

**SAYISAL FİLİGRANLAMA ALGORİTMASININ FPGA
ÜZERİNDE GERÇEKLENMESİ**

**DIGITAL WATERMARKING ALGORITHM
IMPLEMENTATION ON FPGA**

AYŞE AYLİN YAZMAN

PROF. DR ALİ ZİYA ALKAR

Tez Danışmanı

Hacettepe Üniversitesi

Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin

Elektrik ve Elektronik Mühendisliği Anabilim Dalı için Öngördüğü

YÜKSEK LİSANS TEZİ olarak hazırlanmıştır.

2023

ÖZET

SAYISAL FİLİGRANLAMA ALGORİTMASININ FPGA ÜZERİNDE GERÇEKLENMESİ

Ayşe Aylin YAZMAN

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Danışmanı: Prof. Dr. Ali Ziya ALKAR

Temmuz 2023, 60 sayfa

İnternetin yaygınlaşması ile görüntü, video, ses ve metin gibi dijital medya verilerinin kullanımı da gelişmektedir. İnternet ortamında, dijital verilerde telif hakkı ihlalleri, izinsiz değiştirilmesi veya kullanılması problemleri ortaya çıkmaktadır. Bu teknolojik gelişim, beraberinde dijital ağlar üzerinden iletilen verilerin aidiyetinin (authentication) doğrulanması ve verinin gürbüzlüğünün (robustness) sağlanması gerekliliğini ortaya çıkmaktadır. Bu tez çalışması ile dijital verilerin izinsiz kullanılması, telif hak ihlali, izinsiz kopyalama ve dağıtımlarının engellenmesi ve verilerin orijinalliğinin korunması gibi sorunların çözümüne yönelik sayısal filigranlama yöntemi ele alınmaktadır.

Filigran uygulamaları üzerine hem donanım ve hem de yazılım ortamında uygulanabilir çalışmalar mevcuttur. Yazılım ile filigranlama işlemi bilgisayar ortamında kolay bir şekilde uygulanabilir. Ancak bu yaklaşım kimi zaman yavaş kalmaktadır ve gerçek zamanlı uygulamaları için uygun değildir. Bu noktada donanım çözümü olan FPGA uygulamaları gerçekleştirilmiştir. Görüntü, video, ses ve metin verilerini güven altına almaya yönelik sayısal filigranlama yöntemleri mevcuttur. Ancak çoğu çalışma görüntü verileri üzerinden yapılmaktadır. Literatürde video üzerinden filigranlama işlemi çok az sayıda bulunmaktadır.

Bu çalışmada özellikle video üzerine uygulanmış sayısal filigranlama yöntemi ele alınmaktadır.

Yapmış olduğumuz bu çalışma ile video üzerinden filigranlama işlemini sadece FPGA kullanarak, daha iyi filigranlama işlemi uygulayarak ve kaliteli filigranlı verilerin elde edilmesi amaçlanmıştır.

Bu çalışmada, görüntü verileri üzerinde yapılan sayısal filigranlama yöntemleri derinlemesine araştırılarak donanım sistemlerine en uygun sayısal filigranlama yönteminin tasarımı amaçlanmaktadır. Analiz ve araştırma sonuçlarından elde edilen sonuçlar baz alınarak en iyi sonucu veren yöntem ayrık dalgacık dönüşümü olarak seçilmiştir.

Bu çalışma ile dijital görüntü filigranlama teknikleri, önerilen yöntemin uygulanabilirliği ve donanım üzerinde çalışma prensibi gösterilmektedir. Yapılan çalışma sonuçları ile önerilen yöntemin, gerçek zamanlı uygulama esasına dayalı alanlarda uygulanabilirliğini göstermektedir.

Anahtar Kelimeler: Filigranlama, FPGA'da filigran tasarımı, Donanım, Dalgacık dönüşümü, Görüntü işleme, Video İşleme

ABSTRACT

DİGİTAL WATERMARKİNG ALGORİTHM İMPLEMENTATİON ON FPGA

Ayşe Aylin YAZMAN

Master of Science, Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. Ali Ziya ALKAR

July 2023, 60 pages

With the widespread use of the Internet, the use of digital media data such as images, video, audio and text are also developing. In the internet environment, problems of copyright infringement, unauthorized modification or use of digital data arise. This technological development brings with it the necessity of verifying the authenticity of the data transmitted over digital networks and ensuring the robustness of the data. In this thesis, the digital watermarking method for solving problems such as unauthorized use of digital data, copyright infringement, preventing unauthorized copying and distribution, and preserving the originality of the data is discussed.

There are studies on watermark applications that can be applied in both hardware and software environments. The watermarking process with the software can be easily applied in the computer environment. However, this approach is sometimes slow and not suitable for real-time applications. At this point, FPGA applications, which are hardware solutions, have been realized. Digital watermarking methods are available to secure image, video, audio and text data. However, most studies are done on image data. There are very few watermarking processes on video in the literature. In this study, digital watermarking method applied especially on video is discussed.

With this study we have done, it is aimed to obtain quality watermarked data by using only FPGA watermarking process over video, by applying better watermarking process.

In this study, it is aimed to design the most suitable digital watermarking method for hardware systems by investigating digital watermarking methods on image data in depth. Based on the results obtained from the analysis and research results, the method giving the best result was chosen as the discrete wavelet transform.

In this study, digital image watermarking techniques, applicability of the proposed method and working principle on hardware are shown. The results of the study show the applicability of the proposed method in areas based on real-time application.

Keywords: Watermarking, Watermarking on FPGA, Xilinx, Hardware, Discrete Wavelet Transform, Image processing

TEŐEKKÜR

Tez alıőmamn bilgi, grőő ve yorumlarıyla yardımcı olan danıőmanım Prof. Dr. Ali Ziya ALKAR'a teőekkr ederim.

Maddi ve manevi desteklerini esirgemeyen aileme ve beni her zaman destekleyen sevgili eőime teőekkr ederim.

İÇİNDEKİLER

ÖZET	i
ABSTRACT.....	iii
TEŞEKKÜR.....	v
İÇİNDEKİLER	vi
ŞEKİLLER DİZİNİ	viii
TABLolar DİZİNİ.....	ix
SİMGELER VE KISALTMALAR	xi
1. GİRİŞ	1
1.1. Literatür Taraması.....	4
1.2. Tezin Amacı.....	6
2. FİLİGRAN YÖNTEMLERİ VE YAKLAŞIMLARI	8
2.1. Verinin Gizlenmesi Teknikleri	11
2.2. Filigran Algoritmasının Sınıflandırılması.....	13
2.2.1. Verinin Formuna Göre Filigranlama	14
2.2.2. Verinin İnsan Algısına Göre Filigranlama.....	15
2.2.3. Verinin Tekniğine Göre Filigranlama.....	15
2.3. Sayısal Filigranlamada Kullanılan Teknikler	16
2.4. Sayısal Filigranlamanın Tercih Edilme Nedenleri.....	18
2.5. Sayısal Filigranlamanın Kullanım Alanları	18
3. FİLİGRAN ALGORİTMASININ FPGA UYGULAMASI.....	20
3.1. MATLAB Modülü	28
3.2. Dalgacık Dönüşüm Modülü – DWT.....	29
3.3. Ters Dalgacık Dönüşüm Modülü – IDWT Haar	43
3.4. Block RAM Modülü (BRAM).....	47
3.5. Sonuçların Değerlendirilmesi	47
4. SONUÇLAR VE ÖNERİLER.....	53
5. KAYNAKLAR	54

