sayısal imzalarıyla hesaplanan minimum veri yükü 88 bit olarak bulunur. Minimum veri yükünü taşıyacak blok ise sıradüzensel ağaç yapısının 6. seviyesinde 15*20 piksel boyutundadır. Sıradüzensel ağaç yapısının en alt iki seviyesinde (6, 5) S_1 sayısal imza algoritması ve kalan tüm seviyelerde S_2 sayısal imza algoritması kullanılarak toplam veri yükü ise 250 bit olarak bulunur. Önerilen teknik ile filigranlanmış imge şekil 5.39.'da gösterilmiştir.



Şekil 5.38. 480*640 piksellik test imgesi



Şekil 5.39. Önerilen teknik ile filigranlanmış imge (çözünürlük=480*640 piksel)



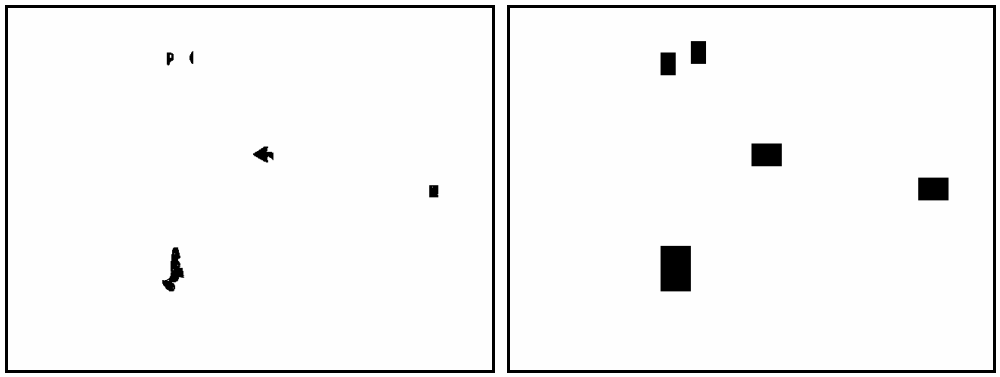
Şekil 5.40. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)

Önerilen tekniğin 2048 bitlik sayısal imza algoritması kullanıldıktan sonraki performansını değerlendirebilmek için şekil 5.39.'da verilen filigranlı imge üzerinde değişimler yapılmış ve şekil 5.40.'daki imge oluşturulmuştur.

Değiştirilmiş filigranlı imgenin doğrulanması için önerilen tekniğin doğrulama algoritması çalıştırılmış ve alınan sonuç şekil 5.41.'de gösterilmiştir. Şekil 5.41.'de doğrulanamayan bölgeler siyah ile gösterilmiştir. Doğrulanamayan piksel sayısı ise 6.000 olarak bulunmuştur.



Şekil 5.41. Önerilen tekniğin doğrulama sonucu (çözünürlük=480*640)



(a) **(b)**

Şekil 5.42. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması (çözünürlük=480*640 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu

Önerilen tekniğin doğrulama sonucuyla, imge üzerinde yapılan gerçek değişimi karşılaştırabilmek için, şekil 5.42.'de gerçek değişim ve önerilen tekniğin doğrulama sonucu gösterilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Gerçekte değiştirilen piksel sayısı ise 1.418 olarak bulunmuştur.

Sıradüzensel Filigranlama Tekniğinde de 2.048 bitlik RSA sayısal imza algoritması kullanılarak şekil 5.38.'de verilen test imgesi filigranlanmıştır. Sıradüzensel Filigranlama Tekniğinde 2.048 bitlik bir sayısal imza kullanıldığında oluşan veri yükü 2.730 bittir. Bu veri yükünü taşıyabilecek bloklar ise 4. seviyedeki 60*80 piksellik büyüklüktedir. Sıradüzensel Filigranlama Tekniği ile filigranlanmış imge Şekil 5.43.'de gösterilmiştir.



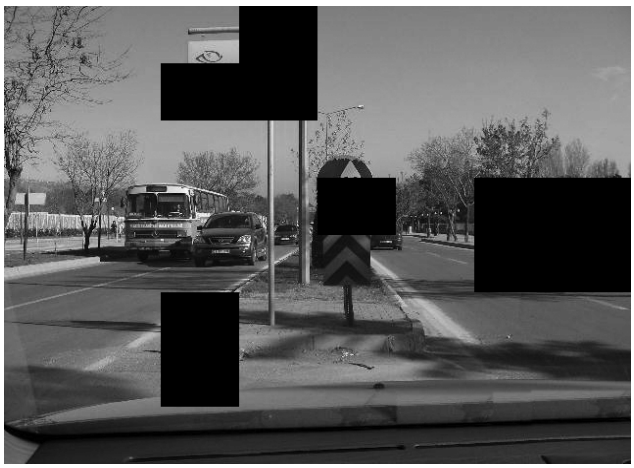
Şekil 5.43. Sıradüzensel Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=480*640 piksel)



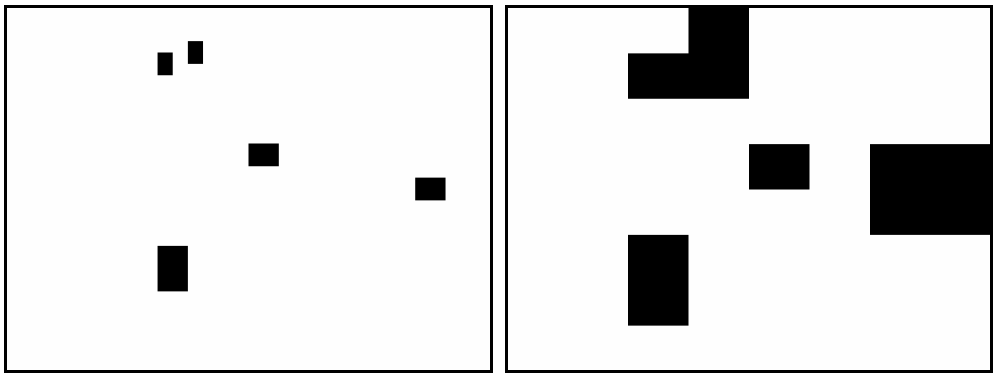
Şekil 5.44. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)

2.048 bitlik RSA sayısal imzası kullanılması durumunda, önerilen teknik ile Sıradüzensel Filigranlama Tekniğinin değişim bölgesi doğrulama niteliklerini karşılaştırabilmek için, şekil 5.40.'a uygulanan aynı değişimler şekil 5.43.'de verilen Sıradüzensel Filigranlama Tekniği ile filigranlanmış imgeye de uygulanmıştır. Değiştirilmiş filigranlı imge şekil 5.44.'de verilmiştir.

Değiştirilmiş filigranlı imgenin doğrulanması için Sıradüzensel Filigranlama Tekniğinin doğrulama algoritması çalıştırılmış ve alınan sonuç şekil 5.45.'de gösterilmiştir. Şekil 5.45.'de doğrulanamayan bölgeler siyah ile gösterilmiştir. Sıradüzensel Filigranlama Tekniği tarafından doğrulanamayan piksel sayısı 48.000 olarak bulunmuştur.



Şekil 5.45. Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=480*640 piksel)



(a) **(b)**

Şekil 5.46. Tekniklerin doğrulama sonuçlarının karşılaştırılması (çözünürlük=480*640 piksel)
(a) Önerilen tekniğin doğrulama sonucu **(b)** Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu

Önerilen tekniğin doğrulama sonucuyla Sıradüzensel Filigranlama Tekniğinin Doğrulama sonuçlarını karşılaştırabilmek için şekil 5.46.'da her iki tekniğin doğrulama sonuçları birlikte verilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Aynı kriptografik güvenlik seviyesinde önerilen teknik tarafından doğrulanamayan piksel sayısı 6.000 iken, Sıradüzensel Filigranlama Tekniği tarafından doğrulanamayan piksel 48.000 olarak bulunmuştur.

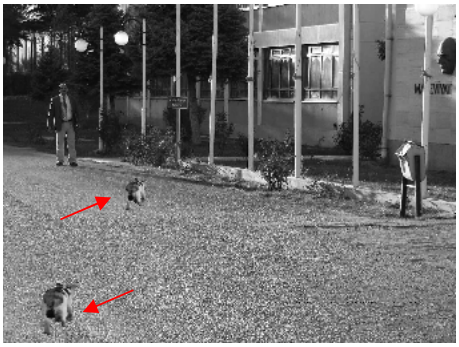
Beşinci deneysel çalışma önerilen tekniğin imge çözünürlüğünden bağımsız olarak çalıştığını göstermek için yapılmıştır. Sıradüzensel Filigranlama Tekniğinin en önemli dezavantajı imge en üst seviyede bir alt seviyeye bölünmediğinde ortaya çıkmaktadır. Örneğin şekil 5.47.'de verilen 255*340 piksellik imge sıradüzenin en üst seviyesinde bir alt seviyeye bölünemediğinden dolayı filigranlama sadece en üst seviyede tüm imge pikselleri kullanılarak yapılır.



Şekil 5.47. 255*340 piksellik test imgesi



Şekil 5.48. Sıradüzensel Filigranlama Tekniği ile filigranlanmış test imgesi (çözünürlük=255*340 piksel)



Şekil 5.49. Değiştirilmiş filigranlı imge (çözünürlük=255*340 piksel)

Ancak bu durumda imgenin bir pikselinin bile bozulması tüm imgenin doğrulanamamasına neden olarak Sıradüzensel Filigranlama Tekniğinin değişim bölgesi belirleme niteliğini yok etmektedir. Sıradüzensel Filigranlama Tekniği ile filigranlanmış 255*340 piksellik imge şekil 5.48.'de verilmiştir. Şekil 5.48.'deki Sıradüzensel Filigranlama Tekniği ile filigranlanmış imge değiştirilerek şekil 5.49.'daki imge oluşturulmuştur. Sıradüzensel Filigranlama Tekniğinin doğrulama algoritması çalıştırılarak elde edilen sonuç

şekil 5.50.'de gösterilmiştir. Şekil 5.50.'de görüldüğü gibi piksel değerlerindeki bir bozulma tüm imgenin doğrulanamamasına neden olmuştur. Sıradüzensel Filigranlama Tekniği imgeyi en üst seviyede bir alt seviyeye bölemediği takdirde, imgenin doğrulamasını yapabilmektedir, fakat değişim yapılan bölgeyi belirleyememektedir.

Sıradüzensel ağaç yapısında tavan ve taban fonksiyonları kullanılarak bu problem tamamen çözülmüştür. Önerilen teknik ile şekil 5.47.'de verilen test imgesi 5. seviyede 15*21 piksellik bloklara kadar bölünerek filigranlanmıştır. Filigranlama yordamında S_2 sayısal imza algoritması olarak DSA ve S_1 sayısal imza algoritması olarak MAC kullanılmıştır. En alt seviyede S_1 kalan tüm seviyelerde S_2 sayısal imzası kullanılarak en alt seviyedeki en küçük bloğun taşıyacağı toplam veri yükü ise 171 bit olarak bulunmuştur. Önerilen teknik ile filigranlanmış imge şekil 5.51.'de gösterilmiştir. Önerilen teknik ile Sıradüzensel Filigranlama Tekniğinin değişim bölgesi doğrulama niteliklerini karşılaştırabilmek için, şekil 5.48'e uygulanan aynı değişimler şekil 5.51.'de gösterilen ve önerilen teknik ile filigranlanmış imgeye de uygulanmıştır. Değiştirilmiş filigranlı imge şekil 5.52.'de verilmiştir.



Şekil 5.50. Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=255*340 piksel)



Şekil 5.51. Önerilen teknik ile filigranlanmış imge (çözünürlük=255*340 piksel)



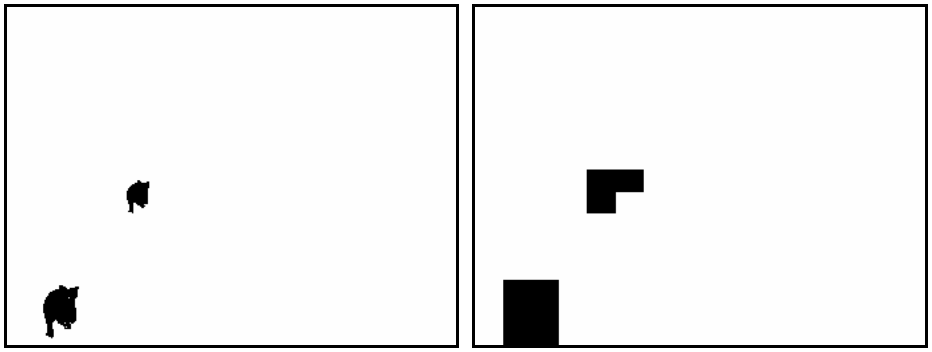
Şekil 5.52. Değiştirilmiş filigranlı imge (çözünürlük=255*340 piksel)



Şekil 5.53. Önerilen tekniğin doğrulama sonucu (çözünürlük=255*340 piksel)

Değiştirilmiş filigranlı imgenin doğrulanması için önerilen tekniğin doğrulama algoritması çalıştırılmış ve alınan sonuç şekil 5.53.'de gösterilmiştir. Şekil 5.53.'de doğrulanamayan bölgeler siyah ile gösterilmiştir. Doğrulanamayan piksel sayısı ise 3.141 olarak bulunmuştur.

Önerilen tekniğin doğrulama sonucuyla, imge üzerinde yapılan gerçek değişimi karşılaştırabilmek için, şekil 5.54.'de gerçek değişim ve önerilen tekniğin doğrulama sonucu gösterilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Gerçekte değiştirilen piksel sayısı ise 955 olarak bulunmuştur.Sıradüzensel Filigranlama Tekniğinin doğrulama sonucu ise 5.54. (c)'de verilmiştir.



(a) **(b)**



(c)

Şekil 5.54. Doğrulama sonuçlarının karşılaştırılması (çözünürlük=255*340 piksel)
(a) Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu **(c)** Sıradüzensel Filigranlama Tekniğinin Doğrulama sonucu

Ayrıca önerilen teknik ile Sıradüzensel Filigranlama Tekniğinin doğrulama çözünürlükleri 3 ayrı çözünürlükte ve 15 farklı test imgesi üzerinde karşılaştırılmıştır. Karşılaştırmada aynı kriptografik güvenlik seviyesini sağlamak için SHA-1 özüt fonksiyonunu kullanan 320 bitlik DSA sayısal imza algoritması kullanılarak yapılmıştır. Elde edilen test sonuçları tablo 5.1.'de verilmiştir. Test imgeleri ve elde edilen imgesel sonuçlar ise Ek-1'de verilmiştir. Tablo 5.1.'den de görüldüğü gibi aynı kriptografik güvenlik seviyesinde önerilen teknik, Sıradüzensel Filigranlama Tekniğine göre daha yüksek doğrulukta değişim bölgesi belirleme niteliğine sahiptir.

Tablo 5.1. Sıradüzensel Filigranlama Tekniği ile önerilen tekniğin karşılaştırılması

Test imgesi	**Çözünürlük (piksel)**	**Sıradüzensel Filigranlama Tekniğinin Veri yükü (bit)**	**Önerilen Tekniğin Veri yükü (bit)**	**Sıradüzensel Filigranlama Tekniğinin En Alt Seviyedeki Blok Büyüklüğü (piksel)**	**Önerilen Tekniğin En Alt Seviyedeki Blok Büyüklüğü (piksel)**	**Değiştirilen Piksel Sayısı**	**Sıradüzensel Filigranlama Tekniği Tarafından Doğrulanamayan Piksel Sayısı**	**Önerilen Teknik Tarafından Doğrulanamayan Piksel Sayısı**
İmge-1	1200*1600	427	94	75*100	9*12	53.819	150.000	63.374
İmge-2	1200*1600	427	94	75*100	9*12	5.890	60.000	12.226
İmge-3	1200*1600	427	94	75*100	9*12	87.075	187.500	99.696
İmge-4	1200*1600	427	94	75*100	9*12	195.401	397.500	222.233
İmge-5	1200*1600	427	94	75*100	9*12	27.759	127.500	43.760
İmge-6	768*1024	427	173	24*32	12*16	32.173	61.440	44.928
İmge-7	768*1024	427	173	24*32	12*16	8.559	43.776	22.848
İmge-8	768*1024	427	173	24*32	12*16	7.879	33.792	18.624
İmge-9	768*1024	427	173	24*32	12*16	17.683	39.168	29.568
İmge-10	768*1024	427	173	24*32	12*16	8.735	23.808	17.856
İmge-11	480*640	427	172	30*40	15*20	1.418	13.200	6.000
İmge-12	480*640	427	172	30*40	15*20	3.689	27.600	17.100
İmge-13	480*640	427	172	30*40	15*20	5.575	31.200	17.700
İmge-14	480*640	427	172	30*40	15*20	6.885	38.400	22.800
İmge-15	480*640	427	172	30*40	15*20	4.878	20.400	12.600

5.3.4. Deneysel Sonuçların Değerlendirilmesi

Sıradüzensel Filigranlama Tekniği'nin en önemli dezavantajı imge en üst seviyede bir alt seviyeye bölünemediğinde ortaya çıkmaktadır. Bu durumda Sıradüzensel Filigranlama Tekniği değişim bölgesi belirleme niteliğini kaybetmektedir. Bu durum şekil 5.47.'de verilen 255*340 piksellik test imgesi üzerinde yapılan deneysel çalışmada görülmektedir. Önerilen Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniği'nde tavan ve taban fonksiyonları kullanılarak bu problem çözülmüştür. Şekil 5.47.'de verilen test imgesi üzerinde önerilen teknik ile yapılan deneysel çalışmada değişim bölgesi belirleme doğruluğu 15*21 piksellik bir çözünürlükte belirlenebilmektedir. Şekil 5.54.'de de görüldüğü gibi önerilen teknik ile değişim yapılan bölge belirlenebilirken, Sıradüzensel Filigranlama Tekniği ile değişim bölgesi belirlenememektedir.

Sıradüzensel Filigranlama Tekniğinin değişim bölgesi belirleme doğruluğu imgenin sıradüzen içerisinde alt seviyelere bölünebilmesine bağımlıdır. Örneğin şekil 5.6.'da verilen 1200*1600 piksellik imge Sıradüzensel Filigranlama Tekniği ile sıradüzenin 5.seviyesinde 75*100 piksellik bloklara kadar bölünebilmektedir. İmgenin sıradüzensel yapı içerisinde veri yükünü taşıyacak minimum boyutlu bloklara kadar bölünememesi, Sıradüzensel Filigranlama Tekniği'nin değişim bölgesi belirleme doğruluğunu oldukça azaltmıştır. Bu durum şekil 5.28.'de gösterilmiştir. Önerilen teknikte ise tavan ve taban fonksiyonları kullanılarak imgenin veri yükünü taşıyacak minimum boyutlu bloklara kadar bölünmesi garanti altına alınmıştır. 1200*1600 piksellik test imgesi üzerinde yapılan deneysel çalışmada, aynı kriptografik güvenlik seviyesinde ve aynı değişimler altında önerilen teknik tarafından doğrulanamayan piksel sayısı 24.226 iken, Sıradüzensel Filigranlama Tekniği tarafından doğrulanamayan piksel sayısı 97.500 olarak bulunmuştur. Önerilen Teknik ile Sıradüzensel Filigranlama Tekniği'nin doğrulama sonuçları şekil 5.29.'da verilmiştir. Ayrıca bu durumun tablo 5.1'de verilen 1200*1600 piksellik ilk 5 test imgesi içinde geçerli olduğu yapılan deneysel çalışmalar sonucunda görülmüştür.

Sıradüzensel Filigranlama Tekniği ile Önerilen tekniğin değişim bölgesi belirleme doğruluğu, Sıradüzensel Filigranlama Tekniği'nin veri yükünü taşıyacak minimum boyutlu bloklara bölünebildiği 768*1024 ve 480*640 piksellik test imgeleri üzerinde de karşılaştırılmıştır. Aynı güvenlik seviyesinde ve aynı değişimler uygulanarak yapılan karşılaştırma sonuçları şekil 5.33. ve 5.37.'de gösterilmiştir. 768*1024 piksellik test imgesi üzerinde önerilen teknik tarafından doğrulanamayan piksel sayısı 26.496 iken, Sıradüzensel Filigranlama Tekniği tarafından doğrulanamayan piksel sayısı 35.328 olarak bulunmuştur. 480*640 piksellik test imgesi üzerinde yapılan deneysel çalışmada ise, önerilen teknik tarafından doğrulanamayan piksel sayısı 12.600 iken, Sıradüzensel Filigranlama Tekniği

tarafından doğrulanamayan piksel sayısı 20.400 olarak bulunmuştur. Önerilen tekniğin Sıradüzensel Filigranlama Tekniğine göre daha yüksek doğrulukta değişim bölgesi belirleme niteliğine sahip olduğu yapılan diğer test imgeleri üzerinde de görülmüştür. 768*1024 ve 480*640 piksellik test imgeleri üzerinde yapılan deneysel çalışmaların sonucu tablo 5.1.'de verilmiştir.

Önerilen tekniğin Sıradüzensel Filigranlama Tekniğine göre diğer önemli bir avantajı daha yüksek kriptografik güvenirlik seviyesine ve daha uzun bit çıkışına sahip sayısal imzalar kullanıldığında ortaya çıkmaktadır. 2048 bitlik RSA sayısal imza algoritması kullanılarak 480*640 piksellik test imgesi üzerinde yapılan deneysel çalışmanın sonucu şekil 5.46.'da verilmiştir. Önerilen teknik tarafından doğrulanamayan piksel sayısı 6.000 iken, Sıradüzensel Filigranlama Tekniği tarafından doğrulanamayan piksel sayısı 48.000 olarak bulunmuştur.

Önerilen Sıradüzensel Ağaç yapılı Kırılgan Filigranlama Tekniği'nde, alt seviyedeki bloklara ait sayısal imzaların hesaplanmasında kriptografik güvenlik seviyesi daha düşük bir sayısal imzanın kullanılması önerilen tekniğin genel güvenlik seviyesini azaltmamaktadır. Çünkü alt seviyedeki bloklara uygulanan bir saldırı üst seviyelerdeki bloklara ait daha güvenilir sayısal imzalar tarafından belirlenmektedir. Örneğin şekil 5.22.'de gösterilen Kes Yapıştır Saldırısı'nda değiştirilen bloklar, üst seviyelerde bu blokları barındıran tüm bloklara ait sayısal imzalar tarafından doğrulanmayarak değişim bölgesi belirlenmektedir. Yine şekil 5.24.'de verilen ve VNTS kullanılarak oluşturulmuş test imgesi, üst seviyelerdeki tüm bloklara ait sayısal imzalar tarafından doğrulanmayarak saldırı yapılması engellenmiştir.

6. ALT BLOK YAKLAŞIMLI KIRILGAN FİLİGRANLAMA TEKNİĞİ

Filigranlama yordamında bloklar arasında bağımlılığın kurulmasıyla, saldırılara karşı güvenilir bir kırılgan filigranlama tekniği geliştirilebilmektedir. Bloksal bağımlılıkta her bir blok içerisine gömülen filigran, bloğun kendi pikselleri ve bağımlı olduğu blokların piksellerinden hesaplanan bir sayısal imzadan oluşur. Bloksal bağımlılığı kullanan Kırılgan Filigranlama Tekniklerinde değişim bölgesi belirleme doğruluğu ise kullanılan bağımlılık sayısına göre değişmektedir. Değişim bölgesini daha yüksek doğrulukta belirleyebilmek için, genellikle blok başına bir bağımlılığın kullanılması önerilir [42]. Ancak bir bağımlılığın kullanılması durumunda bile, bir bloğun bir pikselindeki bozulma iki bloğun birden doğrulanamamasına neden olur. Ayrıca sayısal imzaların filigran olarak kullanıldığı diğer filigranlama tekniklerinde olduğu gibi, bloksal bağımlılığa dayanan kırılgan filigranlama tekniklerinde de sayısal imzaların uzunluğu değişim bölgesi belirleme doğruluğunu etkilemektedir.