EKLER	59
EK 1 – Donanım Tasarıma ait Şematik.....	59
EK 2 – Tez Çalışması Orjinallik Raporu.....	59

ŞEKİLLER DİZİNİ

Şekil 2.1 Filigran Gömme İşlemi	10
Şekil 2.2 Filigran Çıkarma İşlemi	10
Şekil 2.2.1 Filigran Algoritmasının Sınıflandırılması	14
Şekil 2.2.1.1 Filigran Algoritmasının Veri Formuna Göre Sınıflandırılması	14
Şekil 2.2.2.1 Filigran Algoritmasının Verinin İnsan Algısına Göre Sınıflandırılması.....	15
Şekil 2.2.3.1 Filigran Algoritmasının Verinin Tekniğine Göre Sınıflandırılması.....	16
Şekil 3.1 FPGA'nın İç Yapısı.....	23
Şekil 3.2 Gömme İşleminin Temel Uygulanışı	24
Şekil 3.3 Çıkarma İşleminin Temel Uygulanışı	24
Şekil 3.4 Zynq Ultrascale+ serisi ZCU106 geliştirme kartı	25
Şekil 3.5 Sentez Sonuçları.....	25
Şekil 3.6 Tasarımın Genel Yapısı.....	27
Şekil 3.7 FPGA İç Tasarım Detayı.....	28
Şekil 3.8 1.Seviye Dönüşüm	31
Şekil 3.9 2.Seviye Dönüşüm	32
Şekil 3.10 Filigran Görüntüsünün 1. ve 2. seviye Yatay, Dikey ve Köşegene ait Dönüşüm Sonuçları.....	35
Şekil 3.11 Filigran görüntüsünün 1. seviye Dönüşüm Sonuçları	36
Şekil 3.12 Filigran görüntüsünün 2. seviye Dönüşüm Sonuçları	37
Şekil 3.13 Ana Görüntünün 1. ve 2. seviye Yatay, Dikey ve Köşegene ait Dönüşüm Sonuçları	37
Şekil 3.14 Ana (Lena) görüntünün 1. seviye Dönüşüm Sonuçları	38
Şekil 3.15 Ana (Lena) görüntünün 2. seviye Dönüşüm Sonuçları	39
Şekil 3.16 Filigranlama İşlemi Mimarisi.....	42
Şekil 3.17 Filigranla Çıkarma İşlemi Mimarisi	46
Şekil 3.18 Block RAM Yapısı.....	47
Şekil 3.19 Sistem Tasarımı.....	48

TABLÖLÄR DİZİNİ

Tablo 1 Kaynak Kullanım Sonuçları	26
Tablo 2 Kurcalama Saldırılarına ait Örnek Çalışma.....	49
Tablo 3 Kurcalama Saldırılarına ait Gerçek Zamanlı Akan Videodan Örnek Çalışma....	50
Tablo 4 Veri Doğrulama Kapsamında Örnek Çalışma	51
Tablo 5 Veri Doğrulama Kapsamında Gerçek Zamanlı Akan Videodan Örnek Çalışma	52
Tablo 6 Filigranlı Görüntüden Ana Görüntünün Çıkarılması İşlemi	52

SİMGELER VE KISALTMALAR

Kısaltmalar

AES	Gelişmiş Şifreleme Standardı
DES	Veri Şifreleme Standardı
LSB	En Değersiz Bit
MSB	En Önemli Bit
DCT	Ayrık Kosinüs Dönüşümü
IDCT	Ters Ayrık Kosinüs Dönüşümü
DFT	Ayrık Fourier Dönüşümü
DWT	Ayrık Dalgacık Dönüşümü
IDWT	Ters Ayrık Dalgacık Dönüşümü
DSP	Sayısal Sinyal İşleme
FPGA	Alanda Programlanabilir Kapı Dizileri
ASIC	Uygulamaya Özel Entegre Devreler
HDL	Donanım Tanımlama Dili
VHDL	Çok Yüksek Hızlı Tümlleşik Devre Donanım Tanımlama Dili
DHT	Ayrık Hadamard Dönüşümü
PSNR	Tepe Sinyal Gürültü Oranı
IoT	Nesnelerin İnterneti
LUT	Look Up Table
FF	Flip Flop
BRAM	Block RAM
IO	Input Output
BUFG	Clock Buffer

MMCM

Mixed-Mode Clock Manager

1. GİRİŞ

Medya sistemlerinin teknoloji ile birlikte gelişmesiyle dijital verilerin daha kapsamlı şekilde kullanılması gündeme gelmektedir. Bu modern teknoloji, beraberinde olumlu ve olumsuz yönler getirmektedir. Bu teknolojik gelişmelerin yanlış kullanımı ile siber suçların oluşmasına, telif hakkı sorunlarına ve yanlışlıcağa yol açmaktadır. Dijitalleşen veriler, haberleşme kapsamında iletişimde kolay bir araç haline gelmektedir. Verilerde düzenleme, çoğaltma gibi işlemler kullanıcılar tarafından kolaylıkla yapılabilmektedir. Dijital iletişim ve bilgisayar teknolojisinin gelişmesiyle birlikte, bilginin yayılımı ve erişimi giderek daha kolay ve hızlı hale gelmektedir. Kullanıcılar, dijital multimedya (görüntü, ses, metin ve video gibi) verilerini internet aracılığı ile kolayca indirebilmektedir. Dolayısı ile korsan veri ve telif hakkı sorunları da problem haline gelmektedir. Çeşitli şifreleme teknikleri ile verilerin haberleşme sırasında güvenle iletimi sağlanabilmektedir; ancak, dijital verilerin bütünlüğünün korunması ve yetkisiz kopyalamaların önüne geçilmesi konusunda eksiklikler bulunmaktadır. Bu işlemler, fikri mülkiyet haklarının korunmasında yaygın bir sorun yaratmaktadır. Verileri, yetkisiz erişimden korumak adına bir dijital yöntem ihtiyacı duyulmaktadır.

Sayısal filigranlar, telif hakkı sahibinin bilgisini içermektedir ve verilerin yasal sahipliğini doğrulayabilmektedir. Bunun için logo, seri numaraları, karakterler, görüntü sembolleri kullanılabilir. Sayısal filigran, telif hakkı, dijital haklar ve bu ürünlerin içeriğinin bütünlüğünü belirlemeye yardımcı olmak için dijital ürünlere gizlice gömülebilmektedir. Bilgi gizleme işlemi, sayısal filigran teknolojisinin temelini oluşturmaktadır.

Bu nedenle, tamamlayıcı bir şifreleme teknolojisi olan sayısal filigranlama teknolojisi, telif hakkı koruması için etkili bir teknik olarak ortaya çıkmıştır [1]. Yetkisiz erişim, telif hakkı koruması gibi sorunları çözebilmek adına sayısal filigranlama teknolojisi geliştirilmiştir.

Günümüzde sayısal filigranlama, medya bilgilerinin üzerinde oynama yapılmış verilerden uzak durmak ve orijinalliğini doğrulayabilmek adına hızlı bir şekilde yaygınlaşmaktadır. Sayısal filigranlar, bir dijital içeriğin telif hakkını korur ve telif hakkı ihlalini önler. Ek olarak, dijital medyanın geçerliliğini sağlamak için de kullanılır ve bu şekilde, dijital ortam korsanlığı azaltılır ve güvenlik arttırılmaktadır.

Filigranlama yani veri gizleme terimi, içerisinde gizlenmiş olan veriyi temsil etmektedir. Bu gizli veriler filigranlama işlemi sonrasında güvenlik maksadıyla ilgili veriden ayıklanmakta ve algılanmaktadır. Filigran, göz ile algılanamayan bir gizli kod olarak karakterize edilmektedir. Verinin üzerine gömülmüş olan gizli bilgiyi ifade etmektedir.

Tüketicilerin fikri mülkiyet haklarının yeterince farkında olmaması nedeniyle multimedya içeriğinin korunması son zamanlarda önemli bir konu haline gelmiştir. Sayısal filigranlama teknolojisi, günümüz teknolojisi ile birlikte gelişmeye devam etmektedir. Telif hakkı gibi mülkiyet sorunlarını çözmek ve izinsiz erişilmiş olan verilerin kopyalanması, düzenlenmesi, dağıtılması gibi işlemlerin önüne geçmek amacıyla sayısal filigranlama teknolojisi giderek yaygınlaşmaktadır.

Teknolojinin gelişmesi ile birlikte veriler üzerinde yapılan kopyalama ve dağıtılma işlemleri hızlı şekilde artış göstermektedir. Medya üretici firmaların haklarının korunumu veya askeri ve tıbbi uygulamalarda medyanın içeriğinin güvenliğinin sağlanması gibi konularda çeşitli filigranlama yöntemleri geliştirilmiştir. Bunun sonucu olarak farklı uygulama alanlarına özel olarak geliştirilmiş birden çok sayısal filigranlama yöntemi ortaya çıkmıştır.

Sayısal filigranlama teknolojisi, görüntü, video, müzik, metin vb. dijital ortamların yasal mülkiyet ve içerik doğrulama gibi temel sorunlarını çözmek için büyük bir önem sağlamaktadır. Bu sorunlar, son yıllarda internet ve bilgisayar teknolojisinin gelişmesi nedeniyle ortaya çıkmıştır. Bu teknolojilerin bir araya gelmesindeki gelişmeler, verilerin sınırsız kopyalanması ve internette paylaşılması için bir ortam sağlamaktadır [2].

Sayısal filigranlama ile benzer ilişkiye sahip olan şifreleme yöntemleri, dijital verilerin korunumu konusunda tek başına yeterli değildir. Bu yöntemler farklı amaçlara hizmet etmektedir. Şifreleme, yalnızca yetkili tarafların bilgileri anlayabilmesi için kullanılan, verileri karıştırma prensibine dayanan bir yöntemdir.

Örnek olarak; insan tarafından okunabilen düz bir metin verisini, şifreli yani anlaşılabilir bir metine dönüştürme işlemidir. Şifreleme işlemi ile okunabilir veriler alınarak, rastgele görünecek şekilde değiştirilmektedir. Şifreleme işlemi sonrasında kullanıcı, şifrelenmiş veri üzerinde değişiklik yapma olanağına sahiptir, fakat işlem yapabilmesi için şifre çözümü işlemi gerekmektedir. Bu şifreleme işlemleri sayesinde, orijinal görüntüden rastgele gürültü oluşturulmaktadır ve bu şifreleme anahtarına sahip olmayan kullanıcılar tarafından tanınmaz hale gelmektedir. Bunun yanı sıra, dijital veri içerisinde uygulanan şifre bir kez çözüldüğü durumda, sonrasında herhangi bir koruma konusunda devamlılığı kalmamaktadır.

Sayısal filigranlama teknolojisi, şifrelemede değinilmeyen veri bütünlüğüne yönelik eksiklikleri tamamlamak amacıyla ortaya çıkmış bir teknoloji olarak düşünülmektedir [3,4,7]. Dijital verinin içerisine, veriyi görsel olarak direkt bozmadan eklenen filigran işlemi ile tüm süre boyunca veri ile birlikte varlığını sürdürmektedir. Bu sayede kopyalama ve dağıtım gibi girişimlere karşı etkin bir koruma sağlamaktadır [4-6].

Belirli bir verinin içerisine görüntü, video, ses, metin gibi verilerin gömülmesi işlemine *filigran gömme* denilmektedir. Belirli bir verinin içerisinden görüntü, video, ses, metin gibi verilerin çıkarılması işlemine ise *filigran çıkarma* denilmektedir. Uygulanan filigranlama işlemi sonrasında dijital veriye filigranlı veri denmektedir. Sayısal filigran, multimedya verilerini korumak için telif hakkı bilgilerini içeren bilgi sinyali olarak ortaya çıkarmaktadır [8]. Veri kimliğini doğrulamak için, elde edilmiş görüntüden filigran çıkarılabilmektedir. Bu şekilde verinin aslına yönelik bir doğrulama (authentication) yapılmaktadır.

1.1. Literatür Taraması

Literatürde sayısal verilere uygulanabilir birçok sayısal filigranlama sistem tasarımı bulunmaktadır.

Sayısal filigranlama terimi 1988 yılında Komatsu ve Tominaga tarafından ilk kez bahsedilmiştir [9].

Schyndel, Tirkel ve Osborne 1994 yılında, uzamsal alanda LSB ile gerçekleştirilen bir uygulama yapmışlardır [3]. Ancak bu yöntem, saldırılara karşı zayıf kalmıştır.

Filigranlama sürecinde en sıklıkla kullanılan yöntemlerden biri olan Dalgacık dönüşümü ile ilgili ilk literatür ise matematikçi Alfred Haar tarafından 1909'da önerilmiştir [10].

Koch, Rindfrey ve Zhao, görüntüyü bloklara bölerek, her bloğa DCT uygulayarak ve orta frekans bölgesi ile çalışarak telif hakkı korunmasında bir çalışma gerçekleştirilmiştir [11]. Shieh, Huang, Wang ve Pan, DCT kullanarak görünmez bir filigranlama çalışması gerçekleştirmişlerdir [12]. Bu algoritma gürbüz ve görünmez filigranlama olarak önerilmiştir.

Lim, Hyun ve Cho, dijital kameralar için görünmez filigranlama tekniğini tasarlamak için DCT ve LSB tekniğini kullanmışlar ve FPGA üzerinde uygulayarak test etmişlerdir [13]. Filigranın doğruluğunu kanıtlamak için frekans alanında DCT tekniğini kullanarak bir filigranlama çalışması gerçekleştirmişlerdir.