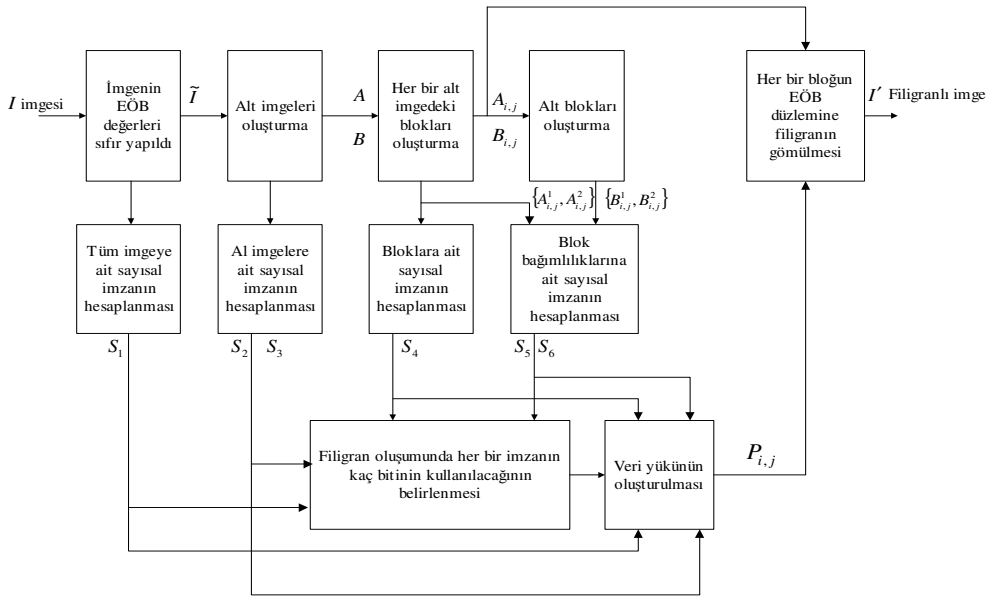
Bu tezde önerilen ikinci kırılgan filigranlama tekniği bloksal bağımlılığa dayanmaktadır. Önerilen teknikte imgenin bloklara bölünmesinde yeni bir yaklaşım geliştirilmiştir. Alt bloklama yapısı olarak adlandırılan bu yaklaşımda, doğrulanacak olan imge iki alt imgeye, her bir alt imge üst üste gelmeyen bloklara, ve bloklar alt bloklara bölünmüştür. Daha sonra her bir alt imge içerisindeki bloklar, diğer alt imgedeki iki farklı alt blok ile bağımlı hale getirilmiştir. Bu bağımlılıklar hem saldırılara karşı güvenliği sağlamak amacıyla hem de değişim bölgesini yüksek bir doğrulukta belirlemek amacıyla kurulmuştur. Önerilen teknikte saldırılara karşı tam bir güvenlik seviyesi elde edebilmek için tüm imgeye ve alt imgelere ait sayısal imzalar kullanılmıştır. Bu imzaların önerilen tekniğin değişim bölgesi belirleme doğruluğunu etkilememesini sağlamak için, tüm imgeye ait sayısal imza her iki alt imge içerisindeki bloklara ve alt imgelere ait sayısal imzalarda her bir alt imgenin kendi blokları içerisine dağıtılmıştır. Önerilen teknikte her bir alt imgedeki bloğun diğer alt imgedeki iki alt blok ile bağımlı hale getirilmesi, bir pikseldeki bozulmanın, pikselin ait olduğu bloğun ve bu bloğun bağımlı olduğu alt blokların doğrulanmamasını gerektirir. Ancak önerilen teknikte yeni bir doğrulama algoritması geliştirilerek, bir pikseldeki bozulmanın sadece o bloğun yarısının doğrulanamamasına neden olması sağlanmıştır. Bu çözüm değişim bölgesinin 4 kata kadar daha doğru bir şekilde belirlenmesine imkan sağlamaktadır. Dolayısıyla geliştirilen algoritma diğer bloksal bağımlılığa sahip teknikler açısından da önemlidir. Ayrıca bu teknikte iki ayrı sayısal imza algoritmasının kullanılması önerilmiştir. Daha yüksek güvenlik seviyesine ve uzun bit çıkışına sahip sayısal imza tüm imgenin ve alt imgelerin doğrulanmasında kullanılırken, daha

düşük güvenlik seviyesine ve kısa bit çıkışına sahip sayısal imza, blok ve bu blokların alt bloklarla oluşturdukları bağımlılıkların doğrulanmasında kullanılmıştır. Bu öneri daha küçük blok büyüklüklerinin seçilmesini sağlayarak, önerilen tekniğin değişim bölgesi belirleme doğruluğunu artırmıştır.

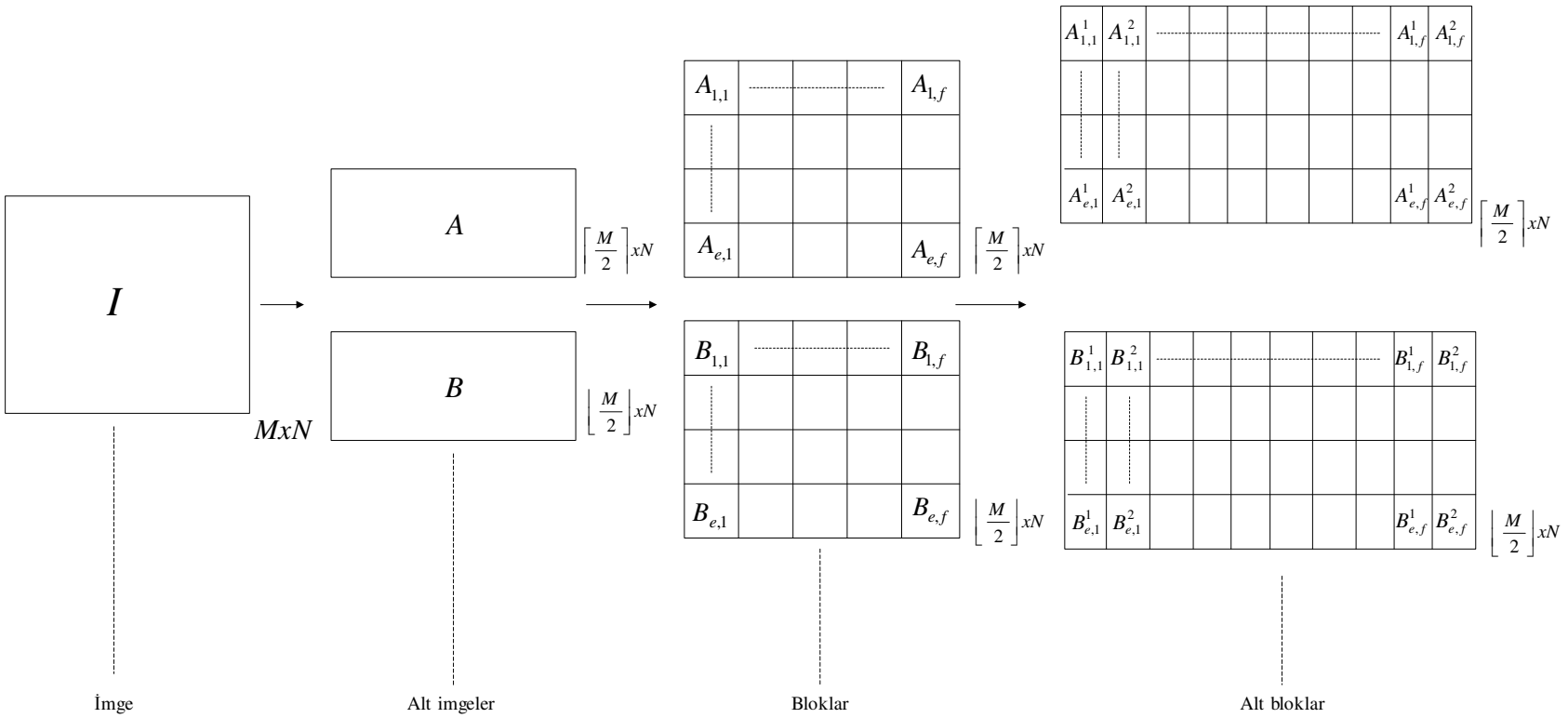
6.1. Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinin Filigranlama Yordamı

Önerilen teknikte ilk olarak imge alt bloklama yapısı kullanılarak üst üste gelmeyen alt imgelere, bloklara ve alt bloklara bölünür. Daha sonra her bir alt imgedeki bloklar, diğer alt imgedeki alt bloklar ile bağımlı hale getirilir. Tüm bloklar için bağımlılık kurulduktan sonra, tüm imgeye, alt imgelere, bloklara, blokların alt bloklar ile kurdukları bağımlılıklara ait sayısal imzalar hesaplanır ve bu imzalar birleştirilerek veri yükü oluşturulur. Son olarak veri yükü her bir blok içerisine gömülerek imge filigranlanır. Önerilen tekniğin filigranlama yordamı şekil 6.1.'de gösterilmiştir.

Önerilen tekniğin filigranlama yordamını daha iyi açıklayabilmek için, filigranlama yordamı beş aşama içerisinde anlatılabilir. İlk aşamada, imgenin EÖB düzlemi sıfır yapılır ve alt blok yaklaşımı olarak adlandırdığımız bölmeleme yapısı kullanılarak imge üst üste gelmeyen bloklara bölünür. Bu yapıda imgeler alt imgelere, alt imgeler bloklara ve bloklar iki alt bloğa bölünmektedir. $M * N$ piksellik I imgesinin alt blok yaklaşımı kullanılarak bloklara bölünmesi şekil 6.2.'de gösterilmiştir.



Şekil 6.1. Alt blok yaklaşımlı kırılgan filigranlama tekniğinin filigranlama yordamı



Şekil 6.2. Alt blok yaklaşımı kullanılarak imgenin bloklara bölünmesi

Şekil 6.2.'de I imgesi ilk olarak $\left\lceil \frac{M}{2} \right\rceil * N$ piksellik A ve $\left\lfloor \frac{M}{2} \right\rfloor * N$ piksellik B alt imgelerine bölünür. Daha sonra her bir alt imge üst üste gelmeyen $r * s$ piksellik $A_{i,j}$ ve $B_{i,j}$ bloklarına bölünür. Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniği'nde olduğu gibi bu teknikte de tavan ve taban fonksiyonları kullanılarak imgenin tüm çözünürlüklerinde bölünebilmesi sağlanmıştır. Son olarak her bir alt imgedeki bloklar $r * (s/2)$ piksellik $\{A^1_{i,j}, A^2_{i,j}\}$ ve $\{B^1_{i,j}, B^2_{i,j}\}$ gibi iki eşit alt bloğa bölünür. Burada $i = \{1,2,...,e\}$ ve $j = \{1,2,...,f\}$ bloğun imge içerisindeki konumunu göstermektedir. Ayrıca $e = (\lceil M \rceil / 2)/r$ ve $f = (N/s)$'dir.

İkinci aşama bloksal bağımlılıkların kurulmasından oluşmaktadır. Önerilen teknikte bloksal bağımlılık, her bir alt imgedeki bloğun diğer alt imgedeki alt bloklar ile birleştirilmesiyle oluşturulur. A alt imgesindeki her bir $A_{i,j}$ bloğu B alt imgesindeki $B^2_{i,j-1}$ ve $B^1_{i,j+1}$ alt blokları ile bağımlı hale getirilir ve benzer şekilde B alt imgesindeki her bir $B_{i,j}$ bloğu A alt imgesindeki $A^2_{i,j-1}$ ve $A^1_{i,j+1}$ alt blokları ile bağımlı hale getirilir.

Üçüncü aşamada tüm imgeye, alt imgelere, bloklara ve kurulan her bir bağımlılığa ait sayısal imzalar hesaplanır. Sayısal imzaların hesaplanması için iki ayrı sayısal imza algoritmasının kullanılması önerilmiştir. Önerilen teknikte tüm imgeye ve alt imgelere ait sayısal imzanın hesaplanmasında simetrik veya asimetrik anahtar yapısı kullanılabilirken, blok ve bloksal bağımlılıklara ait imzaların hesaplanmasında sadece simetrik anahtar yapısının kullanılması zorunlu tutulmuştur. Bu zorunluluğun sebebi, blok ve bloksal bağımlılıklara ait imzaların tüm bitlerini kullanmak yerine bu imzaların sadece bir parçasının kullanılmasının önerilmesinden kaynaklanmaktadır. Bu öneri sunulan tekniğin veri yükünü azaltarak daha küçük boyutlu blokların kullanılmasına imkan sağlamaktadır. Önerilen teknikte 6 adet sayısal imzanın hesaplanmasına ihtiyaç duyulur. Bu imzalar tüm imgeye ait bir sayısal imza, her bir alt imgeye ait birer sayısal imza, bloğa ait bir sayısal imza ve bloğun iki alt blok ile bağımlılığından oluşan birer sayısal imzadan oluşur. Bu imzaların her biri eşitlik 6.1. ile 6.6. arasındaki eşitlikler kullanılarak oluşturulur.

$$S_1 = E(H(\tilde{I}), K_E) \tag{6.1}$$

$$S_2 = E(H(\tilde{A}), K_E) \tag{6.2}$$

$$S_3 = E(H(\tilde{B}), K_E) \tag{6.3}$$

for $i = 1:e$

  for $j = 1:f$

$$S_4 = E\left(H\left(\tilde{A}_{i,j}\right), K\right) \tag{6.4}$$

$$S_5 = E\left(H\left(\tilde{A}_{i,j} + \tilde{B}^2_{i,j-1}\right), K\right) \tag{6.5}$$

$$S_6 = E\left(H\left(\tilde{A}_{i,j} + \tilde{B}^1_{i,j+1}\right), K\right) \tag{6.6}$$

Yukarıdaki eşitliklerde, S_1 tüm imge piksellerinden hesaplanılan sayısal imzayı, S_2 ve S_3 alt imgelere ait sayısal imzaları, S_4 alt imgeler içerisindeki her bir bloğa ait sayısal imzayı, S_5 ve S_6 bloksal bağımlılıklara ait sayısal imzaları göstermektedir. Eşitlik 6.4., 6.5. ve 6.6. A alt imgesindeki bloklar ile bu blokların B alt imgesindeki alt bloklarla yaptığı bağımlılıklara ait sayısal imzaların hesaplanmasında kullanılmaktadır. Benzer şekilde, B alt imgesindeki her bir blok ve bu blokların A alt imgesindeki alt bloklarla yaptığı bağımlılıklara ait sayısal imzalar eşitlik 6.7., 6.8., ve 6.9. kullanılarak hesaplanır.

for $i = 1:e$

  for $j = 1:f$

$$S_4 = E\left(H\left(\tilde{B}_{i,j}\right), K\right) \tag{6.7}$$

$$S_5 = E\left(H\left(\tilde{B}_{i,j} + \tilde{A}^2_{i,j-1}\right), K\right) \tag{6.8}$$

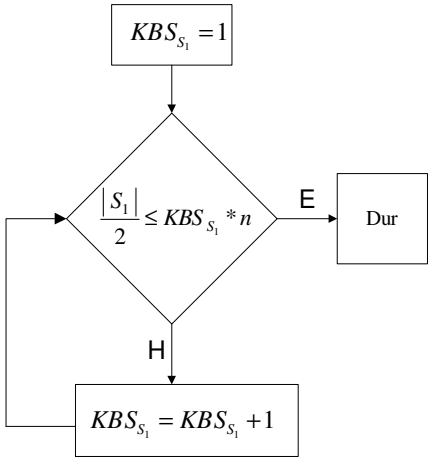
$$S_6 = E\left(H\left(\tilde{B}_{i,j} + \tilde{A}^1_{i,j+1}\right), K\right) \tag{6.9}$$

Dördüncü aşamada, sayısal imzaların kaç bitinin kullanılacağı belirlenir. Önerilen teknikte, çok sayıda sayısal imzanın kullanılması yüksek miktarlı bir veri yükünün oluşmasına neden olur. İmge içerisindeki her bir bloğun yüksek miktarlı bu veri yükünü taşıyabilmesi için blokların oldukça büyük seçilmesi gerekir. Büyük blok seçimi ise değişim bölgesi belirlemedeki doğruluğu azaltır. Önerilen teknikte veri yükü azaltılarak değişim yapılan bölgeyi yüksek bir doğrulukta belirleyebilmek için iki strateji geliştirilmiştir.

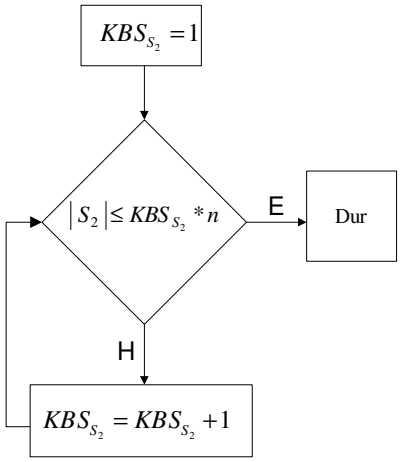
İlk strateji, imgeye ve alt imgelere ait sayısal imzaların bloklar üzerine dağıtılmasından oluşur. Bu stratejide ilk olarak tüm imge piksellerinden hesaplanılan S_1 sayısal imzası iki eşit parçaya ayrılır. S_1 sayısal imzasının ilk parçası ve S_2 sayısal imzası A alt imgesindeki bloklara dağıtılırken, S_1 sayısal imzasının ikinci parçası ve S_3 sayısal imzası B alt

imgesindeki bloklara dağıtılır. Her bir blok içerisine S_1 sayısal imzasına ait kaç bitin gömüleceği şekil 6.3.'de verilen algoritma ile belirlenir. Şekil 6.3.'de KBS_{S_1}, S_1 sayısal imzasının kaç bitinin gömüleceğini belirtirken, $n = e * f$ 'e eşittir ve n her bir alt imge içerisindeki toplam blok sayısını belirtmektedir.

Her bir blok içerisine S_2 ve S_3 sayısal imzasına ait kaç bitin gömüleceği ise şekil 6.4.'de verilen algoritma ile belirlenir. S_2 sayısal imzası için bulunan değer S_3 sayısal imzası için de geçerlidir.



Şekil 6.3. Her blok içerisine S_1 sayısal imzasının kaç bitinin gömüleceğinin belirlenmesi



Şekil 6.4. Her blok içerisine S_2 ve S_3 sayısal imzasının kaç bitinin gömüleceğinin belirlenmesi

Şekil 6.4.'de KBS_{S_2} ve KBS_{S_3}, S_2 ve S_3 sayısal imzalarının kaç bitinin gömüleceğini belirtmektedir. Tüm imgeye ve alt imgelere ait sayısal imzalar saldırılara karşı tam bir güvenlik sağlamak amacıyla kullanılmıştır. Tüm imgeye ait sayısal imza, imge içerisinde bir bozulma olup olmadığını belirlerken, alt imgelere ait sayısal imzalar hangi alt imge içerisinde bir bozulma olduğunu belirler.

İkinci strateji, bloklar ve bu blokların alt bloklar ile oluşturdukları sayısal imzaların bir parçasının kullanılmasından oluşur. Bu strateji veri yükü miktarını oldukça azaltarak, önerilen tekniğin değişim bölgesi belirleme doğruluğunu artırır. Ancak sayısal imzaların bir parçasının kullanılması, bu imzalarda asimetrik anahtar yapısının kullanılmasını engeller. Ayrıca sayısal imzanın bir parçasının kullanılması, güvenlik seviyesinde bir azalmaya neden olabilir. Ancak, önerilen tekniğin asıl güvenliği, tamamıyla imge ve alt imgelere ait sayısal imzalarla sağlandığı için bu kullanım bir güvenlik açığı oluşturmamaktadır. Bloklar ve bu blokların alt bloklar ile oluşturdukları sayısal imzalar hem saldırıları önler hem de değişim bölgesinin daha yüksek bir doğrulukta belirlenmesini sağlar. Blok ve bu blokların alt bloklar ile oluşturdukları sayısal imzaların kaç bitinin kullanılacağı seçilen blok büyüklüğüne $(SBB = r * s)$ bağımlı olarak eşitlik 6.10., 6.11. ve 6.12. kullanılarak hesaplanır.

$$KBS_{S_4} = \left\lfloor \frac{SBB - KBS_{S_1} - KBS_{S_2}}{2} \right\rfloor \tag{6.10}$$

$$KBS_{S_5} = \left\lfloor \frac{SBB - KBS_{S_1} - KBS_{S_2} - KBS_{S_4}}{2} \right\rfloor \tag{6.11}$$

$$KBS_{S_6} = \left\lceil \frac{SBB - KBS_{S_1} - KBS_{S_2} - KBS_{S_4}}{2} \right\rceil \tag{6.12}$$

Burada SBB, her bir alt imge içerisindeki blok büyüklüğünü; KBS_{S_4}, S_4 sayısal imzasının kaç bitinin kullanılacağını; KBS_{S_5}, S_5 sayısal imzasının kaç bitinin kullanılacağını; KBS_{S_6}, S_6 sayısal imzasının kaç bitinin kullanılacağını belirtmektedir.

Son olarak beşinci aşama, veri yükünün oluşturulmasından ve her bir bloğun EÖB düzlemi içerisine gömülmesinden oluşur. Veri yükü, sayısal imzalara ait parçaların birleştirilmesinden oluşur. Her bir sayısal imzaya ait parçacığın kaç bitten oluştuğu dördüncü aşamada belirlenmiştir. Veri yükünün oluşturulması eşitlik 6.13.'de verilmiştir.

```
for  i = 1 : e
  for  j = 1 : f
```

$$P_{i,j} = S_1 \| S_2 \| S_4 \| S_5 \| S_6 \| \tag{6.13}$$

Burada $\|$, sayısal imzalara ait bitlerin birleştirilmesini göstermektedir. Ayrıca B alt imgesine ait veri yükünde S_2 sayısal imzası yerine S_3 sayısal imzası kullanılır. A ve B alt imgesindeki her bir blok için oluşturulan veri yükü bloğun EÖB düzlemine gömülerek imge filigranlanır.

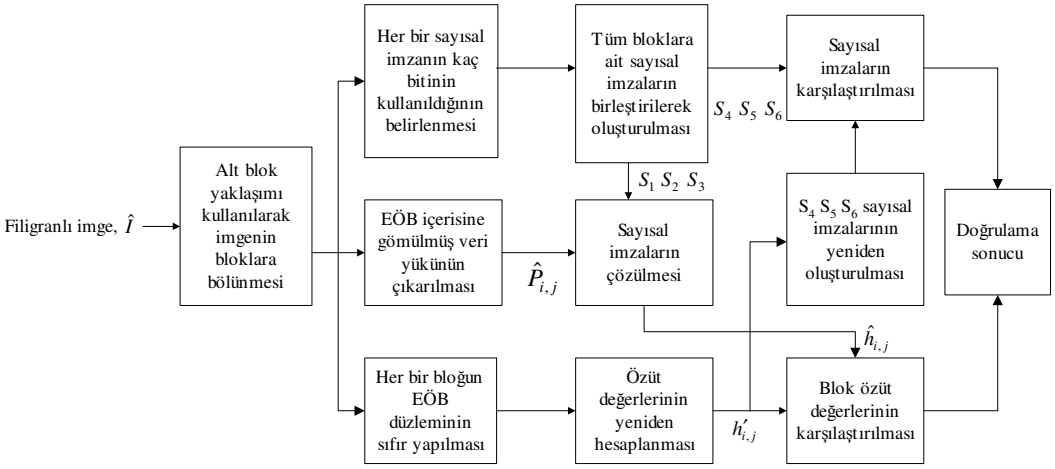
6.2. Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinin Doğrulama Yordamı

Önerilen Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinin doğrulama yordamı 3 temel adımdan oluşmaktadır. İlk adım, filigranlama yordamında olduğu gibi imgenin alt blok yaklaşımı kullanarak bloklara ayrılmasından oluşur. İkinci adımda her bir blok içerisinde sayısal imzaların kaçar bitinin kullanıldığı belirlenerek, bloklar içerisine gömülmüş olan sayısal imzalara ait bitler çıkartılır. Daha sonra bu bitler birleştirilerek sayısal imzalar elde edilir. Üçüncü adımda sayısal imzalar sıra ile çözülerek imgenin aslıyla aynılığının doğrulanması yapılır. Doğrulama yordamında değişim bölgesini yüksek bir doğrulukta belirleyebilmek için bloksal bağımlılıklara ait sayısal imzaları kullanan önemli bir algoritma geliştirilmiştir. Önerilen tekniğin doğrulama yordamı şekil 6.5.'de gösterilmiştir.