Tamilvanan ve Selvakumar, filigran tekniğinin sağlamlığını ve algılanabilirliği kapsamında LSB tekniğini çalışmıştır ve filigranlama tekniğini FPGA üzerinde uygulamıştır [14].

Sonjoy Deb Roy ve diğeri, görünmez ve yarı kırılabilir filigran bilgilerini gerçek zamanlı olarak sıkıştırılmış video akışlarına ekleyebilen DCT tekniğı ile FPGA tabanlı donanım çalışması önermektedir [15]. Çalışma, büyük hesaplamalar gerektirdiğı için genellikle yüksek güç tüketen ve aynı zamanda kalitenin düşmesine neden olan ayrık kosinüs dönüşüm (DCT) alanı kullanılarak gömme filigranı tartışmaktadır.

Enas Duhri Kusuma ve diğeri, donanım uygulaması için yüksek güç tüketimine neden olan DCT tabanlı görüntü sıkıştırmasına ait FPGA tabanlı çalışmayı konu almaktadır [16].

Pankaj U. Lande ve diğeri, Ayrık Hadamard Dönüşümü (DHT) alanındaki damgalama algoritmasını tanımlamaktadır [17]. Makale ayrıca, uzamsal alanda sayısal filigranlama için DCT/IDCT uygulamaları ele almaktadır.

Rajesh Kannan Megalingam ve diğeri, MATLAB, Verilog HDL ve FPGA donanımı kullanarak LSB ve DCT teknikleri ile görüntü filigranlama yönteminin uygulanmasını açıklamaktadır [18]. Tepe sinyal gürültü oranı (PSNR) hesaplamaları yapılmıştır.

S. Sowmya ve diğeri, iletişim sırasında bilgi kaybını önlemek amacıyla filigranlama tekniklerinin FPGA uygulaması önermektedir [19]. Gerçekleştirdikleri donanım uygulamalarında algoritmanın işlem süresi 5 ms olarak belirlenmiştir.

P Karthigaikumara ve diğeri, kırılabilir ve yarı kırılabilir filigranlama tekniklerini ve artan kaynak kullanımı, daha geniş alan gereksinimlerini ve yüksek güç tüketimi gibi bazı ciddi dezavantajları açıklamaktadır [20]. Bunun üstesinden gelmek için, DWT algoritması kullanılarak gürbüz ve görünmez filigranlama tekniğı kullanılmaktadır.

Wael Wasfya ve diğeri yaptıkları çalışmada, görüntü yoğunluğunun hesaplanması sırasında hızlı görüntü işleme algoritmaları için hızın ve doğruluğun artırılmasını hedeflemiştir [21]. Bu çalışmada ayrıca, FPGA içindeki sayısal sinyal işleme (DSP) biriminin kullanımı ile, görüntü algoritmaları için donanımsal olarak hızlı bir yöntem önerilmektedir. DSP biriminin avantajı, hesaplamalarda daha hızlı bir şekilde, yüksek doğruluk ve miktarda bitin işlenebilmesidir. Algoritma hesaplamaları sırasında daha fazla sayıda bit kullanmak, aynı görüntü algoritması hesaplamalarını daha az bit ile yapmaya kıyasla daha yüksek doğruluk sağlamaktadır. Ayrıca bu yaklaşım FPGA kaynak kullanımını mümkün olduğunca minimuma indirmektedir.

Mohammad-Reza Keyvanpura, filigranlama kapsamında çeşitli dönüşüm alanı tekniklerinin karşılaştırılmasını incelemektedir [22]. Bu çalışma, farklı varyanslar ve farklı yöntemler için başarısız olan kimlik doğrulama algoritmasının tasarlanmasında DWT'nin kullanımını ele almaktadır.

Garimella ve diğeri, uzamsal alanda bir filigranlama algoritmasının donanımsal uygulamasına yönelik Çok Büyük Ölçekli Entegrasyon (VLSI) mimarisi kapsamında gerçekleştirilen önemli bir çalışma olarak kabul edilmektedir [23].

Mohanty ve diğeri, LSB tekniği kullanarak JPEG görüntülere iki görünür filigran ekleyebilen bir VLSI tasarımı önermektedir [24]. Başka bir çalışmada Mohanty ve diğeri, bu kez DCT alanındaki dijital görüntülere görünmez veya görünür filigranlar ekleyebilen bir VLSI mimarisi ortaya koymaktadır [25].

1.2. Tezin Amacı

Teknolojinin artan popülaritesi ve bilgi işlemenin gelişmesiyle birlikte bilgi güvenliğinin sağlanması önemli rol oynamaktadır. Veri şifreleme, bir ağ ortamında veri güvenliğini sağlamak için önemli bir yöntemdir. Ancak verilerin açık olarak iletildiği ortamlarda ağ iletimindeki dijital görüntüler, bilgi sahipleri için büyük kayıplara neden olan izinsiz bilgi kopyalama, veri kurcalama ve veri silme dahil olmak üzere çeşitli insan yapımı saldırılara açık hale gelmektedir.

Bu nedenle iletişim hattından açık olarak gönderilen bir görüntünün, sesin veya bilginin izinsiz erişim ve değişimlerden korunmasının nasıl sağlanacağı önemli bir konu olarak önemini korumaktadır. Dijital verilerin içeriğinin korunmasını sağlamak için filigranlama yöntemleri önerilmektedir. Sayısal filigranlama teknolojisi ile telif hakkını etkili bir şekilde korumak için görüntünün tamamına filigran yerleştirilen çalışmalar önemini korumaktadır.

Yazılımsal olarak 25 yılı aşkın bir süredir bu konuda birçok çalışma yapılmıştır. Bu tez çalışmasındaki amaç, yazılımsal olarak kullanılan yöntemlere alternatif olarak donanım üzerinde filigranlama işleminin uygulanmasıdır. Bu amaçla bir donanım sistemi olan FPGA kullanılacaktır. Sayısal filigranlama algoritmasının donanım üzerinde gerçekleşmesiyle, düşük güç tüketimi, daha az alan kullanımı ve güvenilirlik açısından yazılım uygulamasına göre avantaj sağlanmaktadır. Donanım filigranlama çözümü ekonomiktir, çünkü işlemin yapılması, yalnızca kullanılan FPGA silikonun ayrılmış küçük bir alanını kaplamaktadır. Gerçekleştirilen bu çalışmada, algoritma MATLAB kullanılarak geliştirilmiş ve simüle edilmiştir. Sonrasında bu algoritma FPGA üzerinde uygulanmıştır.

Bu tezin diğer bir amacı da göreceli olarak daha az çalışılan bir alan olan filigranlama işleminin video verisi üzerinde gerçekleştirilmesidir. Donanımsal olarak elde edilen hız avantajının video işlemlerinde kullanılması olası hale gelmiştir.