Muhtemelen bozulmuş filigranlı imgeyi $\left(\hat{I}\right)$ doğrulamada alt blok yaklaşımı kullanılarak bloklar oluşturulduktan sonra, ilk olarak tüm imgeye ait özüt doğrulanır. Ancak tüm imgeye ait özütün doğrulanabilmesi için, imge üzerine dağılmış S_1 sayısal imzasının birleştirilmesi gerekir. Her bir blok içerisine S_1 sayısal imzasının kaç bitinin gömüldüğü şekil 6.3.'de verilen algoritma kullanılarak bulunur. Daha sonra her bir blok içerisinden S_1 sayısal imzasına ait bitler çıkarılıp birleştirilerek S_1 sayısal imzası eşitlik 6.14. kullanılarak çözülür.

$$\hat{h} = D\left(\hat{S}_1, K_D\right) \tag{6.14}$$

İmge üzerinde bir değişim yapılıp yapılmadığını belirleyebilmek için tüm imgenin özütü tekrar hesaplanır ve $\hat{h}$ özütüyle karşılaştırılır.



Şekil 6.5. Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinin doğrulama yordamı

Eşitlik 6.15. tüm imgeye ait özütün yeniden hesaplanmasını ve eşitlik 6.16. karşılaştırma işlemini tanımlamaktadır.

$$h' = H\left(\tilde{\hat{I}}\right) \tag{6.15}$$

$$\text{Doğrulama} = \begin{cases} \text{doğru}, & if\ h' = \hat{h} \\ \text{yanlış}, & if\ h' \neq \hat{h} \end{cases} \tag{6.16}$$

Doğrulama işlemi bir bozulma olduğunu gösterirse alt imgelerin doğrulanması yapılır. Her bir alt imgenin doğrulanabilmesi için alt imgelere ait sayısal imzalar çözülmesi gerekir. Ancak alt imgelere ait özütlerin doğrulanabilmesi için, alt imgeler üzerine dağılmış S_2 ve S_3 sayısal imzalarının birleştirilmesi gerekir. Her bir blok içerisine S_2 ve S_3 sayısal imzalarının kaç bitinin gömüldüğü şekil 6.4.'de verilen algoritma kullanılarak bulunur. Daha sonra her bir blok içerisinden S_2 ve S_3 sayısal imzalarına ait bitler çıkartılıp birleştirilerek S_2 ve S_3 sayısal imzası eşitlik 6.17. ve 6.18. kullanılarak çözülür.

$$\hat{h}_A = D\left(\hat{S}_2, K_D\right) \tag{6.17}$$

$$\hat{h}_B = D\left(\hat{S}_3, K_D\right) \tag{6.18}$$

Alt imgeler üzerinde bir değişim yapılıp yapılmadığını belirleyebilmek için alt imge özütleri tekrar hesaplanır ve $\hat{h}_A$, $\hat{h}_B$ özütleriyle karşılaştırılır. Eşitlik 6.19. ve 6.20. alt imgelere ait özütlerin yeniden hesaplanmasını ve eşitlik 6.21. karşılaştırma işlemini tanımlamaktadır.

$$h'_A = H\left(\tilde{\tilde{A}}\right) \tag{6.19}$$

$$h'_B = H\left(\tilde{\tilde{B}}\right) \tag{6.20}$$

$$\text{Doğrulama} = \begin{cases} \text{doğru, } if\ \ h'_A = \hat{h}_A \\ \text{yanlış, } if\ h'_A \neq \hat{h}_A \end{cases}$$

$$\text{Doğrulama} = \begin{cases} \text{doğru, } if\ \ h'_B = \hat{h}_B \\ \text{yanlış, } if\ h'_B \neq \hat{h}_B \end{cases} \tag{6.21}$$

Alt imgelerin herhangi birinde bir bozulma olduğu belirlenirse, değişim yapılan bölgeyi tespit edebilmek için bloklara ve bağımlılıklara ait sayısal imzalar karşılaştırılır. Ancak filigranlama yordamında bloklara ve bağımlılıklara ait sayısal imzaların sadece bir parçası gömüldüğü için, doğrulama sayısal imza parçacıkları karşılaştırılarak yapılır. A ve B alt imgelerindeki bloklara ait sayısal imzanın yeniden hesaplanması eşitlik 6.22. ve 6.23. ile verilmiştir. Hesaplanılan sayısal imzaların kaç bitinin kullanılacağı eşitlik 6.10. kullanılarak bulunur. Blok imzalarının karşılaştırılması eşitlik 6.24.'de verilmiştir.

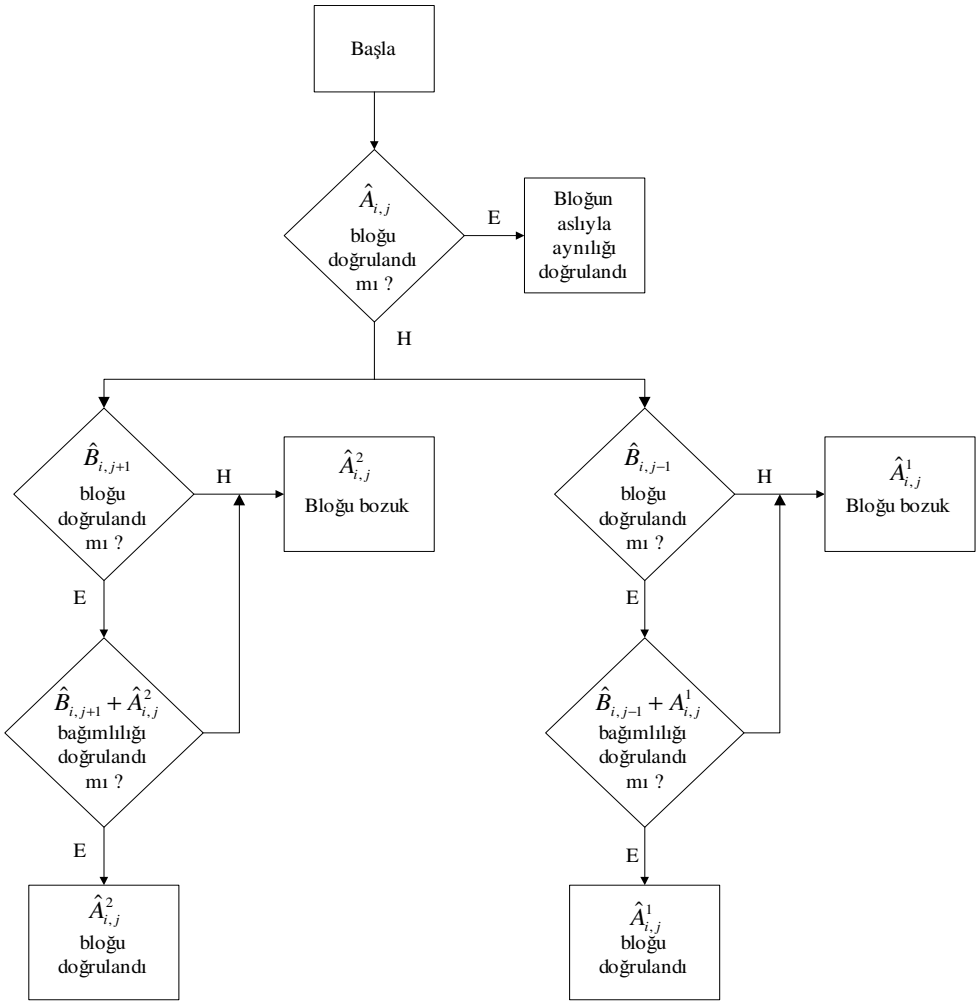
$$S'_4 = E\left(H\left(\tilde{\tilde{A}}_{i,j}\right), K\right) \tag{6.22}$$

$$S'_4 = E\left(H\left(\tilde{\tilde{B}}_{i,j}\right), K\right) \tag{6.23}$$

$$\text{Doğrulama} = \begin{cases} \text{doğru,} \quad if\ \ S'_4 = \hat{S}_4 \\ \text{yanlış,} \quad if\ \ S'_4 \neq \hat{S}_4 \end{cases} \tag{6.24}$$

Bloğun hangi alt bloğunun bozulduğunu belirlemek için bloksal bağımlılığa ait sayısal imzaların doğrulanması yapılır. Ancak bloklara ait sayısal imzaların kullanımında olduğu gibi, bloksal bağımlılığa ait sayısal imzaların da sadece bir parçası kullanıldığı için, doğrulama sayısal imzalara ait parçacıklar karşılaştırılarak yapılır. Bloksal bağımlılığa ait sayısal imzaların kaç bitinin kullanılacağı eşitlik 6.11. ve 6.12. kullanılarak belirlenir.

Şekil 6.6.'da bozulan bir bloğun alt bloklarını doğrulamak için kullanılan algoritma verilmiştir. Bu şekilde, A alt imgesi içerisindeki bir bloğun bozulduğu kabul edilmiştir. Eğer bozulan blok $\hat{A}_{i,j}$ ise, ilk olarak B alt imgesi içerisinde $\hat{A}_{i,j}$ bloğunun alt bloklarını kendisine komşu seçen $\hat{B}_{i,j+1}$ ve $\hat{B}_{i,j-1}$ bloklarının doğrulanması gerekir. $\hat{B}_{i,j+1}$ bloğu doğrulanmazsa, $\hat{A}^2_{i,j}$ alt bloğu da doğrulanamayacağı için bu blok bozuk olarak kabul edilir. Ancak $\hat{B}_{i,j+1}$ bloğu doğrulanırsa, $\hat{B}_{i,j+1} + \hat{A}^2_{i,j}$ bağımlılığına ait sayısal imzanın doğrulanması yapılır. $\hat{B}_{i,j+1} + \hat{A}^2_{i,j}$ bağımlılığının doğrulanması, $\hat{A}^2_{i,j}$ alt bloğunu doğrular. $\hat{B}_{i,j+1} + \hat{A}^2_{i,j}$ bağımlılığının doğrulanamaması durumunda, $\hat{A}^2_{i,j}$ alt bloğunun bozuk olduğu anlaşılır.



Şekil 6.6. Alt blokların doğrulanması için kullanılan algoritma

Benzer şekilde $\hat{B}_{i,j-1}$ bloğu doğrulanır. $\hat{B}_{i,j-1}$ bloğu doğrulanmazsa, $\hat{A}^1_{i,j}$ alt bloğu da doğrulanamayacağı için bu blok da bozuk olarak kabul edilir. Ancak $\hat{B}_{i,j-1}$ bloğu doğrulanırsa, $\hat{B}_{i,j-1} + A^1_{i,j}$ bağımlılığına ait sayısal imzanın doğrulanması yapılır. $\hat{B}_{i,j-1} + A^1_{i,j}$ bağımlılığının doğrulanması, $\hat{A}^1_{i,j}$ alt bloğunu doğrular. $\hat{B}_{i,j-1} + A^1_{i,j}$ bağımlılığının doğrulanamaması durumunda, $\hat{A}^1_{i,j}$ alt bloğunun bozuk olduğu anlaşılır.

6.3. Deneysel Çalışmalar

Bu bölüm içerisinde gerek önerilen tekniğe gerekse karşılaştırma yapılan HBC2 Filigranlama Tekniğine ait filigranlama ve doğrulama algoritmaları MATLAB ortamında yazılan programlar ile gerçekleştirilmiştir.

Deneysel çalışmalar verilmeden önce önerilen tekniği daha iyi açıklayabilmek için, 32*32 piksellik küçük boyutlu bir imge üzerinde ve sadece sayısal imza olarak MD5 algoritmasına dayanan 64 bitlik MAC kullanılarak basit bir uygulama yapılmıştır. Bu uygulamada ilk olarak şekil 6.7.'de sayısal verileri gösterilen test imgesi önerilen teknik ile filigranlanmış ve daha sonra bir blok üzerinde bir değişim yapılarak önerilen tekniğin doğrulama yordamı çalıştırılmıştır.

Filigranlama yordamının ilk aşamasında imgenin EÖB düzlemi sıfır yapılır ve alt blok yaklaşımı kullanılarak 16*32 piksellik iki alt imgeye, her bir alt imge 8*8 piksellik bloklara ve her bir blok 8*4 piksellik alt bloklara bölünür. Bölmeleme işlemi şekil 6.8., 6.9. ve 6.10.'da verilmiştir.

İkinci aşamada her bir alt imgedeki blok diğer alt imgedeki iki blok ile bağımlı hale getirilir. Şekil 6.11.'de $A_{1,2}$ bloğuna ait sayısal veriler ve $A_{1,2}$ bloğunun B alt imgesindeki $B^2_{1,1}$ ve $B^1_{1,3}$ ile oluşturdukları bağımlılıklara ait sayısal veriler gösterilmiştir.

Üçüncü aşamada tüm imgeye alt imgelere, bloklara ve kurulan bağımlılıklara ait özüt ve sayısal imza değerleri aşağıda verilmiştir:

$h_1 = 5050e65dc1cea0d8de30880b2080d941$

$h_2 = 55c86625400a330989466b744e47465b$

$h_3 = 08cc9da9c0cfd7b1f2167e2e509afc4e$

$h_4 = 4b0f518a31fec2ae46d26a6608701f26$

$h_5 = \text{df4b2eaf6277822c20cd0a4516978df8}$

$h_6 = \text{6297a6dfb1adebebc95293f9dfab8ff9}$

$S_1 = \text{648052eea294a2fe}$

$S_2 = \text{20352e9dd2ce1b9b}$

$S_3 = \text{067baefd1fab5743}$

$S_4 = \text{c1de7d551388a541}$

$S_5 = \text{2cda110372f87994}$

$S_6 = \text{24eb04d06e1fc47c}$

Burada h_1 ve S_1, tüm imgeye ait özütü ve sayısal imzayı; h_2 ve S_2, A alt imgesine ait özütü ve sayısal imzayı; h_3 ve S_3, B alt imgesine ait özütü ve sayısal imzayı; h_4 ve S_4, $A_{1,2}$ bloğuna ait özütü ve sayısal imzayı, h_5 ve S_5, $A_{1,2}$ ile $B_{1,1}^2$ bağımlılığına ait özütü ve sayısal imzayı; h_6 ve S_6 $A_{1,2}$ ile $B_{1,3}^1$ bağımlılığına ait özütü ve sayısal imzayı göstermektedir.

Dördüncü aşamada şekil 6.3.'de verilen algoritma kullanılarak her bir blok içerisine S_1 sayısal imzasının 4 $\left(KBS_{S_1} = 4\right)$ bitinin gömüleceği hesaplanır. Şekil 6.4.'de verilen algoritma kullanılarak her bir blok içerisine S_2 ve S_3 sayısal imzasının 8 bitinin $\left(KBS_{S_2} = 8, KBS_{S_3} = 8\right)$ gömüleceği belirlenir. S_4, S_5 ve S_6 sayısal imzalarının kaç bitinin gömüleceği ise eşitlik 6.10., 6.11. ve 6.12. kullanılarak $KBS_{S_4} = 26$, $KBS_{S_5} = 13$ ve $KBS_{S_6} = 13$ olarak bulunur.

Son olarak sayısal imza parçaları birleştirilerek veri yükü oluşturulur ve her bir bloğun EÖB düzlemine gömülür. $A_{1,2}$ bloğu içerisine gömülecek veri yükü, S_1 sayısal imzasının 5-8 arasındaki bitleri, S_2 sayısal imzasının 9-16 arasındaki bitleri, S_4 sayısal imzasının 1-26 arasındaki bitleri, S_5 sayısal imzasının 1-13 arasındaki bitleri ve S_6 sayısal imzasının 1-13 arasındaki bitleri birleştirilerek oluşturulur. Oluşturulan $P_{1,2}$ aşağıda verilmiştir.

$P_{1,2} = \text{0100 0011 0101 1100 0001 1101 1110 0111 1101 01 0010 1100 1101 1 0010 0100 1110 1}$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	136	136	137	137	139	134	129	134	133	137	126	128	130	129	129	126	126	127	132	134	142	142	147	152	148	142	144	134	117	126	97	85
2	136	136	137	137	139	134	129	134	133	137	126	128	130	129	129	126	126	127	132	134	142	142	147	152	148	142	144	134	117	126	97	85
3	136	136	137	137	139	134	129	134	133	137	126	128	130	129	129	126	126	127	132	134	142	142	147	152	148	142	144	134	117	126	97	85
4	133	140	133	131	133	128	128	130	126	128	132	127	129	125	122	127	133	128	132	140	145	145	146	148	150	144	140	132	119	106	86	72
5	128	132	133	129	138	132	135	128	127	138	132	127	133	124	124	131	129	126	135	144	149	152	147	148	151	143	139	132	116	106	87	75
6	130	131	130	126	132	130	133	131	128	126	133	131	125	127	127	123	129	129	135	143	147	143	149	151	144	138	138	131	112	101	97	79
7	130	131	130	132	135	132	128	128	129	137	130	130	124	126	125	123	129	140	136	143	148	148	147	141	137	145	145	133	110	106	91	73
8	131	133	128	133	127	122	128	128	128	127	124	129	125	124	127	128	135	142	144	147	145	146	145	147	144	141	133	131	112	101	89	78
9	128	128	130	130	127	125	131	128	124	126	129	126	127	122	122	139	136	140	150	141	142	146	142	142	144	142	129	131	112	108	95	71
10	130	133	130	126	131	125	133	137	133	136	128	127	126	127	135	134	142	144	147	140	146	143	144	137	146	137	135	123	110	101	91	74
11	131	128	127	127	131	131	131	134	130	128	124	133	126	128	132	134	137	143	142	144	149	139	137	141	135	137	129	127	116	102	99	99
12	132	129	126	130	128	124	129	132	133	130	132	130	120	127	129	135	145	144	142	136	141	145	143	136	134	133	131	122	125	105	100	90
13	129	131	124	133	133	127	130	132	133	126	132	134	124	132	135	139	143	140	148	139	141	141	142	128	132	133	131	126	120	108	92	80
14	125	124	126	127	131	127	134	133	131	128	131	126	127	136	135	138	139	141	146	143	137	135	138	134	135	135	135	120	119	108	86	74
15	131	130	137	131	129	128	132	131	131	130	134	134	131	139	139	146	146	145	142	141	133	132	134	133	131	135	131	125	118	107	88	77
16	130	131	133	129	130	131	133	139	134	132	132	137	137	144	144	146	144	141	141	141	134	134	132	132	130	138	137	129	113	105	99	82
17	129	129	133	127	130	132	132	129	134	137	130	135	145	149	146	144	148	143	147	139	132	132	130	130	133	133	132	129	126	110	100	74
18	135	136	131	130	136	138	134	134	136	134	138	142	144	140	144	144	143	142	148	135	136	134	131	132	135	134	136	125	117	100	88	75
19	132	131	130	136	132	130	137	137	142	140	139	146	142	141	144	147	140	144	136	134	128	128	131	133	135	134	131	128	118	101	100	70
20	132	135	139	131	135	132	131	136	139	138	139	144	144	139	143	137	138	139	137	137	130	129	137	135	136	134	138	123	111	110	89	77
21	139	143	140	130	136	139	134	135	141	139	147	140	139	138	133	138	140	142	137	127	132	131	135	135	135	136	130	126	116	101	87	65
22	133	130	136	131	128	132	135	141	146	147	143	137	134	141	129	132	132	132	131	138	135	129	136	134	131	136	133	124	111	103	91	69
23	136	135	137	139	134	135	136	145	152	145	142	131	129	133	128	128	125	128	128	136	137	136	130	132	138	135	135	124	119	98	89	71
24	140	132	139	137	134	136	139	143	142	140	141	126	125	128	124	124	124	123	132	136	142	140	135	130	138	134	132	126	122	109	89	70
25	138	135	136	131	134	138	146	147	150	139	133	123	120	117	119	124	118	126	135	139	138	140	137	129	139	138	135	125	123	104	98	73
26	135	137	131	133	131	142	145	151	144	135	132	121	116	113	111	116	112	124	134	138	139	132	137	134	139	139	134	130	119	99	94	75
27	133	134	130	137	130	144	148	148	139	132	126	113	112	106	115	102	117	124	136	138	141	136	137	133	140	135	139	124	117	104	90	74
28	132	134	131	135	141	141	147	141	134	128	117	113	95	91	99	105	113	128	131	137	139	140	137	137	135	136	133	132	114	105	86	75
29	133	138	131	136	143	149	146	139	130	123	114	102	92	90	91	107	117	127	133	142	143	139	138	142	141	139	135	136	113	104	96	72
30	133	138	139	136	142	151	146	135	127	123	111	90	83	83	88	108	123	127	134	138	140	144	136	137	140	140	132	126	121	100	91	70
31	132	141	146	146	146	143	137	128	121	110	92	78	73	78	91	104	121	130	136	145	138	144	141	140	134	140	133	131	119	109	96	71
32	134	138	147	148	148	139	130	120	115	100	80	64	64	79	95	109	124	132	138	141	139	136	142	137	139	139	138	129	118	101	91	78

Şekil 6.7. 32*32 piksellik test imgesinin sayısal verileri

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	136	136	136	136	138	134	128	134	132	136	126	128	130	128	128	126	126	126	132	134	142	142	146	152	148	142	144	134	116	126	96	84
2	136	136	136	136	138	134	128	134	132	136	126	128	130	128	128	126	126	126	132	134	142	142	146	152	148	142	144	134	116	126	96	84
3	136	136	136	136	138	134	128	134	132	136	126	128	130	128	128	126	126	126	132	134	142	142	146	152	148	142	144	134	116	126	96	84
4	132	140	132	130	132	128	128	130	126	128	132	126	128	124	122	126	132	128	132	140	144	144	146	148	150	144	140	132	118	106	86	72
5	128	132	132	128	138	132	134	128	126	138	132	126	132	124	124	130	128	126	134	144	148	152	146	148	150	142	138	132	116	106	86	74
6	130	130	130	126	132	130	132	130	128	126	132	130	124	126	126	122	128	128	134	142	146	142	148	150	144	138	138	130	112	100	96	78
7	130	130	130	132	134	132	128	128	128	136	130	130	124	126	124	122	128	140	136	142	148	148	146	140	136	144	144	132	110	106	90	72
8	130	132	128	132	126	122	128	128	128	126	124	128	124	124	126	128	134	142	144	146	144	146	144	146	144	140	132	130	112	100	88	78
9	128	128	130	130	126	124	130	128	124	126	128	126	126	122	122	138	136	140	150	140	142	146	142	142	144	142	128	130	112	108	94	70
10	130	132	130	126	130	124	132	136	132	136	128	126	126	126	134	134	142	144	146	140	146	142	144	136	146	136	134	122	110	100	90	74
11	130	128	126	126	130	130	130	134	130	128	124	132	126	128	132	134	136	142	142	144	148	138	136	140	134	136	128	126	116	102	98	98
12	132	128	126	130	128	124	128	132	132	130	132	130	120	126	128	134	144	144	142	136	140	144	142	136	134	132	130	122	124	104	100	90
13	128	130	124	132	132	126	130	132	132	126	132	134	124	132	134	138	142	140	148	138	140	140	142	128	132	132	130	126	120	108	92	80
14	124	124	126	126	130	126	134	132	130	128	130	126	126	136	134	138	138	140	146	142	136	134	138	134	134	134	134	120	118	108	86	74
15	130	130	136	130	128	128	132	130	130	130	134	134	130	138	138	146	146	144	142	140	132	132	134	132	130	134	130	124	118	106	88	76
16	130	130	132	128	130	130	132	138	134	132	132	136	136	144	144	146	144	140	140	140	134	134	132	132	130	138	136	128	112	104	98	82