İlerleyen bölümlerde uygulama yöntemleri ve gerçekleştirilen çalışmalar sırası ile anlatılmaktadır. İkinci bölümde filigranlama işleminin tekniklerine, sınıflandırılmasına, yöntemlerine, tercih edilme nedenlerine ve kullanım alanlarına yer verilmektedir. Üçüncü bölümde, FPGA üzerinde yapılan çalışmalar ve gerçekleştirilen donanım tasarımı anlatılmaktadır. Son olarak dördüncü bölümde yapılan tez çalışması ile ilgili sonuçlar ve öneriler verilmektedir.

2. FİLİGRAN YÖNTEMLERİ VE YAKLAŞIMLARI

Şifreleme, yayınlanacak bilgilerin güvenliğini sağlamak için ideal çözüm olduğundan, klasik şifreleme teknikleri, telif hakkı koruması bağlamında yetersiz kalmaktadır. Gelişmiş Şifreleme Standardı (AES), Veri Şifreleme Standardı (DES) vb. gibi geleneksel şifreleme teknikleri verilerin açık olarak iletildiği ortamlarda veri güvenilirliğini sağlama amacıyla kullanılamamaktadır.

Filigran sistemleri, yazılım ve donanım olmak üzere çeşitli ortamlar aracılığıyla uygulanabilmektedir. Kullanılacak ortamın avantaj ve dezavantajları bulunmaktadır. Yazılımda, filigran bir bilgisayar ortamında kolayca veriye uygulanabilmektedir. Filigran algoritmasının işlemleri, komut dizileri veya gömülü bir işlemci üzerinde çalışan makine kodu yazılımı olarak gerçekleştirilebilmektedir. Yazılım kodunu programlayarak ve mevcut yazılım araçlarını kullanarak, herhangi bir filigran algoritması herhangi bir karmaşıklık düzeyinde kolayca görüntü, ses, video ve metin gibi medya araçlarına uygulanabilmektedir. Ancak böyle bir uygulamanın hızı yavaştır ve bu nedenle özellikle gerçek zamanlı uygulamalar için kullanılamamaktadır. Düşük güç tüketimi, alandan tasarruf, hızlı performans, bilgisayara bağımsız uygulanması, tekrar programlanabilir olması sebebiyle donanım uygulamaları yazılım uygulamalarına göre avantaj sağlamaktadır.

Potansiyel olarak bakıldığında hızlı ve düşük maliyetli bir filigranlama algoritması medya cihazlarında yazılım veya donanım olarak uygulanabilmektedir [26]. Ancak özellikle gerçek zamanlı medya uygulamalarının, donanım üzerinde gerçekleştirilmesi daha uygundur. Kullanılan filigran tekniklerinin donanım uygulamaları, tasarım karmaşıklığı göz önüne alındığında kullanıcıya kolaylık sağlamaktadır. Donanımda, filigranlama uygulamaları genellikle alan programlanabilir kapı dizileri (FPGA) gibi özel olarak tasarlanmış devreler üzerinde gerçekleştirilmektedir. FPGA kullanıldığı durumlarda, görüntü işleme uygulamalarında, kullanıcıya tekrar tasarlayabilme ve gürbüzlük/güvenilirlik sağlamaktadır [27].

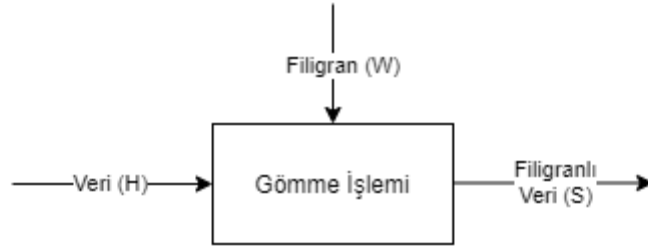
FPGA'lar, otomotiv, iletişim, endüstriyel otomasyon, motor kontrol ve tıbbi görüntüleme gibi gömülü uygulamalarda yaygın olarak kullanılmaktadır [28]. FPGA'lar, tüm sistemi tek bir çipte olacak şekilde tasarlanmasına olanak sağlayabilmekte ve sistemin iç testine ve hata tespitine de izin vermektedir [29].

Çeşitli filigranlama yaklaşımları literatürde 25 yılı aşkın bir süre içinde geliştirilmiştir. Geçmiş araştırmalarda gürbüzlük konusunun, filigranlama algoritmalarında önemli bir role sahip olduğu görülmektedir [30]. Bununla birlikte, gürbüz bir filigranlama tekniği geliştirmek için, kullanılan filigranın uzamsal alandaki en önemli bitlere (MSB) ve dönüşüm alanındaki düşük frekans bandı katsayılarına gömülmesi gerektiği ortaya konulmaktadır.

Filigranlama işlemi, filigran gömme ve filigran çıkarma işlemlerinden oluşmaktadır. Gömme işleminde, amaç filigranı ana veriye eklemek veya yerleştirmektir. Çıkarma işleminde ise amaç, orijinal verinin içeriğini yeniden elde etmektir.

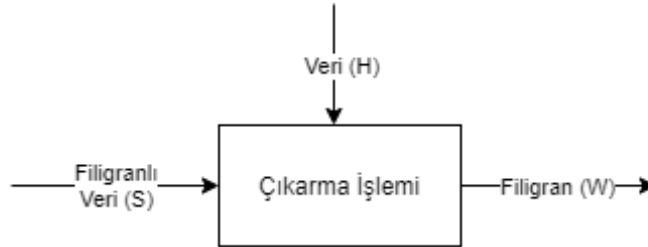
Eklenen filigran, orijinal içeriği değiştirmemelidir. En azından dışarıdan bakıldığında herhangi bir değişiklik görülmemelidir. Filigranlama işlemi, orijinal görüntünün boyutunu büyütmediğinden, mümkün olduğu kadar çok bilgi eklemek tercih edilmektedir. Bununla birlikte, kullanıcı ne kadar fazla bilgi eklerse, içeriğin algı kalitesi üzerinde olumsuz etkisi olabilir. Algı açısından, filigran, orijinal içeriğin işleme sonrası oluşabilecek bozulmayı tolere edecek kadar gürbüz olmalıdır. Bununla beraber gömülü filigran da filigranlı görüntüden çıkarılamayacak şekilde güvenli olmalıdır.

Sayısal filigranlama, bir dijital verinin (filigran) ses, görüntü, video ve metin olabilecek bir başka dijital veriye gömüldüğü süreçtir. Filigranlama işlemi, ilk olarak orijinal veri üzerine dışarıdan oluşturulmuş bir filigran verisinin eklenmesi ile oluşmaktadır. Bu sürece ait temel işlem Şekil 2.1'deki gibi gösterilmiştir.



Şekil 2.1 Filigran Gömme İşlemi

Filigran çıkarma işlemi ise filigranlı veriden, var olan orijinal verinin çıkarılması ile ilgili süreçtir. Bu sürece ait temel işlem Şekil 2.2’deki gibi gösterilmiştir. Görüldüğü üzere filigran çıkarma işleminde orijinal veri olmadan filigranın aslına erişilememektedir.