(a)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	128	128	132	126	130	132	132	128	134	136	130	134	144	148	146	144	148	142	146	138	132	132	130	130	132	132	132	128	126	110	100	74
2	134	136	130	130	136	138	134	134	136	134	138	142	144	140	144	144	142	142	148	134	136	134	130	132	134	134	136	124	116	100	88	74
3	132	130	130	136	132	130	136	136	142	140	138	146	142	140	144	146	140	144	136	134	128	128	130	132	134	134	130	128	118	100	100	70
4	132	134	138	130	134	132	130	136	138	138	138	144	144	138	142	136	138	138	136	136	130	128	136	134	136	134	138	122	110	110	88	76
5	138	142	140	130	136	138	134	134	140	138	146	140	138	138	132	138	140	142	136	126	132	130	134	134	134	136	130	126	116	100	86	64
6	132	130	136	130	128	132	134	140	146	146	142	136	134	140	128	132	132	132	130	138	134	128	136	134	130	136	132	124	110	102	90	68
7	136	134	136	138	134	134	136	144	152	144	142	130	128	132	128	128	124	128	128	136	136	136	130	132	138	134	134	124	118	98	88	70
8	140	132	138	136	134	136	138	142	142	140	140	126	124	128	124	124	124	122	132	136	142	140	134	130	138	134	132	126	122	108	88	70
9	138	134	136	130	134	138	146	146	150	138	132	122	120	116	118	124	118	126	134	138	138	140	136	128	138	138	134	124	122	104	98	72
10	134	136	130	132	130	142	144	150	144	134	132	120	116	112	110	116	112	124	134	138	138	132	136	134	138	138	134	130	118	98	94	74
11	132	134	130	136	130	144	148	148	138	132	126	112	112	106	114	102	116	124	136	138	140	136	136	132	140	134	138	124	116	104	90	74
12	132	134	130	134	140	140	146	140	134	128	116	112	94	90	98	104	112	128	130	136	138	140	136	136	134	136	132	132	114	104	86	74
13	132	138	130	136	142	148	146	138	130	122	114	102	92	90	90	106	116	126	132	142	142	138	138	142	140	138	134	136	112	104	96	72
14	132	138	138	136	142	150	146	134	126	122	110	90	82	82	88	108	122	126	134	138	140	144	136	136	140	140	132	126	120	100	90	70
15	132	140	146	146	146	142	136	128	120	110	92	78	72	78	90	104	120	130	136	144	138	144	140	140	134	140	132	130	118	108	96	70
16	134	138	146	148	148	138	130	120	114	100	80	64	64	78	94	108	124	132	138	140	138	136	142	136	138	138	138	128	118	100	90	78

(b)

Şekil 6.8. 16*32 piksellik alt imgeler **(a)** A alt imgesi **(b)** B alt imgesi

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	136	136	136	136	138	134	128	134	132	136	126	128	130	128	128	126	126	126	132	134	142	142	146	152	148	142	144	134	116	126	96	84
2	136	136	136	136	138	134	128	134	132	136	126	128	130	128	128	126	126	126	132	134	142	142	146	152	148	142	144	134	116	126	96	84
3	136	136	136	136	138	134	128	134	132	136	126	128	130	128	128	126	126	126	132	134	142	142	146	152	148	142	144	134	116	126	96	84
4	132	140	132	130	132	128	128	130	126	128	132	126	128	124	122	126	132	128	132	140	144	144	146	148	150	144	140	132	118	106	86	72
5	128	132	132	128	138	132	134	128	126	138	132	126	132	124	124	130	128	126	134	144	148	152	146	148	150	142	138	132	116	106	86	74
6	130	130	130	126	132	130	132	130	128	126	132	130	124	126	126	122	128	128	134	142	146	142	148	150	144	138	138	130	112	100	96	78
7	130	130	130	132	134	132	128	128	128	136	130	130	124	126	124	122	128	140	136	142	148	148	146	140	136	144	144	132	110	106	90	72
8	130	132	128	132	126	122	128	128	128	126	124	128	124	124	126	128	134	142	144	146	144	146	144	146	144	140	132	130	112	100	88	78
9	128	128	130	130	126	124	130	128	124	126	128	126	126	122	122	138	136	140	150	140	142	146	142	142	144	142	128	130	112	108	94	70
10	130	132	130	126	130	124	132	136	132	136	128	126	126	126	134	134	142	144	146	140	146	142	144	136	146	136	134	122	110	100	90	74
11	130	128	126	126	130	130	130	134	130	128	124	132	126	128	132	134	136	142	142	144	148	138	136	140	134	136	128	126	116	102	98	98
12	132	128	126	130	128	124	128	132	132	130	132	130	120	126	128	134	144	144	142	136	140	144	142	136	134	132	130	122	124	104	100	90
13	128	130	124	132	132	126	130	132	132	126	132	134	124	132	134	138	142	140	148	138	140	140	142	128	132	132	130	126	120	108	92	80
14	124	124	126	126	130	126	134	132	130	128	130	126	126	136	134	138	138	140	146	142	136	134	138	134	134	134	134	120	118	108	86	74
15	130	130	136	130	128	128	132	130	130	130	134	134	130	138	138	146	146	144	142	140	132	132	134	132	130	134	130	124	118	106	88	76
16	130	130	132	128	130	130	132	138	134	132	132	136	136	144	144	146	144	140	140	140	134	134	132	132	130	138	136	128	112	104	98	82

(a)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	128	128	132	126	130	132	132	128	134	136	130	134	144	148	146	144	148	142	146	138	132	132	130	130	132	132	132	128	126	110	100	74
2	134	136	130	130	136	138	134	134	136	134	138	142	144	140	144	144	142	142	148	134	136	134	130	132	134	134	136	124	116	100	88	74
3	132	130	130	136	132	130	136	136	142	140	138	146	142	140	144	146	140	144	136	134	128	128	130	132	134	134	130	128	118	100	100	70
4	132	134	138	130	134	132	130	136	138	138	138	144	144	138	142	136	138	138	136	136	130	128	136	134	136	134	138	122	110	110	88	76
5	138	142	140	130	136	138	134	134	140	138	146	140	138	138	132	138	140	142	136	126	132	130	134	134	134	136	130	126	116	100	86	64
6	132	130	136	130	128	132	134	140	146	146	142	136	134	140	128	132	132	132	130	138	134	128	136	134	130	136	132	124	110	102	90	68
7	136	134	136	138	134	134	136	144	152	144	142	130	128	132	128	128	124	128	128	136	136	136	130	132	138	134	134	124	118	98	88	70
8	140	132	138	136	134	136	138	142	142	140	140	126	124	128	124	124	124	122	132	136	142	140	134	130	138	134	132	126	122	108	88	70
9	138	134	136	130	134	138	146	146	150	138	132	122	120	116	118	124	118	126	134	138	138	140	136	128	138	138	134	124	122	104	98	72
10	134	136	130	132	130	142	144	150	144	134	132	120	116	112	110	116	112	124	134	138	138	132	136	134	138	138	134	130	118	98	94	74
11	132	134	130	136	130	144	148	148	138	132	126	112	112	106	114	102	116	124	136	138	140	136	136	132	140	134	138	124	116	104	90	74
12	132	134	130	134	140	140	146	140	134	128	116	112	94	90	98	104	112	128	130	136	138	140	136	136	134	136	132	132	114	104	86	74
13	132	138	130	136	142	148	146	138	130	122	114	102	92	90	90	106	116	126	132	142	142	138	138	142	140	138	134	136	112	104	96	72
14	132	138	138	136	142	150	146	134	126	122	110	90	82	82	88	108	122	126	134	138	140	144	136	136	140	140	132	126	120	100	90	70
15	132	140	146	146	146	142	136	128	120	110	92	78	72	78	90	104	120	130	136	144	138	144	140	140	134	140	132	130	118	108	96	70
16	134	138	146	148	148	138	130	120	114	100	80	64	64	78	94	108	124	132	138	140	138	136	142	136	138	138	138	128	118	100	90	78

(b)

Şekil 6.9. A ve B alt imgelerinin 8*8 piksellik bloklara bölünmesi

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	136	136	136	136	138	134	128	134	132	136	126	128	130	128	128	126	126	126	132	134	142	142	146	152	148	142	144	134	116	126	96	84
2	136	136	136	136	138	134	128	134	132	136	126	128	130	128	128	126	126	126	132	134	142	142	146	152	148	142	144	134	116	126	96	84
3	136	136	136	136	138	134	128	134	132	136	126	128	130	128	128	126	126	126	132	134	142	142	146	152	148	142	144	134	116	126	96	84
4	132	140	132	130	132	128	128	130	126	128	132	126	128	124	122	126	132	128	132	140	144	144	146	148	150	144	140	132	118	106	86	72
5	128	132	132	128	138	132	134	128	126	138	132	126	132	124	124	130	128	126	134	144	148	152	146	148	150	142	138	132	116	106	86	74
6	130	130	130	126	132	130	132	130	128	126	132	130	124	126	126	122	128	128	134	142	146	142	148	150	144	138	138	130	112	100	96	78
7	130	130	130	132	134	132	128	128	128	136	130	130	124	126	124	122	128	140	136	142	148	148	146	140	136	144	144	132	110	106	90	72
8	130	132	128	132	126	122	128	128	128	126	124	128	124	124	126	128	134	142	144	146	144	146	144	146	144	140	132	130	112	100	88	78
9	128	128	130	130	126	124	130	128	124	126	128	126	126	122	122	138	136	140	150	140	142	146	142	142	144	142	128	130	112	108	94	70
10	130	132	130	126	130	124	132	136	132	136	128	126	126	126	134	134	142	144	146	140	146	142	144	136	146	136	134	122	110	100	90	74
11	130	128	126	126	130	130	130	134	130	128	124	132	126	128	132	134	136	142	142	144	148	138	136	140	134	136	128	126	116	102	98	98
12	132	128	126	130	128	124	128	132	132	130	132	130	120	126	128	134	144	144	142	136	140	144	142	136	134	132	130	122	124	104	100	90
13	128	130	124	132	132	126	130	132	132	126	132	134	124	132	134	138	142	140	148	138	140	140	142	128	132	132	130	126	120	108	92	80
14	124	124	126	126	130	126	134	132	130	128	130	126	126	136	134	138	138	140	146	142	136	134	138	134	134	134	134	120	118	108	86	74
15	130	130	136	130	128	128	132	130	130	130	134	134	130	138	138	146	146	144	142	140	132	132	134	132	130	134	130	124	118	106	88	76
16	130	130	132	128	130	130	132	138	134	132	132	136	136	144	144	146	144	140	140	140	134	134	132	132	130	138	136	128	112	104	98	82

(a)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	128	128	132	126	130	132	132	128	134	136	130	134	144	148	146	144	148	142	146	138	132	132	130	130	132	132	132	128	126	110	100	74
2	134	136	130	130	136	138	134	134	136	134	138	142	144	140	144	144	142	142	148	134	136	134	130	132	134	134	136	124	116	100	88	74
3	132	130	130	136	132	130	136	136	142	140	138	146	142	140	144	146	140	144	136	134	128	128	130	132	134	134	130	128	118	100	100	70
4	132	134	138	130	134	132	130	136	138	138	138	144	144	138	142	136	138	138	136	136	130	128	136	134	136	134	138	122	110	110	88	76
5	138	142	140	130	136	138	134	134	140	138	146	140	138	138	132	138	140	142	136	126	132	130	134	134	134	136	130	126	116	100	86	64
6	132	130	136	130	128	132	134	140	146	146	142	136	134	140	128	132	132	132	130	138	134	128	136	134	130	136	132	124	110	102	90	68
7	136	134	136	138	134	134	136	144	152	144	142	130	128	132	128	128	124	128	128	136	136	136	130	132	138	134	134	124	118	98	88	70
8	140	132	138	136	134	136	138	142	142	140	140	126	124	128	124	124	124	122	132	136	142	140	134	130	138	134	132	126	122	108	88	70
9	138	134	136	130	134	138	146	146	150	138	132	122	120	116	118	124	118	126	134	138	138	140	136	128	138	138	134	124	122	104	98	72
10	134	136	130	132	130	142	144	150	144	134	132	120	116	112	110	116	112	124	134	138	138	132	136	134	138	138	134	130	118	98	94	74
11	132	134	130	136	130	144	148	148	138	132	126	112	112	106	114	102	116	124	136	138	140	136	136	132	140	134	138	124	116	104	90	74
12	132	134	130	134	140	140	146	140	134	128	116	112	94	90	98	104	112	128	130	136	138	140	136	136	134	136	132	132	114	104	86	74
13	132	138	130	136	142	148	146	138	130	122	114	102	92	90	90	106	116	126	132	142	142	138	138	142	140	138	134	136	112	104	96	72
14	132	138	138	136	142	150	146	134	126	122	110	90	82	82	88	108	122	126	134	138	140	144	136	136	140	140	132	126	120	100	90	70
15	132	140	146	146	146	142	136	128	120	110	92	78	72	78	90	104	120	130	136	144	138	144	140	140	134	140	132	130	118	108	96	70
16	134	138	146	148	148	138	130	120	114	100	80	64	64	78	94	108	124	132	138	140	138	136	142	136	138	138	138	128	118	100	90	78

(b)

Şekil 6.10. Blokların 8*4 piksellik alt bloklara bölünmesi

	1	2	3	4	5	6	7	8
1	132	136	126	128	130	128	128	126
2	132	136	126	128	130	128	128	126
3	132	136	126	128	130	128	128	126
4	126	128	132	126	128	124	122	126
5	126	138	132	126	132	124	124	130
6	128	126	132	130	124	126	126	122
7	128	136	130	130	124	126	124	122
8	128	126	124	128	124	124	126	128

(a)

	1	2	3	4	5	6	7	8	9	10	11	12
1	130	132	132	128	132	136	126	128	130	128	128	126
2	136	138	134	134	132	136	126	128	130	128	128	126
3	132	130	136	136	132	136	126	128	130	128	128	126
4	134	132	130	136	126	128	132	126	128	124	122	126
5	136	138	134	134	126	138	132	126	132	124	124	130
6	128	132	134	140	128	126	132	130	124	126	126	122
7	134	134	136	144	128	136	130	130	124	126	124	122
8	134	136	138	142	128	126	124	128	124	124	126	128

(b)

	1	2	3	4	5	6	7	8	9	10	11	12
1	132	136	126	128	130	128	128	126	148	142	146	138
2	132	136	126	128	130	128	128	126	142	142	148	134
3	132	136	126	128	130	128	128	126	140	144	136	134
4	126	128	132	126	128	124	122	126	138	138	136	136
5	126	138	132	126	132	124	124	130	140	142	136	126
6	128	126	132	130	124	126	126	122	132	132	130	138
7	128	136	130	130	124	126	124	122	124	128	128	136
8	128	126	124	128	124	124	126	128	124	122	132	136

(c)

Şekil 6.11. Blok ve bağımlı blokların oluşturduğu sayısal veriler **(a)** $A_{1,2}$ bloğuna ait sayısal veriler **(b)** $A_{1,2}$ ile $B_{1,1}^{2}$ bağımlılığına ait sayısal veriler **(c)** $A_{1,2}$ ile $B_{1,3}^{1}$ bağımlılığına ait sayısal veriler

Son olarak oluşturulan $P_{1,2}$ veri yükü $A_{1,2}$ bloğunun EÖB düzlemine gömülerek filigranlı imge bloğu oluşturulur. $A_{1,2}$ filigranlı imge bloğu şekil 6.12.'de verilmiştir. Tüm imge blokları için aynı işlemler yapılarak filigranlı imge oluşturulur

	1	2	3	4	5	6	7	8
1	132	137	126	128	130	128	129	127
2	132	137	126	129	131	121	128	126
3	132	136	126	129	131	129	128	127
4	127	129	133	126	128	125	123	127
5	127	139	132	127	132	125	124	130
6	129	126	133	131	124	126	127	123
7	128	137	131	130	124	127	124	122
8	129	126	124	129	125	125	126	129

Şekil 6.12. Filigranlı $A_{1,2}$ bloğu

	1	2	3	4	5	6	7	8
1	**131**	137	126	128	130	128	129	127
2	132	137	126	129	131	121	128	126
3	132	136	126	129	131	129	128	127
4	127	129	133	127	128	125	123	127
5	127	139	132	127	132	125	124	130
6	129	126	133	131	124	126	127	123
7	128	137	131	130	124	127	124	122
8	129	126	124	129	125	125	126	129

Şekil 6.13. Değiştirilmiş filigranlı $A_{1,2}$ bloğu

Filigranlı imge üzerinde $A_{1,2}$ bloğunun (1,1) konumundaki pikselinin değerini 131 olarak değiştirerek, önerilen tekniğin doğrulama yordamını çalıştıralım. Değiştirilmiş filigranlı blok şekil 6.13.'de verilmiştir.

Doğrulama yordamının ilk aşamasında imge, filigranlama yordamında olduğu gibi alt blok yaklaşımı kullanılarak bloklara bölünür. İkinci aşamada her bir blok içerisinden veri yükleri çıkartılır ve her bir sayısal imzanın kaç bitinin kullanıldığı filigranlama yordamındaki gibi belirlenerek sayısal imzalar oluşturulur.

S_1 sayısal imzası çözülerek elde edilen özüt yeniden hesaplanan imge özütüyle karşılaştırıldığında özütler eşleşmediği için, S_2 ve S_3 sayısal imzalarının özütleri çözülerek elde edilen özütler ile alt imgelerin yeniden hesaplanan özütleri karşılaştırılmıştır. B alt imgesine ait özütler eşleşirken, A alt imgesine ait özüt eşleşmemiştir. Bu yüzden A alt imgesindeki her bir bloğa ait S_4 sayısal imzası ile her bir bloğun yeniden hesaplanan sayısal imzası karşılaştırılmıştır. Bu karşılaştırmanın sonucunda $A_{1,2}$ bloğuna ait sayısal imzalar eşleşmemiştir. $A_{1,2}$ bloğunun hangi alt yarısının bozulduğunu belirleyebilmek için ilk olarak B alt imgesindeki $B_{1,1}$ ve $B_{1,3}$ bloklarına ait sayısal imzaların karşılaştırılması gerekir. Bu

karşılaştırmanın sonucunda $B_{1,1}$ ve $B_{1,3}$ bloklarının sayısal imzaları eşleştiği için bloklar doğrulanmıştır. Son olarak $\left(B_{1,1}, A_{1,2}^{1}\right)$ bağımlılığına ve $\left(B_{1,3}, A_{1,2}^{2}\right)$ bağımlılığına ait sayısal imzaların karşılaştırılması yapılır. Bu karşılaştırmanın sonucunda $\left(B_{1,1}, A_{1,2}^{1}\right)$ bağımlılığına ait sayısal imzalar eşleşmediği için $A_{1,2}^{1}$ bloğunun tüm pikselleri sıfır yapılırken, $\left(B_{1,3}, A_{1,2}^{2}\right)$ bağımlılığına ait sayısal imzalar eşleştiği için $A_{1,2}^{2}$ bloğu doğrulanır. Doğrulanmış $A_{1,2}$ bloğu şekil 6.14.'de gösterilmiştir.

	1	2	3	4	5	6	7	8
1	0	0	0	0	130	128	129	127
2	0	0	0	0	131	121	128	126
3	0	0	0	0	131	129	128	127
4	0	0	0	0	128	125	123	127
5	0	0	0	0	132	125	124	130
6	0	0	0	0	124	126	127	123
7	0	0	0	0	124	127	124	122
8	0	0	0	0	125	125	126	129

Şekil 6.14. Önerilen tekniğinin doğrulama yordamının $A_{1,2}$ bloğuna ait sayısal verileri

6.3.1. Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinin Değişim Bölgesi Belirleme Doğruluğu

Önerilen Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinin etkinliğini değerlendirebilmek amacıyla 5 farklı imge çözünürlüğünde (128*192, 240*320, 480*640, 768*1024, 1200*1600) ve 19 farklı test imgesi üzerinde deneysel çalışmalar yapılmıştır. Yapılan deneysel çalışmalarda tüm imgeye ve alt imgelere ait sayısal imzaların hesaplanmasında SHA-1 algoritmasına dayanan 320 bitlik DSA; bloklara ve bağımlılıklara ait sayısal imzaların hesaplanmasında MD5 algoritmasına dayanan 64 bitlik MAC kullanılmıştır.

Önerilen teknik ile şekil 6.6.'da gösterilen 1200*1600 çözünürlüğe sahip test imgesinin filigranlanması ayrıntılı olarak aşağıda verilmiştir. İlk aşamada imgenin EÖB düzlemi sıfır yapılmış ve alt blok yaklaşımı kullanılarak imge 600*800 piksellik iki alt imgeye, her bir alt imge 8*8 piksellik bloklara ve her bir blok 8*4 piksellik alt bloklara bölünmüştür. İkinci aşamada her bir alt imge içerisindeki bloklar, diğer alt imgedeki alt bloklar ile bağımlı hale getirilmiştir. Üçüncü aşamada 320 bitlik DSA sayısal imza algoritması kullanılarak tüm imgeye ve alt imgelere ait sayısal imzalar; 64 bitlik MAC algoritması kullanılarak da bloklar ve bağımlılıklara ait sayısal imzalar hesaplanmıştır. Dördüncü aşamada şekil 6.3. ve 6.4.'de verilen

algoritmalar kullanılarak $KBS_{S_1} = 1$, $KBS_{S_2} = 1$ ve $KBS_{S_3} = 1$; eşitlik 6.10., 6.11. ve 6.12. kullanılarak $KBS_{S_4} = 31$, $KBS_{S_5} = 15$ ve $KBS_{S_6} = 16$ olarak hesaplanmıştır. Son aşamada ise sayısal imzalara ait parçalar 4. aşamada belirlenen sayılara göre birleştirilerek veri yükü oluşturulmuş ve her bir bloğun EÖB düzlemine gömülerek filigranlı imge oluşturulmuştur. Filigranlı imge şekil 6.15.'de gösterilmiştir.



Şekil 6.15. Önerilen teknik ile filigranlanmış imge (çözünürlük=1200*1600 piksel)

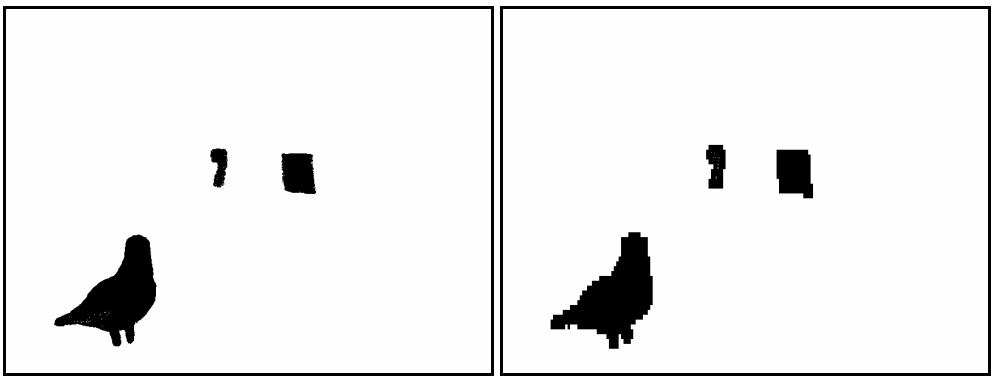


Şekil 6.16. Değiştirilmiş filigranlı imge (çözünürlük=1200*1600 piksel)



Şekil 6.17. Önerilen tekniğin doğrulama sonucu (çözünürlük=1200*1600 piksel)

Önerilen tekniğin 1200*1600 piksellik imgelerin doğrulanmasındaki performansını değerlendirebilmek için, şekil 6.15.'de verilen filigranlı imge üzerinde değişimler yapılmıştır. Şekil 6.16. değiştirilmiş filigranlı imgeyi göstermektedir. Şekil 6.16. incelendiğinde, sahile bir "martı" eklenmiş, katamaran üzerindeki "7" rakamı ve "logo" silinmiştir. Değiştirilmiş bölgeler şekil 6.16.'da oklarla gösterilmiştir. Değiştirilmiş filigranlı imgeyi doğrulamak için önerilen tekniğin doğrulama yordamı çalıştırılmış ve elde edilen sonuç şekil 6.17.'de verilmiştir.



(a) **(b)**

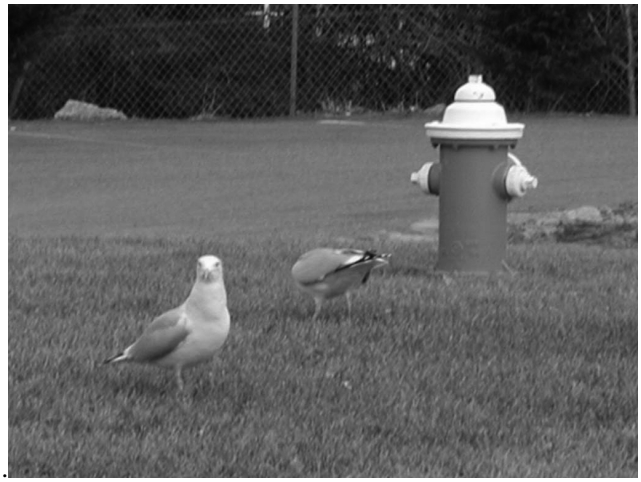
Şekil 6.18. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması (çözünürlük=1200*1600 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu

Şekil 6.17.'de doğrulanamayan bölgeler siyah ile gösterilmiştir. Doğrulanamayan piksel sayısı ise 21.120 olarak bulunmuştur.

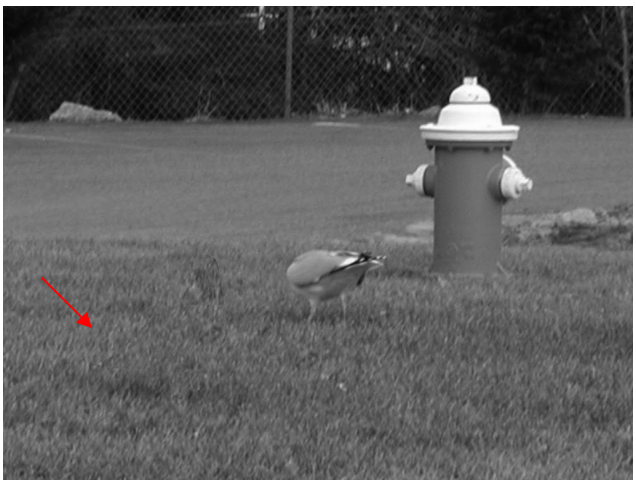
Önerilen tekniğin doğrulama sonucuyla, imge üzerinde yapılan gerçek değişimi karşılaştırabilmek için, şekil 6.18.'de gerçek değişim ve önerilen tekniğin doğrulama sonucu gösterilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Gerçekte değiştirilen piksel sayısı ise 18.000'dir.

Önerilen tekniğin değişim bölgesi belirleme doğruluğunu değerlendirebilmek için, ikinci çalışma 768*1024 piksellik çözünürlükteki bir test imgesi üzerinde yapılmıştır. İlk aşamada imgenin EÖB düzlemi sıfır yapılmış ve alt blok yaklaşımı kullanılarak imge 384*512 piksellik iki alt imgeye, her bir alt imge 8*8 piksellik bloklara ve her bir blok 8*4 piksellik alt bloklara bölünmüştür. İkinci aşamada her bir alt imge içerisindeki bloklar, diğer alt imgedeki alt bloklar ile bağımlı hale getirilmiştir. Üçüncü aşamada 320 bitlik DSA sayısal imza algoritması kullanılarak tüm imgeye ve alt imgelere ait sayısal imzalar; 64 bitlik MAC algoritması kullanılarak da bloklar ve bağımlılıklara ait sayısal imzalar hesaplanmıştır. Dördüncü aşamada şekil 6.3. ve 6.4.'de verilen algoritmalar kullanılarak $KBS_{S_1} = 1$, $KBS_{S_2} = 1$ ve $KBS_{S_3} = 1$; eşitlik 6.10., 6.11. ve 6.12. kullanılarak $KBS_{S_4} = 31$, $KBS_{S_5} = 15$ ve $KBS_{S_6} = 16$ olarak hesaplanmıştır. Son aşamada ise sayısal imzalara ait parçalar 4. aşamada belirlenen sayılara göre birleştirilerek veri yükü oluşturulmuş ve her bir bloğun EÖB düzlemine gömülerek filigranlı imge oluşturulmuştur. Filigranlı imge şekil 6.19.'da gösterilmiştir.

Önerilen tekniğin 768*1024 piksellik imgelerin doğrulanmasındaki performansını değerlendirebilmek için, şekil 6.19.'da verilen filigranlı imge üzerinde değişimler yapılmıştır. Şekil 6.20. değiştirilmiş filigranlı imgeyi göstermektedir.



Şekil 6.19. Önerilen teknik ile filigranlanmış imge (çözünürlük=768*1024 piksel)



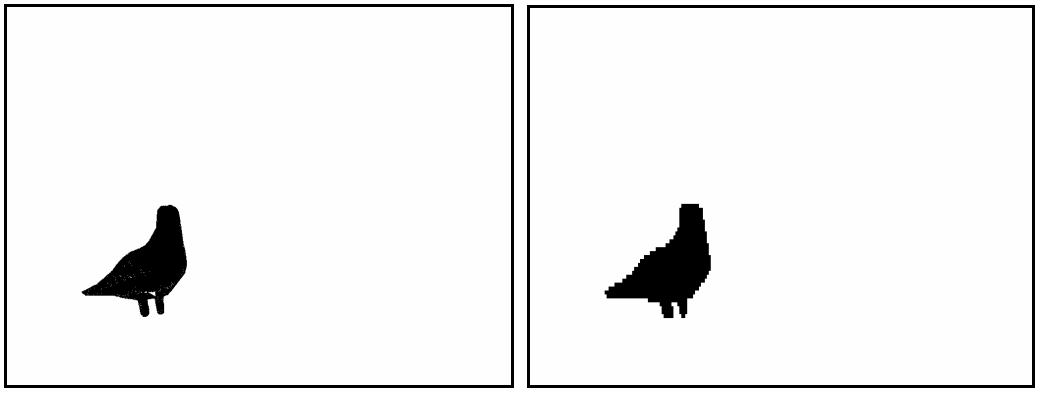
Şekil 6.20. Değiştirilmiş filigranlı imge (çözünürlük=768*1024 piksel)



Şekil 6.21. Önerilen tekniğin doğrulama sonucu (çözünürlük=768*1024 piksel)

Değiştirilmiş bölgeler şekil 6.20.'de oklarla gösterilmiştir. Değiştirilmiş filigranlı imgeyi doğrulamak için önerilen tekniğin doğrulama yordamı çalıştırılmış ve elde edilen sonuç şekil 6.21.'de verilmiştir. Şekil 6.21.'de doğrulanamayan bölgeler siyah ile gösterilmiştir. Doğrulanamayan piksel sayısı ise 23.104 olarak bulunmuştur.

Önerilen tekniğin doğrulama sonucuyla, imge üzerinde yapılan gerçek değişimi karşılaştırabilmek için, şekil 6.22.'de gerçek değişim ve önerilen tekniğin doğrulama sonucu gösterilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Gerçekte değiştirilen piksel sayısı ise 20.459'dur.



(a) **(b)**

Şekil 6.22. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması (çözünürlük=768*1024 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu

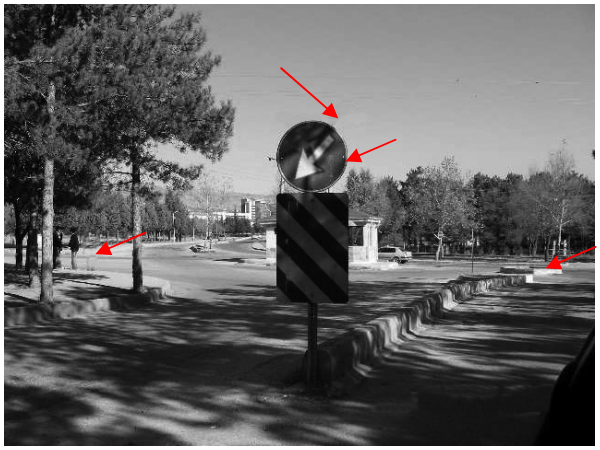
Önerilen tekniğin değişim bölgesi belirleme doğruluğunu değerlendirebilmek için, üçüncü çalışma 480*640 piksellik çözünürlükteki bir test imgesi üzerinde yapılmıştır. İlk aşamada imgenin EÖB düzlemi sıfır yapılmış ve alt blok yaklaşımı kullanılarak imge 240*320 piksellik iki alt imgeye, her bir alt imge 8*8 piksellik bloklara ve her bir blok 8*4 piksellik alt bloklara bölünmüştür. İkinci aşamada her bir alt imge içerisindeki bloklar, diğer alt imgedeki alt bloklar ile bağımlı hale getirilmiştir. Üçüncü aşamada 320 bitlik DSA sayısal imza algoritması kullanılarak tüm imgeye ve alt imgelere ait sayısal imzalar; 64 bitlik MAC algoritması kullanılarak da bloklar ve bağımlılıklara ait sayısal imzalar hesaplanmıştır. Dördüncü aşamada şekil 6.3. ve 6.4.'de verilen algoritmalar kullanılarak $KBS_{S_1} = 1$, $KBS_{S_2} = 1$ ve $KBS_{S_3} = 1$; eşitlik 6.10., 6.11. ve 6.12. kullanılarak $KBS_{S_4} = 31$, $KBS_{S_5} = 15$ ve $KBS_{S_6} = 16$ olarak hesaplanmıştır. Son aşamada ise sayısal imzalara ait parçalar 4. aşamada belirlenen sayılara göre birleştirilerek veri yükü oluşturulmuş ve her bir bloğun EÖB düzlemine gömülerek filigranlı imge oluşturulmuştur. Filigranlı imge şekil 6.23.'de gösterilmiştir.

Önerilen tekniğin 480*640 piksellik imgelerin doğrulanmasındaki performansını değerlendirebilmek için, şekil 6.23.'de verilen filigranlı imge üzerinde değişimler yapılmıştır. Şekil 6.24. değiştirilmiş filigranlı imgeyi göstermektedir.

Değiştirilmiş bölgeler şekil 6.24.'de oklarla gösterilmiştir. Değiştirilmiş filigranlı imgeyi doğrulamak için önerilen tekniğin doğrulama yordamı çalıştırılmış ve elde edilen sonuç şekil 6.25.'de verilmiştir. Şekil 6.25.'de doğrulanamayan bölgeler siyah ile gösterilmiştir. Doğrulanamayan piksel sayısı ise 7.392 olarak bulunmuştur.



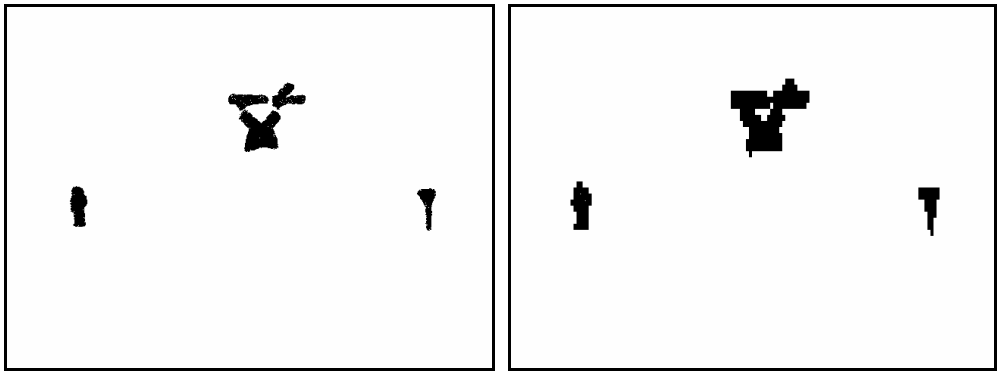
Şekil 6.23. Önerilen teknik ile filigranlanmış imge (çözünürlük=480*640 piksel)



Şekil 6.24. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)



Şekil 6.25. Önerilen tekniğin doğrulama sonucu (çözünürlük=480*640 piksel)



(a) **(b)**

Şekil 6.26. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması (çözünürlük=480*640 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu

Önerilen tekniğin doğrulama sonucuyla, imge üzerinde yapılan gerçek değişimi karşılaştırabilmek için, şekil 6.26.'da gerçek değişim ve önerilen tekniğin doğrulama sonucu gösterilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Gerçekte değiştirilen piksel sayısı ise 4.878'dir.

6.3.2. Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinin Saldırılara Karşı Güvenirliği

Önerilen tekniğin; Kes Yapıştır, Vektör Nicemlemeli Taklit Saldırısı (VNTS), Doğum Günü Saldırısı ve Aktarıma Dayalı Saldırı'ya karşı güvenirliği aşağıda yapılan uygulamalar ile gösterilmiştir.

6.3.2.1. Kes Yapıştır Saldırısına Karşı Güvenirlik

Önerilen tekniğin Kes Yapıştır Saldırısı'na karşı güvenirliğini değerlendirebilmek için, 240*320 piksel çözünürlükteki bir test imgesi üzerinde çalışma yapılmıştır. İlk aşamada imgenin EÖB düzlemi sıfır yapılmış ve alt blok yaklaşımı kullanılarak imge 120*160 piksellik iki alt imgeye, her bir alt imge 8*8 piksellik bloklara ve her bir blok 8*4 piksellik alt bloklara bölünmüştür. İkinci aşamada her bir alt imge içerisindeki bloklar, diğer alt imgedeki alt bloklar ile bağımlı hale getirilmiştir. Üçüncü aşamada 320 bitlik DSA sayısal imza algoritması kullanılarak tüm imgeye ve alt imgelere ait sayısal imzalar; 64 bitlik MAC algoritması kullanılarak da bloklar ve bağımlılıklara ait sayısal imzalar hesaplanmıştır. Dördüncü aşamada şekil 6.3. ve 6.4.'de verilen algoritmalar kullanılarak $KBS_{S_1} = 1$, $KBS_{S_2} = 2$ ve $KBS_{S_3} = 2$;

eşitlik 6.10., 6.11. ve 6.12. kullanılarak $KBS_{S_4} = 30$, $KBS_{S_5} = 15$ ve $KBS_{S_6} = 16$ olarak hesaplanmıştır. Son aşamada ise sayısal imzalara ait parçalar 4. aşamada belirlenen sayılara göre birleştirilerek veri yükü oluşturulmuş ve her bir bloğun EÖB düzlemine gömülerek filigranlı imge oluşturulmuştur. Filigranlı imge şekil 6.27.'de gösterilmiştir.

Önerilen tekniğin Kes Yapıştır Saldırısı'na karşı güvenirliğini değerlendirebilmek için şekil 6.27.'de verilen filigranlı imge üzerinde Kes Yapıştır Saldırısı yapılmıştır. Bu saldırı, imge içerisindeki 41 - 48 satırları ve 169 - 176 sütunları (41:48, 169:176) arasındaki 8*8 piksellik bloğun kesilerek maskelenmek istenen blokların (41:72, 177:208 piksel aralığındaki bloklar) üzerine yapıştırılmasıyla yapılmıştır. Saldırı yapılmış imge şekil 6.28.'de gösterilmiştir.



Şekil 6.27. Önerilen teknik ile filigranlanmış imge (çözünürlük=240*320 piksel)



Şekil 6.28. Önerilen teknik ile filigranlanmış imgenin kes yapıştır saldırısı ile değiştirilmesi



Şekil 6.29. Önerilen tekniğin doğrulama sonucu (çözünürlük=240*320 piksel)

Saldırı yapılmış filigranlı imgenin doğrulanması için önerilen tekniğin doğrulama algoritması çalıştırılmıştır. Doğrulama algoritmasında tüm imgeye ve A alt imgesine ait özütler eşleşmediği için bloklara ve bağımlılıklara ait sayısal imzaların karşılaştırılması yapılmıştır. Bu karşılaştırma sonucunda tüm bloklara ait sayısal imzalar doğrulanmış, fakat bağımlılıklara ait sayısal imzaların eşleşmemesi saldırı yapılan bölgenin belirlenmesini sağlamıştır. Önerilen tekniğin doğrulama algoritmasının çıkışı şekil 6.29.'da verilmiştir.

6.3.2.2. Vektör Nicemlemeli Taklit Saldırısına Karşı Güvenirlik

Önerilen tekniğin VNTS'ye karşı güvenirliğini değerlendirebilmek için, 128*192 piksel çözünürlükte ve VNTS kullanılarak oluşturulmuş bir test imgesi üzerinde çalışma yapılmıştır. Oluşturulan sahte test imgesindeki 8*8 piksellik her bir blok, önerilen teknik ile filigranlanmış imge veri tabanındaki benzer bir 8*8 piksellik bloğa nicemlenerek oluşturulmuştur. Oluşturulan imge şekil 6.30.'da gösterilmiştir.



Şekil 6.30. Vektör Nicemlemeli Taklit Saldırısıyla oluşturulmuş 128*192 piksellik imge



Şekil 6.31. VNTS'ye karşı önerilen tekniğin doğrulama sonucu

Doğrulama algoritmasında tüm imgeye, A ve B alt imgelerine ait özütler eşleşmediği için bloklara ve bağımlılıklara ait sayısal imzaların karşılaştırılması yapılmıştır. Bu karşılaştırma sonucunda tüm bloklara ait sayısal imzalar doğrulanmış, fakat bağımlılıklara ait sayısal imzaların eşleşmemesi saldırı yapılan bölgenin belirlenmesini sağlamıştır. Önerilen tekniğin doğrulama algoritmasının çıkışı şekil 6.31. ile gösterilmiştir.

6.3.2.3. Doğum Günü Saldırısına Karşı Güvenirlik

Önerilen tekniğin Doğum Günü Saldırısı'na karşı güvenirliği S_2 sayısal imza algoritmasında kullanılan özüt fonksiyonunun uzunluğuna bağımlıdır. Yukarıda yapılan deneysel çalışmalarda özüt fonksiyonu olarak SHA algoritması kullanıldığından dolayı, herhangi bir blok üzerinde Doğum Günü Saldırısının yapılabilmesi için 2^{80} adet bloğa ihtiyaç duyulmaktadır. Bir bloğun Doğum Günü Saldırısıyla değiştirilmesi durumunda bu değişim, değişimi barındıran alt imgeye ait özüt ve tüm imgeye ait özüt tarafından doğrulanmayacağı için bu saldırının yapılması mümkün değildir.

6.3.2.4. Aktarıma Dayalı Saldırıya Karşı Güvenirlik

Önerilen teknik üzerinde Aktarıma Dayalı Saldırı'nın yapılabilmesi ise tüm imgeye ve alt imgelere ait sayısal imzaların kullanılmasından dolayı mümkün değildir. Çünkü bloklar üzerinde bu saldırı yapıldığı takdirde, tüm imgeye ve alt imgelere ait sayısal imzalar doğrulanmayarak yapılan saldırı belirlenecektir.

6.3.3. Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniği ile HBC2 Filigranlama Tekniğinin Karşılaştırılması

Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniği, HBC2 Filigranlama Tekniği ile aynı filigran niteliklerine sahip olduğu için karşılaştırılmıştır. Bu nitelikler arasında filigran olarak sayısal imzaların kullanılması, filigranın imgenin EÖB düzlemine gömülmesi, filigranlama yordamında bağımlı blok yapısının kullanılması verilebilir. HBC2 Filigranlama Tekniği, Schnorr Sayısal İmza Algoritmasını kullanmaktadır [42]. Önerilen teknikte kullanılan DSA Sayısal İmza Algoritması ile aynı kriptografik güvenlik seviyesini sağlamak için, yapılan deneysel çalışmalarda 288 bitlik Schnorr Sayısal İmzası kullanılmıştır. HBC2 Filigranlama Tekniği'nin değişim bölgesi belirleme doğruluğunu değerlendirebilmek için bölüm 6.3.1.'de verilen 1200*1600, 768*1024 ve 480*640 piksellik çözünürlüğe sahip aynı test imgeleri kullanılmıştır.

HBC2 Filigranlama Tekniği ile imgeleri filigranlamak için, ilk olarak imgelerin EÖB düzlemi sıfır yapılır ve imgeler üst üste gelmeyen bloklara bölünür. Deneysel çalışmalarda blok büyüklüğü 288 biti barındırabilen en küçük blok büyüklüğü seçilerek yapılmıştır. İlk deneysel çalışma 1200*1600 piksellik test imgesi üzerinde yapılmıştır. Her bir blok büyüklüğü ise 12*25 piksel olarak seçilmiştir. HBC2 Filigranlama Tekniği ile filigranlanmış 1200*1600 piksellik test imgesi şekil 6.32.'de gösterilmiştir.

Önerilen teknik ile HBC2 Filigranlama Tekniği'nin değişim bölgesi doğrulama niteliklerini karşılaştırabilmek için şekil 6.15.'e uygulanan aynı değişimler şekil 6.32.'deki imgeye de uygulanmıştır. Değiştirilmiş filigranlı imge şekil 6.33.'de verilmiştir.



Şekil 6.32. HBC2 Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=1200*1600 piksel)

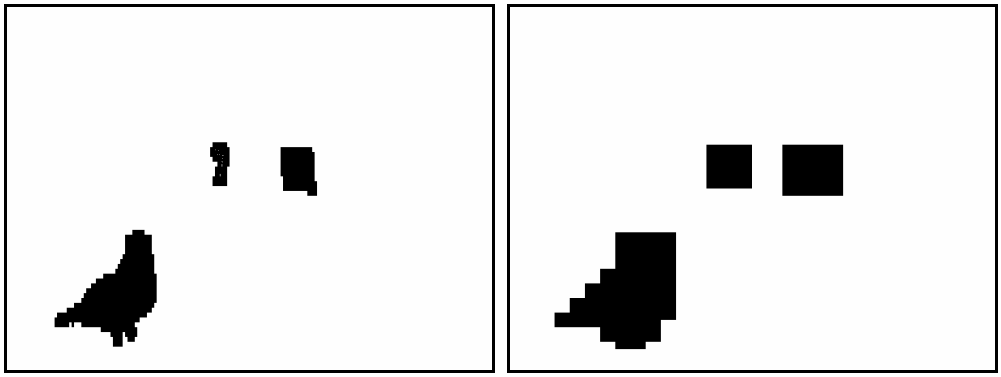


Şekil 6.33. Değiştirilmiş filigranlı imge (çözünürlük=1200*1600 piksel)



Şekil 6.34. HBC2 Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=1200*1600 piksel)

Değiştirilmiş filigranlı imgenin doğrulanması için HBC2 Filigranlama Tekniği'nin doğrulama algoritması çalıştırılmış ve alınan sonuç şekil 6.34.'de gösterilmiştir. Şekil 6.34.'de doğrulanamayan bölgeler siyah ile gösterilmiştir. HBC2 Filigranlama Tekniği tarafından doğrulanamayan piksel sayısı 38.100 olarak bulunmuştur.



(a) **(b)**

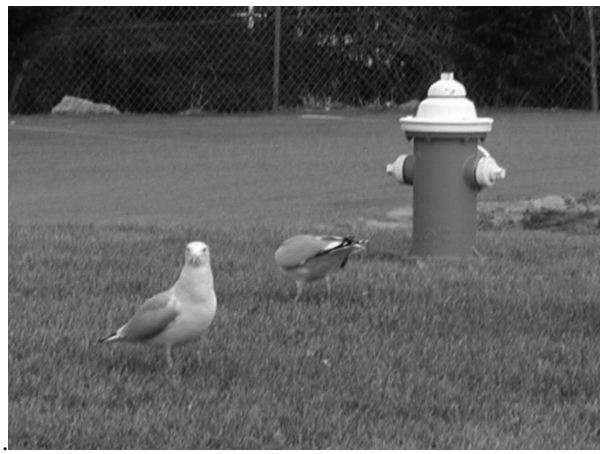
Şekil 6.35. Tekniklerin doğrulama sonuçlarının karşılaştırılması (çözünürlük=1200*1600)
(a) Önerilen tekniğin doğrulama sonucu **(b)** HBC2 Filigranlama Tekniğinin doğrulama sonucu

Önerilen tekniğin doğrulama sonucuyla HBC2 Filigranlama Tekniği'nin doğrulama sonuçlarını karşılaştırabilmek için şekil 6.35.'de her iki tekniğin doğrulama sonuçları birlikte verilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Aynı kriptografik güvenlik seviyesinde önerilen teknik tarafından doğrulanamayan piksel sayısı 21.120 iken, HBC2 Filigranlama Tekniği tarafından doğrulanamayan piksel 38.100 olarak bulunmuştur.

İkinci deneysel çalışma 768*1024 piksellik test imgesi üzerinde yapılmıştır. Önerilen teknik ile aynı kriptografik güvenlik seviyesini sağlamak için bu deneysel çalışmada da 288 bitlik Schnorr sayısal imza algoritması kullanılmıştır. Her bir blok büyüklüğü ise 24*32 piksel olarak seçilmiştir. HBC2 Filigranlama Tekniği ile filigranlanmış 768*1024 piksellik test imgesi şekil 6.36.'da gösterilmiştir. Önerilen teknik ile HBC2 Filigranlama Tekniği'nin değişim bölgesi doğrulama niteliklerini karşılaştırabilmek için şekil 6.19.'a uygulanan aynı değişimler şekil 6.36.'daki imgeye de uygulanmıştır. Değiştirilmiş filigranlı imge şekil 6.37.'de verilmiştir.