Şekil 2.2 Filigran Çıkarma İşlemi

Filigranlama işlemi, veri doğrulama ve veriye yapılan kurcalama saldırılarına karşı etkili bir çözüm olmuştur. Filigranlı veri üzerine yapılan saldırılar karşısında, verinin içerisinde varlığını sürdürmesi sebebiyle filigran bu saldırılardan etkilenmemektedir. Filigranlama işleminin veri iletimindeki gürbüz olması özelliği ortaya konulmaktadır

[27].

2.1. Verinin Gizlenmesi Teknikleri

Teknolojinin gelişmesi ile birlikte günümüzde çeşitli veri gizleme modelleri bulunmaktadır:

Steganografi (Gizli yazı): Mesaj bir dosyada gizlenmektedir. Belgelerdeki bilgileri gizlemeye bilgi gizleme veya steganografi denmektedir. Steganografi, tespit edilmekten kaçınmak için gizli bir veriyi sıradan/gizli olmayan bir dosya veya mesaj içinde saklama tekniğidir [31]. Bu gizli veriler daha sonra hedefinde çıkarılmaktadır.

Steganografi, metin, resim, video veya ses içeriği dahil olmak üzere hemen hemen her türlü dijital içeriği gizlemek için kullanılabilir; gizlenecek veriler hemen hemen her türlü dijital içeriğin içine gizlenebilir durumdadır. Steganografi, gizli bir mesaj veya şifre iletmek isteyenler tarafından uygulanmaktadır. Steganografinin pek çok kullanımı olsa da kötü amaçlı yazılım geliştiricilerin, kötü amaca sahip kodun iletimini gizlemek için steganografıyı kullandıkları da bulunmaktadır.

Kriptografi (Şifreleme): Anlamını gizlemek için bir mesajı bir koda karıştırmaktadır. Mesajın karıştırılması gizli anahtar yardımıyla yapılmaktadır. Karıştırılmış mesaja şifreli mesaj denilmektedir ve çözülmesi için gizli şifreye ihtiyaç bulunmaktadır. Kriptografi mesajın güvenliğini sağlamaktadır. Kriptoloji, güvenli iletişim bilimine denmektedir. Kriptografi, gizli anlamı olan mesajlar yaratma bilimine denir. Kriptografi, yalnızca gönderen ve alıcı tarafından anlaşılabilir bir tür güvenli iletişim olan gizli yazıdır. Verilerin iletilmediği bilinse de bu verilerin içeriği diğer kullanıcılar tarafından bilinmemektedir [32].

Filigranlama (Damgalama): Ek bilgilerin orijinal içeriğe veya ana sinyale doğrudan yerleştirilmesi işlemidir. İdeal bir işlemde, filigranlı ve orijinal sinyal arasında algılanabilir bir fark olmamasıdır. Filigranın ana veriye zarar vermeden çıkarılması veya değiştirilmesi zor olmalıdır. Damgalama, veriler hakkındaki bilgileri gömmek için bir veri parçasını fark edilmeden değiştirme uygulamasıdır. Filigranlama, telif *hakı koruması* ve *kimlik doğrulama* olmak üzere iki ana uygulama alanı için tasarlanmıştır.

Görüntü işleme teknolojisine ait çeşitli alanlarda uygulamalar bulunmaktadır. Veri içeriğinin değiştirilmesi, özellikle yasal davalar söz konusu olduğunda problem yaratan bir konu haline gelmektedir [33]. Bu nedenle, kurcalanmış görüntülerin tespiti ile ilgili araştırmalar önemli bir konu haline gelmektedir. Yaygın olarak kullanılan bir yaklaşım olan sayısal filigranlama tekniğidir. Bu teknik, veriyi göndermeden önce gizli bilgileri, orijinal veriye ekleme işlemidir. Orijinal veri, gizli bilgilerin eklendiği veridir; gizli veriler ise bir filigrandır [34-36].

Haberleşme sistemlerinde genellikle Steganografi ve Kriptografi kullanılmaktadır. Kriptografi işleminde haberleşme esnasında bilgi, istenmeyen alıcılar tarafından okunamayacak hale dönüştürülme yani şifreleme tekniğidir. Veriler karşı tarafa şifrelenmiş ve içeriği kolayca belirli olmayan şekilde ulaşmaktadır. Bilgi korunumu ve güvenliği günümüzde olduğu kadar eski yıllarda da önemli bir konu olmuştur. Bu sebeple, Kriptografi çok eski dönemlerden itibaren uzanan bir bilimdir. 2.Dünya Savaşı zamanı, Alman ordusu Enigma cihazı ile şifreli metin ile iletişimi sağlamıştır [37]. Bu cihaz ile şifreli metinler Mors kodları kullanılarak iletilmiştir. Bu örnek, iletişimde şifreleme ihtiyacının önemli bir örneğini temsil etmektedir.

Steganografide, mesajın kendisi gizli tutulmaktadır. Ancak filigranlamada, mesajın gizli olması gerekmemektedir. Çoğu zaman sayısal filigranlama ve steoganografi terimlerini birbirinden ayırmakta zorluk çekilmektedir. Kriptografi kavramı, bu veri güvenliği yaklaşımlarından tamamen farklı tanımlanmaktadır. Kriptografide dijital içerik, vericide bir anahtar kullanılarak şifrelenmektedir ve yalnızca ve ancak doğru anahtarın mevcut olması durumunda alıcıda şifresi çözülebilmektedir.

Kriptografi sadece kanal üzerinden avantaj sağlamaktadır. Alıcıda bir anahtar kullanılarak şifrelenmiş içeriğin şifresi çözüldükten sonra, dijital içeriği telif hakkına karşı korumak için herhangi bir güvenlik aracı mevcut değildir. Bu nedenle, şifreleme, şifre çözme işleminden sonra dijital içeriği koruyan bir yöntemle değiştirilmelidir. Burada filigran kavramı ortaya çıkmaktadır.

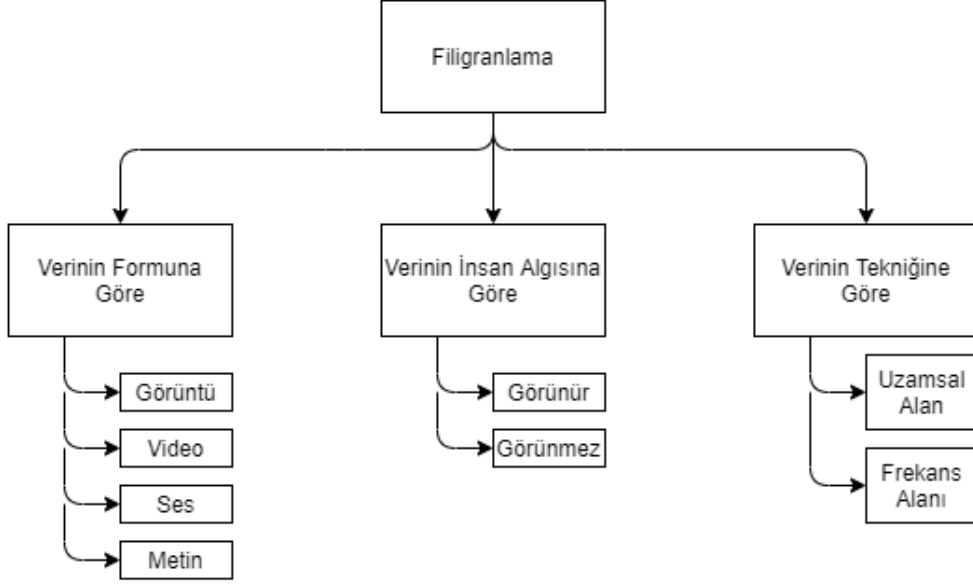
Kriptografi ve filigranlama arasındaki temelinde fark bulunmaktadır. Kriptografi, verileri şifresi çözülmeye kadar okunamayacak şekilde eşlerken, filigranlamada içerikleri orijinal biçiminde tutan veriler yerleştirilmektedir. En önemli fark Kriptografide şifrelenen veriler karşı taraf için anlam ifade etmezken filigranlamada içerikler karşı tarafa açık görünür halde ulaşır. Bu da filigranlama tekniğini, özellikle medya iletiminde ideal bir veri güvenirliliği yöntemi olarak öne çıkarmaktadır.