Değiştirilmiş filigranlı imgenin doğrulanması için HBC2 Filigranlama Tekniği'nin doğrulama algoritması çalıştırılmış ve alınan sonuç şekil 6.38.'de gösterilmiştir. Şekil 6.38.'de doğrulanamayan bölgeler siyah ile gösterilmiştir. HBC2 Filigranlama Tekniği tarafından doğrulanamayan piksel sayısı 34.176 olarak bulunmuştur.

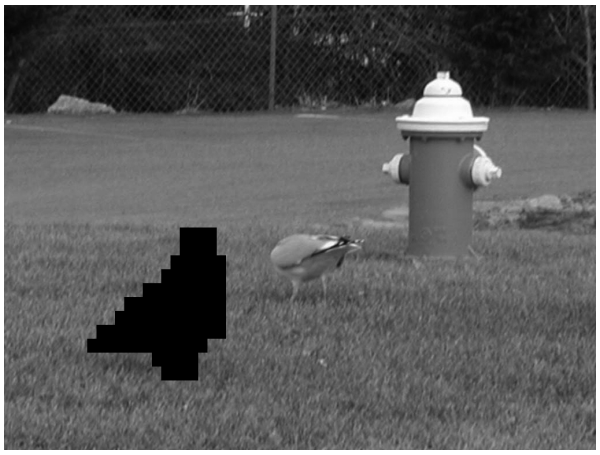
Önerilen tekniğin doğrulama sonucuyla HBC2 Filigranlama Tekniği'nin doğrulama sonuçlarını karşılaştırabilmek için şekil 6.39.'da her iki tekniğin doğrulama sonuçları birlikte verilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Aynı kriptografik güvenlik seviyesinde önerilen teknik tarafından doğrulanamayan piksel sayısı 23.104 iken, HBC2 Filigranlama Tekniği tarafından doğrulanamayan piksel 34.176 olarak bulunmuştur.



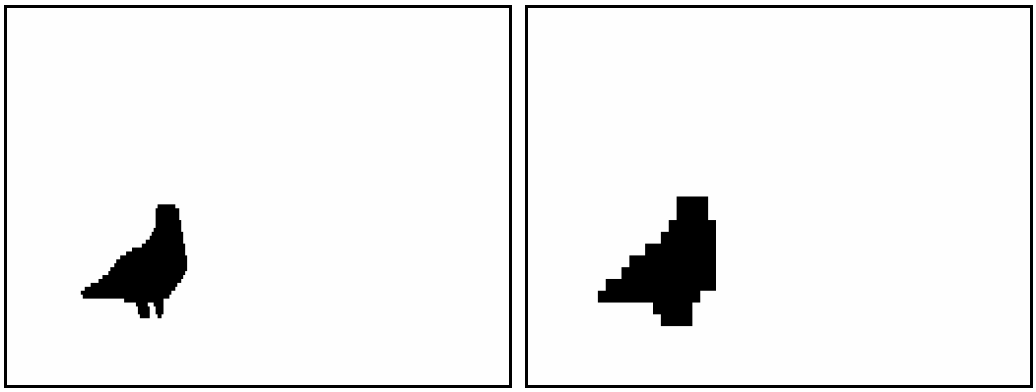
Şekil 6.36. HBC2 Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=768*1024 piksel)



Şekil 6.37. Değiştirilmiş filigranlı imge (çözünürlük=768*1024 piksel)



Şekil 6.38. HBC2 Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=768*1024 piksel)



(a) **(b)**

Şekil 6.39. Tekniklerin doğrulama sonuçlarının karşılaştırılması (çözünürlük=768*1024 piksel) **(a)** Önerilen tekniğin doğrulama sonucu **(b)** HBC2 Filigranlama Tekniğinin doğrulama sonucu

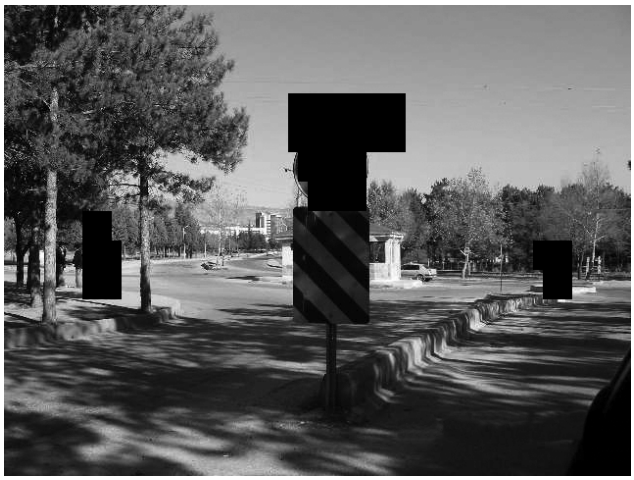
Üçüncü deneysel çalışma 480*640 piksellik test imgesi üzerinde yapılmıştır. Önerilen teknik ile aynı kriptografik güvenlik seviyesini sağlamak için bu deneysel çalışmada da 288 bitlik Schnorr sayısal imza algoritması kullanılmıştır. Her bir blok büyüklüğü ise 30*10 piksel olarak seçilmiştir. HBC2 Filigranlama Tekniği ile filigranlanmış 480*640 piksellik test imgesi şekil 6.40.'da gösterilmiştir. Önerilen teknik ile HBC2 Filigranlama Tekniği'nin değişim bölgesi doğrulama niteliklerini karşılaştırabilmek için şekil 6.23.'e uygulanan aynı değişimler şekil 6.40.'daki imgeye de uygulanmıştır. Değiştirilmiş filigranlı imge şekil 6.41.'de verilmiştir.



Şekil 6.40. HBC2 Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=480*640 piksel)



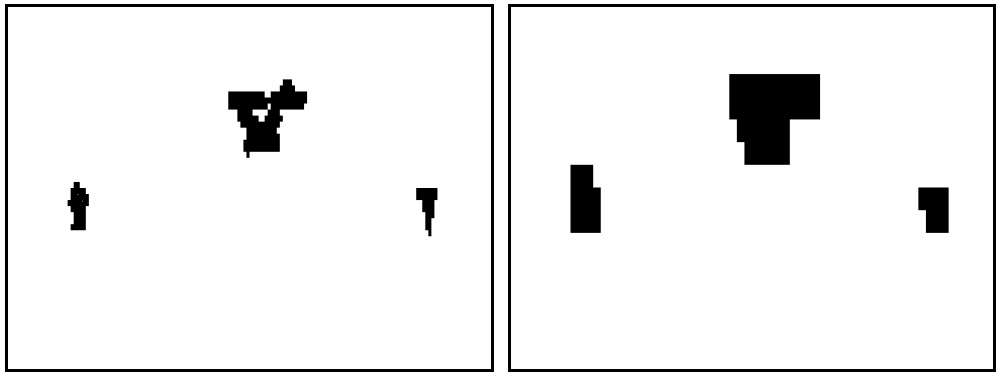
Şekil 6.41. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)



Şekil 6.42. HBC2 Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=480*640 piksel)

Değiştirilmiş filigranlı imgenin doğrulanması için HBC2 Filigranlama Tekniği'nin doğrulama algoritması çalıştırılmış ve alınan sonuç şekil 6.42.'de gösterilmiştir. Şekil 6.42.'de doğrulanamayan bölgeler siyah ile gösterilmiştir. HBC2 Filigranlama Tekniği tarafından doğrulanamayan piksel sayısı 16.500 olarak bulunmuştur.

Önerilen tekniğin doğrulama sonucuyla HBC2 Filigranlama Tekniği'nin doğrulama sonuçlarını karşılaştırabilmek için şekil 6.43.'de her iki tekniğin doğrulama sonuçları birlikte verilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Aynı kriptografik güvenlik seviyesinde önerilen teknik tarafından doğrulanamayan piksel sayısı 7.392 iken, HBC2 Filigranlama Tekniği tarafından doğrulanamayan piksel 16.500 olarak bulunmuştur.



(a) **(b)**

Şekil 6.43. Tekniklerin doğrulama sonuçlarının karşılaştırılması (çözünürlük=480*640 piksel)
(a) Önerilen tekniğin doğrulama sonucu **(b)** HBC2 Filigranlama Tekniğinin doğrulama sonucu

Dördüncü deneysel çalışmada önerilen teknik ile HBC2 Filigranlama Tekniği 2048 bitlik RSA sayısal imza algoritması kullanılarak karşılaştırılmıştır. Önerilen teknikte tüm imgeye ve alt imgelere ait sayısal imzaların hesaplanmasında RSA sayısal imza algoritması; bloklara ve bağımlılıklara ait sayısal imzaların hesaplanmasında 64 bitlik MAC kullanılmıştır. Karşılaştırma şekil 6.38.'de gösterilen 480*640 bitlik test imgesi üzerinde yapılmıştır.

Önerilen teknik ile imgenin filigranlanmasında ilk olarak imgenin EÖB düzlemi sıfır yapılmış ve alt blok yaklaşımı kullanılarak imge 240*320 piksellik iki alt imgeye, her bir alt imge 8*8 piksellik bloklara ve her bir blok 8*4 piksellik alt bloklara bölünmüştür. İkinci aşamada her bir alt imge içerisindeki bloklar, diğer alt imgedeki alt bloklar ile bağımlı hale getirilmiştir. Üçüncü aşamada 2048 bitlik RSA sayısal imza algoritması kullanılarak tüm imgeye ve alt imgelere ait sayısal imzalar; 64 bitlik MAC algoritması kullanılarak da bloklar ve bağımlılıklara ait sayısal imzalar hesaplanmıştır. Dördüncü aşamada şekil 6.3. ve 6.4.'de verilen algoritmalar kullanılarak $KBS_{S_1} = 1$, $KBS_{S_2} = 2$ ve $KBS_{S_3} = 2$; eşitlik 6.10., 6.11. ve 6.12. kullanılarak $KBS_{S_4} = 30$, $KBS_{S_5} = 15$ ve $KBS_{S_6} = 16$ olarak hesaplanmıştır. Son aşamada ise sayısal imzalara ait parçalar 4. aşamada belirlenen sayılara göre birleştirilerek veri yükü oluşturulmuş ve her bir bloğun EÖB düzlemine gömülerek filigranlı imge oluşturulmuştur. Filigranlı imge şekil 6.44.'de gösterilmiştir.

Önerilen tekniğin 2048 bitlik sayısal imza algoritması kullanıldıktan sonraki performansını değerlendirebilmek için şekil 6.44.'de verilen filigranlı imge üzerinde değişimler yapılmış ve şekil 6.45.'deki imge oluşturulmuştur.



Şekil 6.44. Önerilen teknik ile filigranlanmış imge (çözünürlük=480*640 piksel)



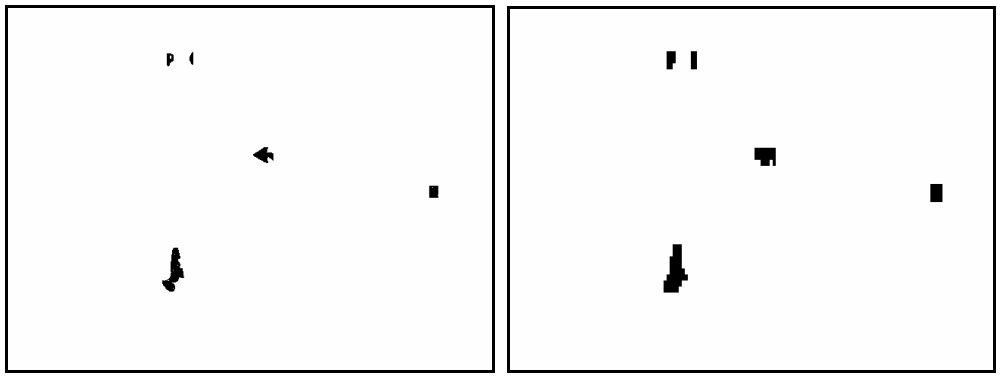
Şekil 6.45. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)

Değiştirilmiş filigranlı imgenin doğrulanması için önerilen tekniğin doğrulama algoritması çalıştırılmış ve alınan sonuç şekil 6.46.'da gösterilmiştir. Şekil 6.46.'da doğrulanamayan bölgeler siyah ile gösterilmiştir. Doğrulanamayan piksel sayısı ise 2.592 olarak bulunmuştur.

Önerilen tekniğin doğrulama sonucuyla, imge üzerinde yapılan gerçek değişimi karşılaştırabilmek için, şekil 6.47.'de gerçek değişim ve önerilen tekniğin doğrulama sonucu gösterilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Gerçekte değiştirilen piksel sayısı ise 1.418 olarak bulunmuştur.



Şekil 6.46. Önerilen tekniğin doğrulama sonucu (çözünürlük=480*640 piksel)



(a) **(b)**

Şekil 6.47. Gerçek değişim ile önerilen tekniğin doğrulama sonucunun karşılaştırılması (çözünürlük=480*640 piksel) **(a)** Gerçek değişim **(b)** Önerilen tekniğin doğrulama sonucu

HBC2 Filigranlama Tekniğinde de 2.048 bitlik RSA sayısal imza algoritması kullanılarak şekil 6.38.'de verilen test imgesi filigranlanmıştır. 2.048 bitlik veri yükünü taşıyacak blok büyüklüğü ise 32*64 piksel olarak seçilmiştir. HBC2 Filigranlama Tekniği ile filigranlanmış imge Şekil 6.48.'de gösterilmiştir.

2.048 bitlik RSA sayısal imzası kullanılması durumunda, önerilen teknik ile HBC2 Filigranlama Tekniğinin değişim bölgesi doğrulama niteliklerini karşılaştırabilmek için, şekil 6.44.'e uygulanan aynı değişimler şekil 6.48.'de verilen HBC2 Filigranlama Tekniği ile filigranlanmış imgeye de uygulanmıştır. Değiştirilmiş filigranlı imge şekil 6.49.'da verilmiştir.



Şekil 6.48. HBC2 Filigranlama Tekniği ile filigranlanmış imge (çözünürlük=480*640 piksel)



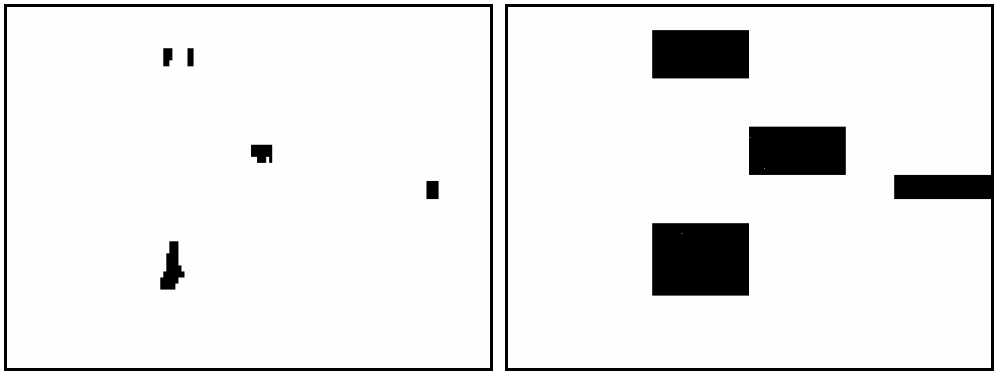
Şekil 6.49. Değiştirilmiş filigranlı imge (çözünürlük=480*640 piksel)

Değiştirilmiş filigranlı imgenin doğrulanması için HBC2 Filigranlama Tekniğinin doğrulama algoritması çalıştırılmış ve alınan sonuç şekil 6.50.'de gösterilmiştir. Şekil 6.50.'de doğrulanamayan bölgeler siyah ile gösterilmiştir. HBC2 Filigranlama Tekniği tarafından doğrulanamayan piksel sayısı 32.768 olarak bulunmuştur.

Önerilen tekniğin doğrulama sonucuyla HBC2 Filigranlama Tekniğinin doğrulama sonuçlarını karşılaştırabilmek için şekil 6.51.'de her iki tekniğin doğrulama sonuçları birlikte verilmiştir. Bu şekilde doğrulanan bölgeler beyaz, doğrulanamayan bölgeler siyah ile gösterilmiştir. Aynı kriptografik güvenlik seviyesinde önerilen teknik tarafından doğrulanamayan piksel sayısı 2.592 iken, HBC2 Filigranlama Tekniği tarafından doğrulanamayan piksel 32.768 olarak bulunmuştur.



Şekil 6.50. HBC2 Filigranlama Tekniğinin doğrulama sonucu (çözünürlük=480*640 piksel)



(a) **(b)**

Şekil 6.51. Tekniklerin doğrulama sonuçlarının karşılaştırılması (çözünürlük=480*640 piksel)
(a) Önerilen tekniğin doğrulama sonucu **(b)** HBC2 Filigranlama Tekniğinin doğrulama sonucu

Ayrıca önerilen teknik ile HBC2 Filigranlama Tekniğinin doğrulama çözünürlükleri 3 ayrı çözünürlükte ve 15 farklı test imgesi üzerinde karşılaştırılmıştır. Karşılaştırmada aynı kriptografik güvenlik seviyesi altında yapılmıştır. Elde edilen test sonuçları tablo 6.1.'de verilmiştir. Test imgeleri ve elde edilen imgesel sonuçlar ise Ek-2'de verilmiştir. Tablo 6.1.'den de görüldüğü gibi aynı kriptografik güvenlik seviyesinde önerilen teknik HBC2 Filigranlama Tekniğine göre daha yüksek doğrulukta değişim bölgesi belirleme niteliğine sahiptir. Tekniklerin detaylı karşılaştırılması bölüm 7.'de verilmiştir.

Tablo 6.1. HBC2 Filigranlama Tekniği ile önerilen tekniğin karşılaştırılması

Test imgesi	Çözünürlük (piksel)	HBC2 Filigranlama Tekniğinin Veri yükü (bit)	Önerilen Tekniğin Veri yükü (bit)	HBC2 Filigranlama Tekniğinin En Alt Seviyedeki Blok Büyüklüğü (piksel)	Önerilen Tekniğin En Alt Seviyedeki Blok Büyüklüğü (piksel)	Değiştirilen Piksel Sayısı	HBC2 Filigranlama Tekniği Tarafından Doğrulanamayan Piksel Sayısı	Önerilen Teknik Tarafından Doğrulanamayan Piksel Sayısı
İmge-1	1200*1600	288	64	12*25	8*8	53.819	82.200	58.976
İmge-2	1200*1600	288	64	12*25	8*8	5.890	23.100	9.152
İmge-3	1200*1600	288	64	12*25	8*8	87.075	129.900	92.832
İmge-4	1200*1600	288	64	12*25	8*8	195.401	293.700	207.584
İmge-5	1200*1600	288	64	12*25	8*8	27.759	71.400	35.232
İmge-6	768*1024	288	64	24*16	8*8	32.173	62.208	36.992
İmge-7	768*1024	288	64	24*16	8*8	8.559	43.776	13.792
İmge-8	768*1024	288	64	24*16	8*8	7.879	30.720	10.912
İmge-9	768*1024	288	64	24*16	8*8	17.683	40.704	21.440
İmge-10	768*1024	288	64	24*16	8*8	8.735	25.728	11.232
İmge-11	480*640	288	64	30*10	8*8	1.418	8.400	2.592
İmge-12	480*640	288	64	30*10	8*8	3.689	21.300	7.296
İmge-13	480*640	288	64	30*10	8*8	5.575	25.200	8.800
İmge-14	480*640	288	64	30*10	8*8	6.885	29.400	10.464
İmge-15	480*640	288	64	30*10	8*8	4.878	16.500	7.392

6.3.4. Deneysel Sonuçların Değerlendirilmesi

Bloksal Bağımlı Kırılgan Filigranlama Tekniklerinde bir bloktaki bozulmanın, hem bozulma olan bloğun doğrulanamamasına hem de bozulan bloğa bağımlı diğer blokların doğrulanamamasına neden olması, bu tekniklerin değişim bölgesi belirlemedeki doğruluklarını önemli miktarda azaltmaktadır. Bu durum bloksal bağımlı bir kırılgan filigranlama tekniği olan HBC2 Filigranlama Tekniği ile yapılan deneysel çalışmaların sonucunda görülmüştür. Örneğin önerilen teknik ile HBC2 Filigranlama tekniğinin 1200*1600 piksellik bir imgenin doğrulanmasındaki test sonuçlarının birlikte verildiği şekil 6.35.'i inceleyelim. Bu şekilde önerilen teknik tarafından değiştirilen bölgelerin daha yüksek bir doğrulukta belirlendiği görülmektedir. Önerilen teknik tarafından doğrulanamayan piksel sayısı 21.120 iken, HBC2 Filigranlama tekniği tarafından doğrulanamayan piksel sayısı 38.100 olarak bulunmuştur. Gerçekte değiştirilen piksel sayısı ise 18.000'dir. Önerilen Tekniğin, HBC2 Filigranlama Tekniğine göre daha yüksek doğrulukta değişim bölgesi belirleme niteliğine sahip olduğu 768*1024 ve 480*640 piksellik test imgeleri üzerinde yapılan ve şekil 6.39. ve 6.43.'de sonuçları verilen deneysel çalışmalarda da görülmektedir. Ayrıca tablo 6.1.'de verilen ve 15 farklı test imgesi üzerinde yapılan deneysel çalışmaların sonucu da önerilen tekniğin HBC2 Filigranlama Tekniğine göre daha yüksek doğrulukta değişim bölgesi belirleme niteliğine sahip olduğunu göstermektedir.

Önerilen tekniğin VNTS ve Kes Yapıştır Saldırılarına karşı güvenilir bir yapıya sahip olduğu şekil 6.29. ve 6.31.'de verilen doğrulama sonuçlarında görülmektedir. Önerilen tekniğe karşı Aktarıma Dayalı Saldırı'nın yapılabilmesi ise tüm imgeye ve alt imgelere ait sayısal imzaların kullanımıyla engellenmiştir. Doğum Günü Saldırı'sının yapılabilmesi ise tüm imge ve alt imgelere ait sayısal imzaların kullanımıyla engellenmiştir.

Önerilen tekniğin HBC2 Filigranlama Tekniğine göre diğer önemli bir avantajı daha yüksek kriptografik güvenirlik seviyesine ve daha uzun bit çıkışına sahip sayısal imzalar kullanıldığında ortaya çıkmaktadır. 2048 bitlik RSA sayısal imza algoritması kullanılarak 480*640 piksellik test imgesi üzerinde yapılan deneysel çalışmanın sonucu şekil 6.51.'de verilmiştir. Önerilen teknik tarafından doğrulanamayan piksel sayısı 2.592 iken, HBC2 Filigranlama Tekniği tarafından doğrulanamayan piksel sayısı 32.768 olarak bulunmuştur. Bu sonuç, yüksek bit çıkışına sahip daha güvenilir bir sayısal imza algoritması kullanılması durumunda önerilen tekniğin HBC2 Filigranlama Tekniğine göre oldukça yüksek bir performansa sahip olduğunu göstermektedir.

7. SONUÇLAR

Bu tez çalışmasında imgelerin aslıyla aynılığını doğrulamak için hem saldırılara karşı güvenilir hem de yüksek doğrulukta değişim bölgesi belirleme niteliğine sahip iki Kırılgan Filigranlama Tekniği sunulmuştur. Önerilen tekniklerde filigran olarak sayısal imzaların kullanılmasıyla, sayısal imzaların sahip olduğu yüksek seviyeli kriptografik güvenlik, bütünlük doğrulama, kaynak doğrulama ve inkar edememe nitelikleri filigranın bünyesine katılmıştır.

Önerilen Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniği, Sıradüzensel Filigranlama Tekniği temel alınarak geliştirilmiştir. Önerilen bu teknik, Sıradüzensel Filigranlama Tekniğine göre üç önemli avantaja sahiptir:

1) Sayısal bir imge sıradüzenin en üst seviyesinde bir alt seviyeye bölünemediğinde, Sıradüzensel Filigranlama Tekniğinin değişim bölgesi belirleme niteliği yok olmaktadır. Dolayısıyla Sıradüzensel Filigranlama Tekniği'nin imge çözünürlüğüne bağımlı olarak değişim bölgesi belirleyebilme niteliğine sahip olduğu söylenebilir. Bu problem önerilen teknik içerisinde sıradüzensel ağaç yapısının kullanılmasıyla çözülmüştür. Sıradüzensel ağaç yapısı, tavan ve taban fonksiyonlarının sıradüzen içerisinde kullanılmasıyla oluşturulmuştur. Bu çözüm önerilen tekniğin imge çözünürlüğünden bağımsız olarak değişim bölgesi belirleyebilme niteliğine sahip olmasını sağlamıştır.

2) Sayısal bir imgenin sıradüzensel yapının üst seviyelerinde bir alt seviyeye bölünememesi durumunda, Sıradüzensel Filigranlama Tekniği'nin değişim bölgesi belirleme doğruluğu oldukça azalmaktadır. Önerilen teknikte sıradüzensel ağaç yapısıyla bu problem de çözülmüştür. Tavan ve taban fonksiyonları sıradüzenin her bir seviyesindeki blokların bir alt seviyeye bölünmesini garanti ederek veri yükünü taşıyacak minimum boyutlu blokların oluşmasını sağlamıştır. İmge çözünürlüğünden bağımsız olarak imgenin istenilen alt seviyelere kadar bölünebilmesi, önerilen tekniğin değişim bölgesini yüksek doğrulukta belirleyebilmesini temin etmiştir.

3) Sıradüzensel Filigranlama Tekniğinde daha yüksek kriptografik güvenlik seviyesi elde edebilmek için daha uzun bit çıkışına sahip sayısal imza algoritmalarının kullanılması durumunda, en alt seviyedeki blokların taşıyacağı veri yükü önemli miktarda artmaktadır. Veri yükünün artması ise en alt seviyedeki blokların büyüklüğünü artırarak değişim yapılan bölgenin belirlenmesindeki doğruluğu azaltmaktadır. Gerek Sıradüzensel Filigranlama Tekniğinde gerekse önerilen teknikte imgenin saldırılara karşı güvenliği en üst seviyedeki bloklara ait sayısal imzalarla sağlanmaktadır. En alt seviyedeki bloklara ait sayısal imzalar değişim bölgesini belirlemek amacıyla kullanılmaktadır. Bu yüzden önerilen teknikte iki ayrı sayısal

imza algoritması kullanılmıştır. Daha güvenilir ve daha uzun bit çıkışına sahip sayısal imza algoritması üst seviyelerdeki bloklara ait sayısal imzaların hesaplanmasında kullanılırken, daha az bit uzunluğuna ve güvenilirliğe sahip sayısal imza algoritması alt seviyedeki bloklara ait sayısal imzaların hesaplanmasında kullanılmıştır. Bu kullanım önerilen tekniğin veri yükünü azaltmış ve en alt seviyedeki blokların daha küçük boyutlu seçilmesine imkan vererek önerilen tekniğin değişim bölgesi belirleme doğruluğunda önemli bir performans artışı sağlamıştır. Önerilen tekniğin asıl güvenliği üst seviyelerde kullanılan daha güvenilir Sayısal imza algoritmasıyla sağlandığı için, alt seviyelerdeki bloklarda daha az kriptografik güvenlik seviyesine sahip bir sayısal imza algoritmasının kullanılması önerilen tekniğin genel güvenlik seviyesinde bir azalmaya neden olmamıştır. Ayrıca bu kullanım 2048 bitlik RSA sayısal imza algoritması gibi çok uzun bit çıkışına sahip imzaların kullanılması durumunda, oluşan veri yükü ile güvenlik seviyesi arasında bir denge sağlayarak önerilen tekniğin genel performansını artırmıştır.

Bu tez çalışmasında sunulan ikinci kırılgan filigranlama tekniği bloksal bağımlılığı kullanmaktadır. Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniği olarak adlandırılan bu teknik, bloksal bağımlılığı kullanan tekniklere göre 3 önemli avantaja sahiptir.

1) Bloksal bağımlılığı kullanan filigranlama tekniklerinde saldırılara karşı güvenlik bloklar arasında kurulan bağımlılık ile sağlanır. Önerilen teknikte ise güvenlik hem her bir alt imgedeki blokların diğer alt imgedeki iki farklı alt blok ile oluşturdukları bağımlıklar ile hem de tüm imgeye ve alt imgelere ait sayısal imzaların kullanımıyla sağlanmaktadır. Bu yapı saldırılara karşı yüksek seviyede bir güvenlik sağlamaktadır.

2) Bloksal bağımlı kırılgan filigranlama tekniklerinde bir bloktaki bozulma, hem bozulma olan bloğun doğrulanamamasına hem de bozulan bloğa bağımlı blokların doğrulanamamasına neden olur. Önerilen teknikte ise yeni bir doğrulama algoritması geliştirilerek bloktaki bir bozulmanın sadece o bloğun yarısının doğrulanamamasına neden olması sağlanmıştır. Bu çözüm değişim bölgesinin 4 kata kadar daha doğru bir çözünürlükte belirlenebilmesini sağlamıştır.

3) Bloksal bağımlı kırılgan filigranlama tekniklerinde daha yüksek kriptografik güvenlik seviyesi elde edebilmek için daha uzun bit çıkışına sahip sayısal imza algoritmalarının kullanılması durumunda, her bir bloğun taşıyacağı veri yükü önemli miktarda artmaktadır. Bu da blok büyüklüğünü artırarak değişim bölgesinin belirlenmesindeki doğruluğu azaltmaktadır. Önerilen teknikte, Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniğinde olduğu gibi iki ayrı sayısal imza algoritmasının kullanılması önerilmiştir. Daha güvenilir ve daha uzun bit çıkışına sahip sayısal imza algoritması tüm imgeye ve alt imgelere ait sayısal imzaların hesaplanmasında kullanılırken, daha az bit uzunluğuna ve güvenilirliğe sahip sayısal imza

algoritması blokların ve blokların diğer alt imgedeki bloklarla oluşturdukları bağımlılıklara ait sayısal imzaların hesaplanmasında kullanılması önerilmiştir. Ayrıca tüm imgeye ve alt imgelere ait sayısal imzalar imgedeki bloklara dağıtıldığı için her bir blok üzerindeki veri yükü miktarı azaltılarak önerilen tekniğin değişim bölgesi belirleme performansı artırılmıştır. Ayrıca önerilen teknikte blok ve bloksal bağımlılıklara ait sayısal imzaların bir parçası kullanılmıştır. Bu kullanım önerilen tekniğin değişim bölgesi doğrulama performansını büyük miktarda artırırken, blok ve bağımlı bloklara ait güvenlik seviyesinde bir azalmaya neden olabilir. Ancak önerilen tekniğin asıl güvenliği tüm imgeye ve alt imgelere ait sayısal imzalar ile sağlandığı için sistemin genel güvenliği tehlikeye atılmamıştır. Ayrıca sayısal imzaların bir parçasının kullanılması, blok ve bağımlı bloklara ait sayısal imzalarda simetrik anahtar yapısının kullanılmasını zorunlu kılmaktadır.

Bu tez çalışmasında önerilen tekniklerde, blokların aslıyla aynılığını doğrulamak amacıyla sayısal imza algoritmaları kullanılmıştır. Ancak sayısal imzaların kullanılmasında bir bitlik bir bozulma tüm bloğun doğrulanamamasına neden olmaktadır. İleriki çalışmalarda bu probleme çözüm olarak sayısal imza algoritmalarıyla, hata sezme ve düzeltme kodlarının bir arada kullanılabileceği bir yapı geliştirilerek daha verimli bir kırılgan filigranlama tekniği geliştirilebilir.

KAYNAKLAR

1. Schneier, B., 1996, Applied Cryptography Second Edition: Protocols, Algorithms, and Source Code in C, John Wiley & Sons, Inc, New York, 758 p.

2. X10. Menezes, A., Oorchot, P., Vanstone, S., 1997, Handbook of Applied Cryptography, CRC Press, Boca Raton, 816 p.

3. Mel, H. X., Baker, D., 2000, Cryptography Decrypted, Addison-Wesley, Boston, 352 p.

4. Stinson, D. R., 2002, Cryptography Theory and Practice, Chapman & Hall/CRC Press Company, Boca Raton, 339 p.

5. Cox, I. J., Miller, M. L., Bloom, J. A., 2002, Digital Watermarking, Morgan Kaufmann Publishers, San Francisco, 542 p.

6. Lin, E. T., Podilchuk, C. I. and Delp, E. J., 2000, Detection of Image Alterations Using Semifragile Watermarks, Proceeding of the SPIE Security and Watermarking of Multimedia Contents II, 3971, 152-163.

7. Kundur, D., Hatzinakos, D., 1999, Digital Watermarking for Telltale Tamper Proofing and Authentication, Proceedings of the IEEE, 87, No. 7, 1167-1180.

8. Fei, C., Kundur, D., and Kwong, R., 2004, Analysis and Design of Authentication Watermarking, Proceedings of the SPIE Security, Steganography and Watermarking of Multimedia Contents VI, 5306, 760-771.

9. Queluz, M. P., 2002, Spatial Watermark for Image Content Authentication, Journal of Electronic Imaging, 11, No.2, 275-285.

10. Ekici, Ö., Sankur, B., Coşkun, B., Naci, U., and Akcay, M., Comparative Evaluation of Semifragile Watermarking Algorithms, Journal of Electronic Imaging, 13, No. 1, 209-216.

11. Lin, C. Y., Chang, S. F., 2001, A Robust Image Authentication Method Distinguishing, JPEG Compression from Malicious Manipulation, IEEE Transactions on Circuit and Systems for Video Technology, 11, No. 2, 153-168.

12. Maeno, K., Sun, Q., Chang, S., Suto, M., 2006, New Semi-fragile Image Authentication Watermarking Techniques Using Random Bias and Nonuniform Quantization, IEEE Transactions on Multimedia, 8, No. 1, 32-45.

13. Tong, L., Xiangyi, H., Yiqi, D., 2004, Semi-fragile Watermarking for Image Content Authentication, International Conference on Signal Processing, ICSP'04, 3, 2342-2345

14. Cox, I. J., Miller, M. L., Bloom, J. A., 2000, Watermarking Applications and Their Properties, Proceedings of the International Conference on Information Technology: Coding and Computing - ITCC2000, 6-10.

15. Podilchuk, C. I., Delp, E. J., 2001, Digital Watermarking: Algorithms and Applications, IEEE Signal Processing Magazine, 18, No. 4, 33-46.

16. Vidyasagar, M. P., Han, S., Chang, E., 2005, A Survey of Digital Image Watermarking Techniques, Proceedings of the 3rd International IEEE Conference on Industrial Informatics (INDIN), 709-716.

17. Langelaar, G., Setyawan, I., Lagendijk, R. L., 2000, Watermarking Digital Image and Video Data: A State of The Art Overview, IEEE Signal Processing Magazine, vol. 17, no. 5, 20-46.

18. Petitcolas, F. A. P., 2000, Watermarking Schemes Evaluation, IEEE Signal Processing Magazine, 17, No. 5, 58-64.

19. Miller, M. L., Doerr, G., Cox, I. J., 2004, Applying Informed Coding and Embedding to Design A Robust High Capacity Watermark“, IEEE Transaction on Image Processing, 13, No. 6, 792-807.

20. Cox, I. J., Miller, M. L., McKellips, A. L., 1999, Watermarking As Communication With Side Information, Proocedings of the IEEE, Special Issue on Identification and Protection of Multimedia Information, 87. No. 7, 1127-1141.

21. Chen, B., and. Wornell, G.W., 1998, Digital Watermarking and Information Embedding using Dither Modulation, Proc. of IEEE Workshop on Multimedia Signal Processing, 273-278.

22. Chen, B., and. Wornell, G.W., 1999, Dither modulation: A New Approach to Digital Watermarking and Information Embedding, Proc. of SPIE Security and Watermarking of Multimedia Contents, 3657, 342 – 353.

23. Chen, B., and. Wornell, G.W., 1999, An Information-theoretic Approach to the Design of Robust Digital Watermarking Systems, Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '99), 4, 2061 – 2064.

24. Chen, B., and. Wornell, G.W., 1999, Achievable Performance of Digital Watermarking Systems, Proc. of IEEE International Conference on Multimedia Computing and Systems (ICMCS-99), Special Session on Security and Watermarking, 1, 13 - 18,

25. Chen, B., and. Wornell, G.W., 2000, Preprocessed and Postprocessed Quantization Index Modulation Methods for Digital Watermarking," Proc. of SPIE: Security and Watermarking of Multimedia Contents II, 3971, 48 – 59.

26. Chen, B., and. Wornell, G.W., 2001, Quantization Index Modulation: A Class of Provably Good Methods for Digital Watermarking and Information Embedding, IEEE Transactions on Information Theory, 47, no. 4, 1423-1443.

27. Eggers, J.J., Su, J.K., and Girod, B., 2000, A Blind Watermarking Scheme Based on Structured Codebooks, Secure Images and Image Authentication, Proc. IEE Colloquium, 4, 1-6.

28. Eggers, J.J.; Bauml, R.; Tzschoppe, R.; Girod, B., 2003, Scalar Costa Scheme for Information Embedding,IEEE Transactions on Signal Processing, 51, No. 4, 1003-1019.

29. Pickholtz, R. L., Schilling, D. L., Milstein, L. B., 1982, Theory of Spread Spectrum Communications - A Tutorial, IEEE Transactions on Communications, 30, No. 5, 855-884.

30. Malvar H. S., Florencio, D. A. F., 2003, Improved Spread Spectrum: A New Modulation Technique for Robust Watermarking. IEEE Trans. on Signal Processing, 51, No. 4, 898-905.

31. Dan, Y., Sattar, F., Razul, S.G., 2004, Transparent Robust Information Hiding for Ownership Verification, IEEE International Conference on Acoustics, Speech, and Signal Processing, (ICASSP '04), 3, 401-404

32. Seshadrinathan, K., Sheikh, H.R., Bovik, A.C., 2005, Detecting Spread Spectrum Watermarks Using Natural Scene Statistics, IEEE International Conference on Image Processing, ICIP 2005, 2, 1106-1109.

33. Barni, M., Bartolini, F., De Rosa, A., 2002, On the Performance of Multiplicative Spread Spectrum Watermarking, IEEE Workshop on Multimedia Signal Processing, 324-327.

34. Lin, E. T., and Delp, E. J., 1999, A Review of Fragile Image Watermarks, Proc. Multimedia and Security Workshop at ACM Multimedia' 99, 25-29.

35. Fridrich, J., 1999, Methods for Tamper Detection in Digital Images, Proc. Multimedia and Security Workshop at ACM Multimedia' 99, 19-23.

36. Albanesi, M. G., Ferretti, M., Guerrini, F., 2001, A Taxonomy for Image Authentication Techniques and Its Application to the Current State of the Art, Proc. 11th Int. Conf. Image Analysis and Processing, 535 -540.

37. Wong, P. W., 1998, A Watermark for Image Integrity and Ownership Verification, Proc. IS&T PIC, 374-379.

38. Wong, P. W., 1998, A Public Key Watermark for Image Verification and Authentication, Proc. IEEE Int. Conf. Image Processing, 1, 455-459.

39. Holliman M., and Memon, N., 2000, Counterfeiting Attacks on Oblivious Blockwise Independent Invisible Watermarking Schemes, IEEE Trans. Image Processing, 9, No. 3, 432-441.

40. Wong, P. W., Memon, N., 2000, Secret and Public Key Authentication Watermarking Schemes that Resist Vector Quantization Attack, Proc. SPIE, 3971, No. 40, 417-427.

41. Fridrich, J., Goljan, M., Baldoza, A. C., 2000, New Fragile Authentication Waternark for Images, IEEE International Conference on Image Processing, 1, 446-449.

42. Barreto, P. S. L. M., Kim, H. Y., Rijmen, V., 2002, Toward Secure Public Key Blockwise Fragile Authentication Watermarking, IEE Proceedings Vision, Image and Signal Processing, 149, No. 2, 57-62.

43. Ertürkler, M., Tatar, Y., 2006, A Novel Fragile Watermarking Technique with High Accuracy Tamper Localization, IEEE 14.th Signal Processing and Communications Applications, (SIU 2006), 1-4.

44. Walton, S., 1995, Information Authentication for a Slippery New Age, Dr. Dobbs Journal, 20, No. 4, 18-26.

45. Yeung M. M. and Mintzer, F., 1997, An Invisible Watermarking Technique for Image Verification, IEEE Int. Conf. Image Processing, 2, 680-683.

46. Wu, M., and Liu, B., 1998, Watermarking for Image Authentication, IEEE Int. Conf. Image Processing, 2, 437-441.

47. Memon, N., Shende, S., and Wong, P. W., 1999, On the Security of the Yeung- Mintzer Authentication Watermark, Proc. IS&T PICS Symp. Savannah, Georgia, 301-306.

48. Fridrich, J., Goljan, M., and Memon, N., 2000, Further Attacks on Yeung-Mintzer Fragile Watermarking Scheme, Proc. SPIE Photonic West, Electronic Imaging 2000, Security and Watermarking of Multimedia Contents, 428–437.

49. Coppersmith, D., Mintzer, F., Tresser, C., Wu, C. W. and Yeung, M. M. 1999, Fragile Imperceptible Digital Watermark with Privacy Control," Proc. SPIE/IS&T Int. Symp. Electronic Imaging: Science and Technology, 3657, 79-84.

50. Fridrich, J., 2002, Security of Fragile Authentication Watermarks with Localization, Proc. SPIE Photonic West, Vol. 4675, Electronic Imaging 2002, Security and Watermarking of Multimedia Contents, San Jose, California, January, 2002, pp. 691-700.

51. Wong, P. W. and Memon, N., 2001, Secret and Public Key Image Watermarking Schemes for Image Authentication and Ownership Verification, IEEE. Trans. Image Processing, 10, No. 10, 1593-1601.

52. Çelik, M. U., Sharma, G., Saber, E. and Tekalp, A. M. 2002, Hierarchical Watermarking for Secure Image Authentication with Localization, IEEE Transactions on Image Processing, 11, No. 6, 585-595.

53. Yuan, H., Zhang, X.P., 2006, Multiscale Fragile Watermarking Based on the Gaussian Mixture Model, IEEE Transactions on Image Processing, 15, No.10, 3189-3200.

54. Li, C.-T., 2004, Digital Fragile Watermarking Scheme for Authentication of JPEG Images, IEE Proceedings Vision, Image and Signal Processing, 151, No. 6, 460-466.

55. Honsinger, C. W., Jones, P. W., Rabbani, M. and Stoffel, J. C., 2001, Lossless Recovery of an Original Image Containing Embedded Data," U. S. Patent No. 6,278,791.

56. Fridrich, J., Goljan, M. and Du, R., 2001, Invertible Authentication, Proc. SPIE 3971, Security and Watermarking of Multimedia Contents III, 197-208.

57. Goljan, M., Fridrich, J., and Du, R., 2001, Distortion-free Data Embedding, 4th Information Hiding Workshop, Pittsburgh, Pennsylvania, April, 2001.

58. Leest, A.V., Veen, M. V., Bruekers, F., 2003, Reversible Image Watermarking, Proc. of the IEEE International Conference on Image Processing, II, 731-734.

59. Fridrich, J., Goljan, M., and Du, R., 2002, Lossless Data Embedding – New Paradigm in Digital Watermarking, EURASIP Journal of Applied Signal Processing, Special Issue on Emerging Applications of Multimedia Data Hiding, 2002, No.2, 185-196.

60. Holliman, M., Memon, N. and Yeung, M. M., 1999, On the Need for Image Dependent Keys for Watermarking, Proc. Content Security and Data Hiding in Digital Media, Newark, NJ, May 14.

61. J. Fridrich, 1999, Robust Bit Extraction from Images," Proc. IEEE Int. Conf. Multimedia Computing & Systems, 2, 536-540.

62. Hunter, D., 1967, Handmade Paper and Its Watermark: A Bibliography, New York: B. franklin

63. Parlatır, İ., Gözaydın, N., Zülfikar, H., 1998, Türkçe Sözlük, Türk Dil Kurumu, 1136 s.

64. Eggers, J., Girod, B., 2002, Informed Watermarking, Kluwer Academic Publishers, Boston, 182 p.

65. Swanson, M., Kobayashi, Tewfik, A., 1998, Multimedia Data Embedding and Watermarking Technologies, Proceedings of the IEEE, 86, No. 6, 1064-1087.

66. Hartung, F., Kutter, M., 1999, Multimedia Watermarking Techniques, Proceedings of the IEEE, 87, No. 7, 1079-1107.

67. Rey, C., Dugelay, J., 2002, A Survey of Watermarking Algorithms for Image Authentication, EURASIP Journal of Applied Signal Processing, 6, 613-621.

68. Freire, L. P., Comesana, P., Pastoriza, J. R. and Gonzalez, F. P., 2006, Watermarking Security: A Survey, LNCS Transactions on Data Hiding and Multimedia Security, To appear.

69. Moulin, P., Koetter, R., 2005, Data Hiding Codes, Proceedings of the IEEE, 93, No. 12, 2083 – 2126.

70. Lin, E. T., 2005, Video and Image Watermark Synchronization, Ph.D. Thesis, Purdue University, 217s.

71. Petitcolas, F. A. P., Anderson,R. J., Kuhn, M. G., 1999, Information hiding: A Survey, Proceedings of the IEEE, Special Issue on Identification and Protection of Multimedia Information, 87, No. 7, 1062-1078.

72. Ramkumar, M., 2000, Data Hiding in Multimedia: Theory and Applications. PhD thesis, New Jersey Institute of Technology. 153p.

73. Cox, I. J., Miller, M. L., 2002, The First 50 Years Of Electronic Watermarking, EURASIP Journal of Applied Signal Processing, No. 2, 126-132.

74. Cox, I. J., Doerr, G., Furon, T. 2006, Watermarking Is Not Cryptography, Proceedings of the 5th International Workshop on Digital Watermarking, 1-15.

75. Johnson, N. F., Duric, Z., Jajodia, S., 2001, Information Hiding: Steganography and Watermarking-Attacks and Countermeasures, Kluwer Academic Publishers, Boston, 137 p.

76. Katzenbeisser S., and Petitcolas, F. A. P., 2000, Information Hiding: Techniques for Steganography and Digital Watermarking. Norwood, MA: Artech House. 168 p.

77. Fridrich J., and Goljan, M., 2002, Practical Steganalysis of Digital Images - State of the Art", Proceedings of SPIE, Security and Watermarking of Multimedia Contents IV, 4675, 1-13.

78. Dabeer O., Sullivan, K., Madhow, U., Chandrasekaran, S., and Manjunath, B. S., 2004, Detection of Hiding in the Least Significant Bit, IEEE Transactions On Signal Processing, 52, No. 10, 3046-3058.

79. Fridrich J., Goljan M., Soukal, D., 2006, Wet Paper Codes with Improved Embedding Efficiency, IEEE Transactions on Information Security and Forensics, 1, No. 1, 102-110.

80. Kipper, G., 2004, Investigator's Guide to Steganography, Auerbach Publications, A CRC Press Company, Boca Raton, 220 p.

81. Kahn, D., 1996, The Codebreakers, 2nd edition, New York: Macmillan. 974 p.

82. Kilburn, D., 1997, Dirty Linen Dark Secrets, Adweek, 38, No. 40, 35-40.

83. Kalker, T., Depovere, G., Haitsma, J., Maes, M., 1999, A Video Watermarking System for Broadcast Monitoring, Proceedings of the SPIE Security and Watermarking of Multimedia Contents, 3657, 103-112.

84. Bloom, J. A., Cox, I. J., Kalker, T., Linnartz, J. P., Miller, M. L., Traw, B., 1999, Copy Protection for DVD Video, Proceedings of The IEEE, 87, No. 7, 1267-1276.

85. Friedman, G. L., 1993, The Thrustworthy Camera: Restoring Credibility to the Photographic Image, IEEE Transactions on Consumer Electronics, 39, No. 4, 905-910.

86. Friedman, G. L., 1996, Digital Camera with Apparatus for Authentication of Images Produced from an Image File, United States Patent, 5,499,294.

87. Çelik, M. U., Sharma, G., Tekalp, A. M., 2004, Collusion-Resilient Fingerprinting by Random Pre-Warping, IEEE Signal Processing Letters, 11, No. 10, 831-835.

88. Ertürkler, M., Tatar, Y., 2002, Tıbbi Görüntülerin Arşivlenmesinde Sayısal Filigranlama Kullanılarak Güvenilirliğin Artırılması, Biyomedikal Mühendisliği Ulusal Toplantısı, Biyomut 2002, 107-110.

89. Voyatzis, G., Nikolaides, N., Pitas, I., 1998, Digital Watermarking: An Overview, Proceedings of IX European Signal Processing Conference (EUSIPCO), 13-16,

90. Jayant, N., Johnston, J., Safranek, R., 1993, Signal Compression Based on Models of Human Perception, Proceedings of the IEEE, 81, No. 10, 1385-1442.