Herhangi bir filigranlama tekniğinin geçerli olabilmesi için üç önemli ihtiyacı karşılaması gerekmektedir; algısal görünmezlik, çeşitli görüntü işleme saldırılarına karşı gürbüzlük ve kapasite [38,39]. Algısal görünmezlik ifadesi, orijinal veri ile damgalı veri arasındaki *benzerlik* olarak ifade edilmektedir. Orijinal veriye saklanan veri *damga* olarak adlandırılmaktadır. Damga, orijinal veri içinde fark edilmemelidir. Filigranlı veri görüntü işleme saldırılarına karşı gürbüz olmalıdır. Kapasite ise, orijinal veriye en fazla yüklenebilen veri miktarı olarak tanımlanmaktadır [40,41].

2.2. Filigran Algoritmasının Sınıflandırılması

Filigranlama işleminde, çeşitli filigran modelleri bulunmaktadır. Filigranlama işlemi verinin formuna göre, insan algısına göre ve tekniğine göre temel olarak 3 farklı sınıflandırma ile ifade edilmektedir.

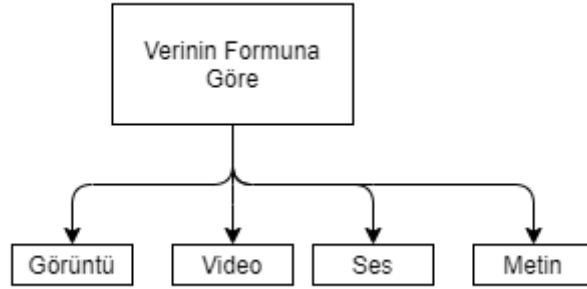
Verinin formuna göre görüntü, video, ses ve metin olmak üzere ele alınmaktadır. Verinin insan algısına göre filigranlama ise; görünür ve görünmez olarak ele alınmaktadır. Verinin tekniğine göre ise; uzamsal alan ve frekans olanı şeklinde ele alınmaktadır. Bu sınıflandırma Şekil 2.2.1'deki gibi gösterilebilmektedir [74].



Şekil 2.2.1 Filigran Algoritmasının Sınıflandırılması

2.2.1. Verinin Formuna Göre Filigranlama

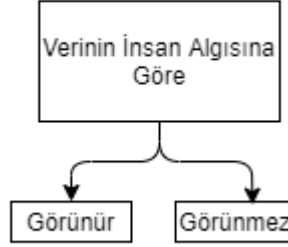
Verinin formuna göre görüntü, video, ses ve metin olmak üzere dört farklı şekilde sınıflandırılmaktadır. Bu sınıflandırma Şekil 2.2.1.1’deki gibi gösterilebilmektedir.



Şekil 2.2.1.1 Filigran Algoritmasının Veri Formuna Göre Sınıflandırılması

2.2.2. Verinin İnsan Algısına Göre Filigranlama

Görünür filigran ve görünmez filigran olmak üzere temelde iki farklı çeşidi bulunmaktadır. Bu sınıflandırma Şekil 2.2.2.1'deki gibi gösterilebilmektedir.



Şekil 2.2.2.1 Filigran Algoritmasının Verinin İnsan Algısına Göre Sınıflandırılması

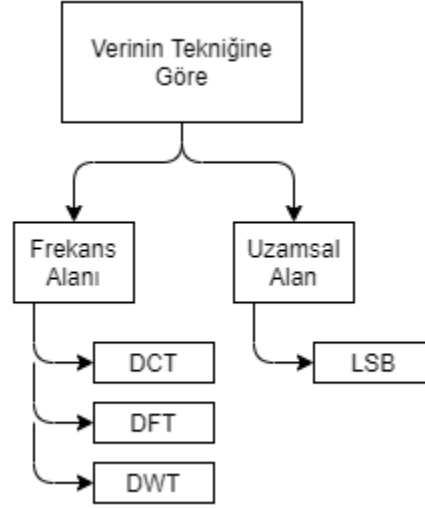
Görünür filigran, sanat beyanında (imza sahipliği) kullanılan tiptir. Görünmez filigran, savaş sırasında gizli mesajlar gibi bilgi alışverişinde kullanılan tiptir.

Görünür filigran, logosu TV köşesinde görünür şekilde bindirilmiş televizyon kanalları gibi dijital olarak filigranlanmaktadır. Görünmez filigranlama ise çok daha karmaşık bir kavramdır. Yazar, distribütör vb. gibi telif hakkı verilerini tanımlamak ve gizli mesajlar gibi bilgi alışverişinde sıklıkla kullanılmaktadır [42].

Resmi sınavlarda, filigranlama işlemi görünür filigranlama olarak kullanılmaktadır. Telif hakları gibi kullanım alanlarında ise görünmez filigranlama kullanılmaktadır.

2.2.3. Verinin Tekniğine Göre Filigranlama

Verinin tekniğine göre temelde Uzamsal ve Frekans Alanı olmak üzere iki şekilde sınıflandırılmaktadır. Bu sınıflandırma Şekil 2.2.3.1'deki gibi gösterilebilmektedir.



Şekil 2.2.3.1 Filigran Algoritmasının Verinin Tekniğine Göre Sınıflandırılması

Uzamsal alan tekniğinde, piksel bilgisi direkt olarak değişmesinden dolayı görüntü ve video kalitesini büyük ölçüde düşürmektedir ve çeşitli saldırılar karşısında zayıf bir tekniktir. Günümüzde veri gizlemede kullanılan en yaygın tekniklerden biri, en az anlamlı bit (LSB) ekleme olarak adlandırılır. LSB ekleme, en önemsiz bite veri ekleme algoritmasının uygulanması işlemine denmektedir. Bu sebeple, verilerin küçültülmesi durumunda veride kayıplar ve bozulmalar olmasından dolayı, kullanımı çok yaygın değildir [43]. Aynı zamanda dışarıdan gelen gürültü saldırılarına karşı dayanıksızdır.