91. Mayache, A. M., Eude, T., and Cherifi, H., 1998, A Comparison of İmage Quality Models and Metrics Based on Human Visual Sensitivity, IEEE International Conference on Image Processing, 3, 409 – 413.

92. Watson, A. B., 1993, DCT Quantization Matrices Visually Optimized for Individual Images, Human Vision, Visual Processing, and Digital Display IV, SPIE-1913, 202-216.

93. Tao, B., Dickinson, B., 1997, Adaptive Watermarking In The DCT Domain, Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, 4, 2985-2988.

94. Podilchuk, C., Zeng, W., 1998, Image-Adaptive Watermarking Using Visual Models, IEEE Journal On Selected Areas In Communications, 16, No. 4, 525-539.

95. Porter, J. P., Rajan, P. K., 2006, Image Adaptive Watermarking Techniques Using Models of the Human Visual System, Proceeding of the Thirty-Eighth Southeastern Symposium on System Theory, 354-357.

96. Barni, M., Bartolini, F., Furon, T., 2003, A General Framework for Robust Watermark Security, Signal Processing, 83, No. 10, 2069-2084.

97. Cayre, F., Fontaine, C., Furon, T., 2005, Watermarking Security: Theory and Practice, IEEE Transactions on Signal Processing, 53, No. 10, 3976-3987.

98. Pérez-Freire, L., Comesaña, P., Troncoso-Pastoriza, J. R., Pérez-González, F., 2006, Watermarking Security: A Survey, LNCS Transactions on Data Hiding and Multimedia Security*, To appear.*

99. Chang, E. C., Orchard, M., 2000, Geometric Properties of Watermarking Schemes, IEEE International Conference on Image Processing, 3, 714-717.

100. Moulin, P., O'Sullivan, J. A., 1999, Information Theoretic Analysis of Information Hiding, IEEE Transactions on Information Theory, 49, No. 3, 563-593.

101. Kerckhoffs, A., 1883, La Cryptographie Militaire, Journal des sciences militaires, 9:5–38.

102. Furon, T., 2002, Use of Watermarking Techniques for Copy Protection, PhD. thesis, Ecole Nationale Supérieure des Télécommunications, 178s.

103. Vleeschouwer, C., Delaigle, J. F., Macq, B., 2002, Invisibility and Application Functionalities In Perceptual Watermarking: An Overview, Proceedings of The IEEE, 90, No. 1, 64-77.

104. Kutter, M., Winkler, S., 2002, A Vision Based Masking Model for Spread Spectrum Image Watermarking, IEEE Transaction on Image Processing, 11, No. 1, 16-25.

105. Barni, M., Bartolini, F., Piva, A., 2001, Improved Wavelet Based Watermarking Through Pixel Wise Masking, IEEE Transaction On Image Processing, 10, No. 5, 783-791.

106. Podilchuk, C., Zeng, W., 1997, Perceptual Watermarking of Still Images, in Proc. IEEE Multimedia Signal Processing, 363–368.

107. Erturkler, M., Tatar, Y., Arslan, A., 2002, A Novel Method Based on Fuzzy Logic Technique for Digital Watermarking In The Frequency Domain, 2. FAE International Symposium,357-360.

108. Gonzalez, R. C., Woods, R. E., 2002, Digital Image Processing, Prentice Hall, New Jersey, 793p,

109. Pitas, I., 1995, Applying Signatures On Digital Images, IEEE International Conference On Image Processing, 3, 215-218.

110. Wolfgang, R. B., Delp, E. J., 1996, A Watermark For Digital Images, IEEE International Conference On Image Processing, 3, 219-222.

111. Bender, W.,Gruhl, D., Morimoto, N., 1995, Techniques For Data Hiding, In Proc. SPIE, Storage and Retrieval for Image and Video Databases III, 2420, 165-173.

112. Langelaar, G. C., Van der Lubbe, J. C. A., Lagendjick, R. L., 1997, Robust Labeling Methods for Copy Protection of Images, In Proc. SPIE Electronic Imaging'97, Storage and Retrieval for Image and Video Databases, 298-309.

113. Zeng, W., Liu, B., 1997, On Resolving Rightful Ownerships of Digital Images by Invisible Watermarks, In Proc. ICIP'97, IEEE International Conference on Image Processing, 552-555.

114. Cox, I., Kilian, J., Leighton, T., Shamoon, T., 1997, "Secure Spread Spectrum Watermarking For Multimedia," IEEE Trans. Image Processing, 6, 1673–1687.

115. Van Schyndel, R. G., Tirkel, A. Z., Osborne, C. F., 1994, A Digital Watermark, In Proc. IEEE Int. Conference on Image Processing, 2, 86-89.

116. Hernandez, J. R., Gonzalez, F., Rodriguez, J. M., Nieto, G., 1998, Performance Analysis of A 2D Multipulse Amplitude Modulation Scheme for Data Hiding and Watermarking of Still Images, IEEE J. Select Areas Communication, 16, 510-524.

117. James, F., 1990, A Review of Pseudo Random Number Generators, Computer Physics Communications, 60, No. 3, 329-344.

118. Ripley, B., 1983, Computer Generation of Random Variables: A Tutorial, International Statistical Review, 51, 301-319.

119. Cannons, J., Moulin, P., 2004, Design and Statistical Analysis of a Hash-Aided Image Watermarking System, IEEE Transactıons On Image Processıng, 13, No. 10, 1393-1408.

120. Solachidis, V.; Pitas, I., 2003, Optimal Detection For Multiplicative Watermarks Embedded In DFT Domain, International Conference on Image Processing, 3, 723-726.

121. Staring, M., 2002, "Analysis of Quantization Based Watermarking," Master's thesis, University of Twente, 113s.

122. Piva, A., Barni, M., Bartolini, F., Cappelini, V., 1998, Threshold Selection for Correlation-Based Watermark Detection, Proceedings of COST 254 Workshop on Intelligent Communications, L'Aquila, Italy, 67-72.

123. Solachidis, V., Pitas, I., 2001, Circularly Symmetric Watermark Embedding in the 2-D DFT Domain, IEEE Transactions on Image Processing, 10, No. 11, 1741-1753.

124. Barni, M., Bartolini, F., Rosa, A. D., and Piva, A., 2001, A New Decoder for the Optimum Recovery of Nonadditive Watermarks, IEEE Transactions on Image Processing, 10, No. 5, 755-766.

125. Solachidis, V., Pitas, L., 2001, Circularly symmetric watermark embedding in 2-D DFT domain, IEEE Transactions on Image Processing 10, No. 11, 1741-1753

126. Pennebaker, W. B., Mitchell, J. L., 1993, JPEG Stil Image Data Compression Standard, Van Nostrand Reinhold, New York, 638p.

127. Barni, M., Bartolini, F., Cappelini, V., Piva, A., 1998, A DCT Domain System for Robust Image Watermarking, Signal Processing, 66, No. 3, 357-372.

128. Wolfgang, R., Podilchuk, C., and Delp, E., 1999, Perceptual Watermarks for Digital Images and Video, Proceedings of the IEEE , 87, No. 7, 1108-1126.

129. Barni, M., Bartolini, F., and Piva, A., 2002, Multichannel Watermarking of Color Images, IEEE Transactions on Circuit and Systems for Video Technology, 12, No. 3, 142-156.

130. Barni, M., Bartolini, F., Piva, A., 2002, Multichannel watermarking of color images**,** IEEE Transactions on Circuits and Systems for Video Technology, 12, No. 3, 142-156.

131. Lin, C. Y., Wu, M., Bloom, J. A., Cox, I. J., Miller, M. L., Lui, Y. M., 2001, Rotation, Scale and Translation Resilient Watermarking for Images, IEEE Transactions on Image Processing, 10, No. 5, 767-782.

132. Joseph, J. K., Ruanaidh, O., Pun, T., 1998, Rotation, Scale and Translation Invariant Spread Spectrum Digital Image Watermarking, Elsevier Signal Processing, 66, 303-317

133. Mallat, S. G., 1989, A Theory for Multiresolution Signal Decomposition: The Wavelet Representation, IEEE Transactions on Pattern Analysis and Machine Intelligence, 11, No. 7. 674-693

134. Rioul, O., and Vetterli, M., 1991, Wavelets and Signal Processing," IEEE Signal Processing Magazine, 8, No. 4, 14-38.

135. Xia, X. G., Boncelet, C., and Arce, G., 1998, Wavelet Transform Based Watermark for Digital Images, Optics Express, 3, No. 12, 497-511.

136. Inoue, H., Miyazaki, A., Katsura, T., 1999, An Image Watermarking Method Based on the Wavelet Transform, International Conference on Image Processing, 1, 296-300.

137. Miyazaki, A., 2003, On the Evaluation of Wavelet Filter Banks for Wavelet-based Image Watermarking, Proceedings of the 3rd International Symposium on Image and Signal Processing and Analysis, 2, 877-882.

138. Hsieh, M. S., Tseng, D., Huang, Y. H., 2001, Hiding Digital Watermarks Using Multiresolution Wavelet Transform, IEEE Transactions On Industrial Electronics, 48, No. 5, 875-882.

139. Costa, M., 1983, Writing on Dirty Paper, IEEE Transactions on Information Theory, 29, 439-441.

140. Swanson, M., D., Zhu, B., Tewfik, A. H., 1999, Transparent Robust Image Watermarking, In Proceedings of the IEEE International Conference on Image Processing, 3, 211-214.

141. Vleeschouwer, C., Delaigle, J. F., Macq, B., 1998, Watermarking Algorithm Based on a Human Visual Model, Signal Processing, 66, No. 3, 319-335.

142. Braudaway, G. W., Margerlein, K. A., and Mintzer, F. C., 1996, Protecting Public-Available Images with a Visible Image Watermark, in Proc. SPIE Conf. Optical Security and Counterfeit Deterrence Techniques, 2659, 126–132.

143. Meng, J. and Chang, S. F., 1998, Embedding visible Video Watermarks in The Compressed Domain, in Proc. IEEE Int. Conf. Image Processing, 1, 474–477.

144. Mohanty, S. P., Ramakrishnan, K. R. and Kankanhalli, M. S., 2000, A DCT Domain Visible Watermarking Technique For Images, in Proc. IEEE Int. Conf. Multimedia and Expo., 2, 1029–1032.

145. Chen; P. M., 2000, A Visible Watermarking Mechanism Using A Statistic Approach, WCCC-ICSP 2000 5th International Conference on Signal Processing Proceedings, 2, 910 – 913.

146. Hu, Y. J. and Kwong, S., 2001, Wavelet Domain Adaptive Visible Watermarking, IEE Electron. Letter, 37, No. 9, 1219–1220.

147. Huang, C. H. and Wu, L, 2004, Attacking Visible Watermarking Schemes, IEEE Transaction On Multimedia, 6, No. 1, 16-30.

148. Lin S. D. and Shie, S. C., 2004, Improving Robustness of Visible Watermarking Schemes for Images, IEEE International Symposium on Consumer Electronics, 11 – 14.

149. Hu, Y. J. and Kwong, S., 2006, An Algorithm for Removable Visible Watermarking, IEEE Transactions On Circuits and Systems for Video Technology, 16, No. 1, 129-133.

150. Kankanhalli, M.S., and Ramakrishnan, K.R., 1999, Adaptive Visible Watermarking of Images, IEEE International Conference on Multimedia Computing and Systems, 1, 568-573.

151. Craver, S., Yeo, B. L., Yeung, M., 1998, Technical Trial and Legal Tribulations, Communications of the ACM, 41, No. 7, 45-54.

152. Voloshynovskiy, S., Pereira, S., Pun, T., 1999, Watermark Attacks, In Erlangen Watermarking Workshop 99.

153. Hartung, F., Su, J., Girod, B., 1999, Spread Spectrum Watermarking: Malicious Attacks and Counterattacks, In Proceedings SPIE Security and Watermarking of Multimedia Contents, 3657, 147-158.

154. Setyawan, I., 2004, Geometric Distortion in Image and Video Watermarking: Robustness and Perceptual Quality Impact, Master of Science in Electrical Engineering, Delft University of Technology, 128 p.

155. G.C. Langelaar, R.L. Lagendijk and J. Biemond, 1998, Watermark Removal based on Non-Linear Filtering, ASCI '98 Conference, Lommel, Belgium.

156. Craver, S., Memon, N., Yeo, B. L., and Yeung, M. M., 1996, Can Invisible Watermarks Resolve Rightful Ownerships, IBM Technical Report RC 20509.

157. Craver, S., Memon, N., Yeo, B. L., and Yeung, M. M., 1997, On the Invertibility of Invisible Watermarking Techniques, in the Proceedings of the International Conference on Image Processing, 1, 540-543.

158. Craver, S., 2002, The Return of Ambiguity Attacks, in Proceedings of SPIE, Security and Watermarking of Multimedia Contents IV, 4675, 252–259.

159. Rey C., Dugelay, J. L., 2000, Blind Detection of Malicious Alterations on Stil Images Using Robust Watermarking, IEE Seminar on Secure Images and Image Authentication , 1-6.

160. Lin, C. Y., Chang, S. F., 2000, Semi-Fragile Watermarking for Authenticating JPEG Visual Content, SPIE Security and Watermarking of Multimedia Contents II, 140-151.

161. Bassali, H. S., Chhugani, J., Agarwal, S., Aggarwal, A., and Dubey, P., 2000, Compression Tolerant Watermarking for Image Verification, IEEE Int. Conf. Image Processing, 1, 430-433.

162. Fridrich, J., 1998, Image Watermarking for Tamper Detection, Proc. IEEE Int. Conf. Image Processing, 2, 404-408.

163. Dittmann, J., Steinmetz, A., and Steinmetz, R., 1999, Content-based Digital Signature for Motion Pictures Authentication and Content-Fragile Watermarking, IEEE Int. Conf. Multimedia Computing and Systems, 2, 209-213.

ÖZGEÇMİŞ

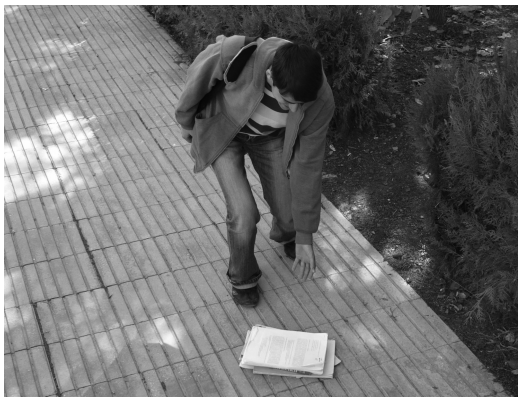
Metin ERTÜRKLER, 1977 yılında Elazığ'da doğdu. İlk öğrenimini İsmet Paşa İlk Öğretim Okulunda, orta ve lise öğrenimini ise Mezre Ortaokulu ve Mehmet Akif Ersoy Lisesi'nde tamamladı.

1995 yılında F.Ü. Teknik Eğitim Fakültesi Elektronik-Bilgisayar Öğretmenliği Bölümünü kazandı ve 1999 yılında aynı bölümden mezun oldu. 1999 yılında F.Ü. Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümünde yüksek lisans yapmaya başladı. 2000 yılında aynı bölümde Araştırma Görevlisi olarak görev yapmaya başladı. Yüksek lisans çalışmalarında Veri Sıkıştırma Teknikleri, Sayısal İmge İşleme ve Gürbüz Filigranlar üzerinde çalıştı. 2001 yılında "Ayrık Kosinüs Dönüşüm Temelli Sayısal Filigranlama" isimli yüksek lisans tez çalışmasını sunarak Yüksek Lisansı tamamladı.

2002 yılında F.Ü. Elektrik-Elektronik Mühendisliği Bölümünde Doktora öğrenimine başlayıp, ***"Sayısal Filigranlar İle Kripto İmzalarının Birlikte Kullanılması ve Çoklu Ortam Verisi Üzerindeki Uygulamaları"*** isimli tez çalışmasını yapmıştır.

EKLER

EK 1



Test imgesi 1



Test imgesi 2



Test imgesi 3



Test imgesi 4



Test imgesi 5

1200*1600 piksellik test imgeleri



Test imgesi 6



Test imgesi 7



Test imgesi 8



Test imgesi 9



Test imgesi 10

768*1024 piksellik test imgeleri



Test imgesi 11



Test imgesi 12



Test imgesi 13



Test imgesi 14



Test imgesi 15

480*640 piksellik test imgeleri



Değiştirilmiş Test imgesi 1



Değiştirilmiş Test imgesi 2



Değiştirilmiş Test imgesi 3



Değiştirilmiş Test imgesi 4



Değiştirilmiş Test imgesi 5

1200*1600 piksellik değiştirilmiş test imgeleri



Değiştirilmiş Test imgesi 6



Değiştirilmiş Test imgesi 7



Değiştirilmiş Test imgesi 8



Değiştirilmiş Test imgesi 9



Değiştirilmiş Test imgesi 10

768*1024 piksellik değiştirilmiş test imgeleri



Değiştirilmiş Test imgesi 11



Değiştirilmiş Test imgesi 12



Değiştirilmiş Test imgesi 13



Değiştirilmiş Test imgesi 14



Değiştirilmiş Test imgesi 15

480*640 piksellik değiştirilmiş test imgeleri



Test 1 imgesi doğrulama



Test 2 imgesi doğrulama



Test 3 imgesi doğrulama



Test 4 imgesi doğrulama



Test 5 imgesi doğrulama

Sıradüzensel Filigranlama Tekniğin 1200*1600 piksellik test imgeleri üzerindeki doğrulama sonucu



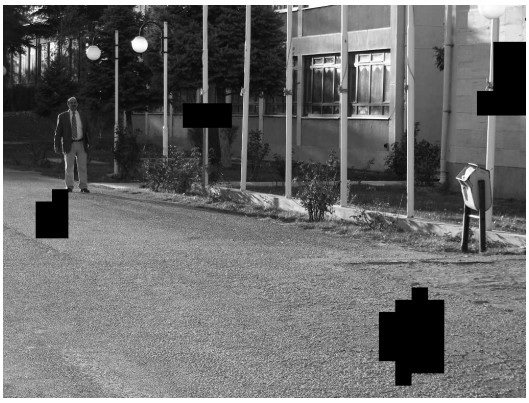
Test 6 imgesi doğrulama



Test 7 imgesi doğrulama



Test 8 imgesi doğrulama



Test 9 imgesi doğrulama



Test 10 imgesi doğrulama

Sıradüzensel Filigranlama Tekniğinin 768*1024 piksellik test imgeleri üzerindeki doğrulama sonucu



Test 11 imgesi doğrulama



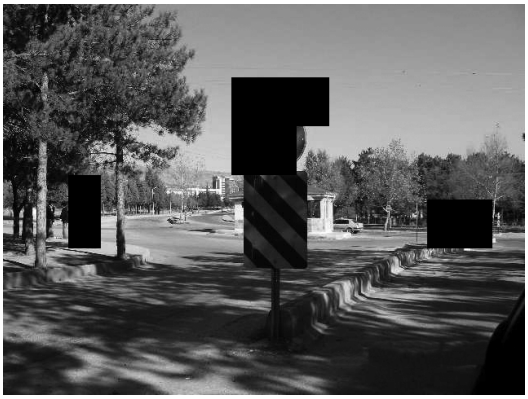
Test 12 imgesi doğrulama



Test 13 imgesi doğrulama



Test 14 imgesi doğrulama



Test 15 imgesi doğrulama

Sıradüzensel Filigranlama Tekniğinin 480*640 piksellik test imgeleri üzerindeki doğrulama sonucu



Test 1 imgesi doğrulama



Test 2 imgesi doğrulama



Test 3 imgesi doğrulama



Test 4 imgesi doğrulama



Test 5 imgesi doğrulama

Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniğinin 1200*1600 piksellik test imgeleri üzerindeki doğrulama sonucu



Test 6 imgesi doğrulama



Test 7 imgesi doğrulama



Test 8 imgesi doğrulama



Test 9 imgesi doğrulama



Test 10 imgesi doğrulama

Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniğinin 768*1024 piksellik test imgeleri üzerindeki doğrulama sonucu



Test 11 imgesi doğrulama



Test 12 imgesi doğrulama



Test 13 imgesi doğrulama



Test 14 imgesi doğrulama



Test 15 imgesi doğrulama

Sıradüzensel Ağaç Yapılı Kırılgan Filigranlama Tekniğinin 480*640 piksellik test imgeleri üzerindeki doğrulama sonucu

EK 2



Test 1 imgesi doğrulama



Test 2 imgesi doğrulama



Test 3 imgesi doğrulama



Test 4 imgesi doğrulama



Test 5 imgesi doğrulama

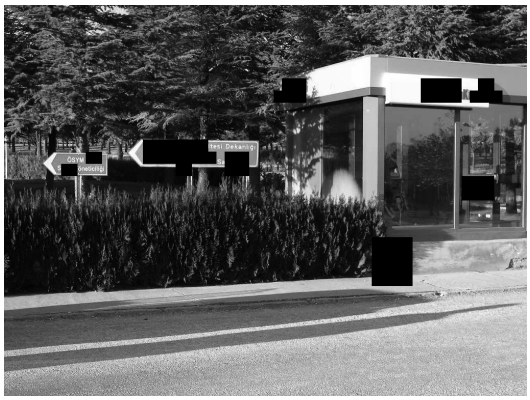
HBC2 Filigranlama Tekniğinin 1200*1600 piksellik test imgeleri üzerindeki doğrulama sonucu



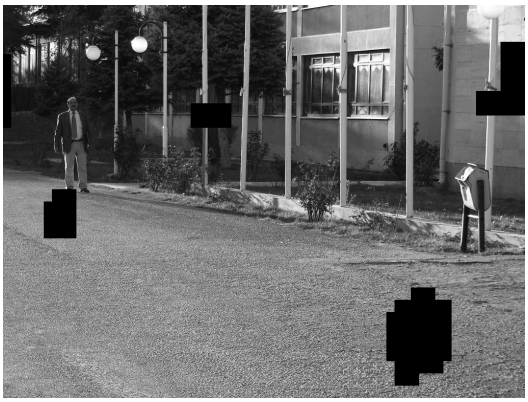
Test 6 imgesi doğrulama



Test 7 imgesi doğrulama



Test 8 imgesi doğrulama



Test 9 imgesi doğrulama



Test 10 imgesi doğrulama

HBC2 Filigranlama Tekniğinin 768*1024 piksellik test imgeleri üzerindeki doğrulama sonucu



Test 11 imgesi doğrulama



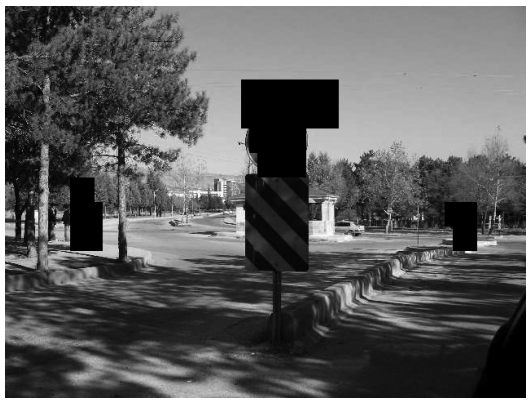
Test 12 imgesi doğrulama



Test 13 imgesi doğrulama



Test 14 imgesi doğrulama



Test 15 imgesi doğrulama

HBC2 Filigranlama Tekniğinin 480*640 piksellik test imgeleri üzerindeki doğrulama sonucu



Test 1 imgesi doğrulama



Test 2 imgesi doğrulama



Test 3 imgesi doğrulama



Test 4 imgesi doğrulama



Test 5 imgesi doğrulama

Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinin 1200*1600 piksellik test imgeleri üzerindeki doğrulama sonucu



Test 6 imgesi doğrulama



Test 7 imgesi doğrulama



Test 8 imgesi doğrulama



Test 9 imgesi doğrulama



Test 10 imgesi doğrulama

Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinin 768*1024 piksellik test imgeleri üzerindeki doğrulama sonucu



Test 11 imgesi doğrulama



Test 12 imgesi doğrulama



Test 13 imgesi doğrulama



Test 14 imgesi doğrulama



Test 15 imgesi doğrulama

Alt Blok Yaklaşımlı Kırılgan Filigranlama Tekniğinin 480*640 piksellik test imgeleri üzerindeki doğrulama sonucu