Frekans alanındaki tekniklerde, filigran için daha fazla bit verisi gizlenmektedir. Ayrıca karşı saldırılara, filtrelere ve sıkıştırmaya daha güçlü yanıt vermektedir. Bu alandaki tekniklere ilerleyen bölümlerde detay verilmektedir.

2.3. Sayısal Filigranlamada Kullanılan Teknikler

Görüntü işleme tekniklerinde uzamsal alanda tüm bilgiler elde edilememektedir. Bu sebeple, tüm bilgilerin elde edilebilmesi için frekans alanı kullanılarak işlemler yapılmaktadır. Sayısal filigranlama işlemlerinde frekans alanı olan Kosinüs Dönüşümü (DCT), Fourier dönüşümü (DFT) ve Ayrık Dalgacık dönüşümü (DWT) teknikleri kullanılmaktadır.

DCT tekniğinde dijital veri, farklı genliklere sahip sinüs ve kosinüs frekanslarına bölünmektedir [20]. DCT, temel olarak veri sıkıştırma, örüntü tanıma gibi alanlarda kullanılmaktadır. Filigran verisi, orta frekans katsayısına gömülmektedir ve orta frekans bandında görüntünün algılanabilirliği etkilenmemektedir.

DWT ise, DFT veya DCT işlemlerinden farklı olarak hem zamansal hem de frekans alanındaki bileşenleri ifade etmektedir. Bu sebeple, sinyalde hem zaman hem de frekans bileşeni elde edilmesi için DWT yöntemi tercih edilmektedir.

DWT tekniği 1980'lerin başlarında ortaya atılmıştır [44]. Dalgacık dönüşümü ses ve görüntü sinyalleri gibi birçok dijital veri üzerinde bilgi gömme ve çıkarma işleminde kullanılan matematiksel bir tekniktir. Dalgacık dönüşümü, veri sıkıştırma, örüntü tanıma, görüntü işleme gibi birçok veri iletimi konusunda sıklıkla kullanılmaktadır [45]. Bu teknikle, görüntü filigranlama uygulamalarında, gizli bilgi gürbüz şekilde dijital bir içeriğe gizlenebilmektedir.

DWT işleminde, sinyal hem zaman hem de frekans olarak yerleştirilmiş işlevler cinsinden temsil edilmektedir [46]. Son yıllarda, DWT işlemleri görüntü işlemede, özellikle kodlama uygulamalarında çok popüler hale gelmiştir [47].

DFT ve DWT, görüntü işlemede kullanılan en önemli iki dönüşüm türüdür. DFT ve DWT işlemleri farklı görünse de bazı benzerlikler vardır. DWT hem uzamsal hem de frekans verileri sağlarken, DFT'nin de benzer verileri sağladığı gösterilmektedir [48]. DFT ve DWT arasındaki temel fark, yüksek geçiş bantlarında oluşan farklılıktır. Yüksek geçişli DFT bantları, daha yüksek frekans çözünürlüğü sağlar, ancak daha düşük uzamsal çözünürlük sağlar. Sonuç olarak, daha fazla frekans bandı vardır, ancak uzamsal verileri tanımak zordur. Öte yandan, DWT alt bantları daha yüksek uzamsal çözünürlük ve daha düşük frekans çözünürlüğü sağlar. Sonuç olarak, alt bant sayısı azdır, ancak uzamsal çözünürlük üstündür.

2.4. Sayısal Filigranlamamın Tercih Edilme Nedenleri

Dijital içeriklerin güvenliğine dayanarak filigranlama uygulamaları yeni teknoloji ile birlikte hızlı bir şekilde hayatımıza giriş yapmıştır. Kullanılan her filigranlama tekniğinin kendi özelliklerine göre tasarımları mevcuttur. Filigranlamada bazı temel gereksinimler bulunmaktadır, bunlar aşağıdaki gibidir:

- **Gürbüzlük/Sağlamlık:** Algoritma işlemlerinde farklı güvenlik saldırılarına karşın sağlam bir algoritmaya sahip olması gerekmektedir. Orijinal veriye gömülmüş olan filigran, kolaylıkla bulunamamalıdır.
- **Kapasite/Boyut:** Filigranlamada boyut, saldırılar üzerine karşı olan filigranın ortadan kaldırılmasıdır.
- **Görünmez Olma:** Filigranlamada görünmez olmanın anlamı, orijinal veriye eklenen filigranın göz ile algılanamaz olmasıdır. Filigranlı veri, orijinal verinin aslımı bozmayacak şekilde algoritma tarafından içeriğe yerleştirilmektedir.

2.5. Sayısal Filigranlamamın Kullanım Alanları

- Sayısal filigranlama, telif haklarının korunumunda önemli rol oynamaktadır. Bu teknikle dijital veri içine aslında veri sahibine ait olan özel bir filigran gömülmekte ve bu sayede, veri üzerine işlenen filigran ile aidiyet bilgisi korunmuş olmaktadır.
- Sayısal filigranlama, kopyalama işlemlerinde önemli rol oynamaktadır. Verilerin çoğaltılması gibi basit bir işlemde, istenmeyen kullanımlara karşı korunmayı sağlamaktadır. Çoğaltmanın kullanıldığı araçlar bu filigranı tespit ederek, verinin çoğaltılması işlemine engel olabilmektedir. Sinemalarda ve Blu-Ray disklerde filmlerin sesi filigranlıdır. Kod çözücü bir Blu-Ray disk oynatıcıya yerleştirilmiştir. Bir içeriği oynatırken bir filigran algılanırsa, bu içeriğin gerçekten de korsan bir kopya olduğu sonucuna varılır ve oynatım durdurulur.
- Yayıncılık sektöründe, yayın haklarının kontrol ve izinin sürülmesi maksadı ile çoğunlukla kullanılmaktadır.
- Tıp sektöründe, teknolojik gelişmeler sonucuyla hastalara ait tedavi ve tanı süreçlerinde, hastaya ait bilgilere filigran gömme işlemi uygulanmaktadır.

- Adli işlemlerde/davalarda, elde edilen veri/kanıtların değiştirilmesine karşı sıklıkla kullanılmaktadır.
- Televizyon yayınlarında izlenme oranlarını ölçmek amacıyla kullanılabilir. TV yayın kanallarının içeriklerine yerleştirilen filigranlarla ölçümler yapılabilir. Bir filigranın saptanması üzerine, veriler analiz edilir ve yayın ayrıntıları hızla doğrulanır ve raporlanır.

3. FİLİGRAN ALGORİTMASININ FPGA UYGULAMASI

Modern iletişim ağlarının hızla gelişmesiyle birlikte, günümüzde bilgi daha önce görülmemiş bir hızla iletilmektedir. Aynı zamanda, dijital ortamda yasa dışı olarak üzerinde oynamalar yapılmış şekilde veriler dağıtılabilmektedir. Telif hakkı koruması dünya çapında önemli bir sorun haline gelmiştir. Sayısal filigran, bu sorunun üstesinden gelmek için geliştirilen bir tekniktir. Sayısal filigran ses [49], [50], görüntü [51], [52] ve video [53], [54]'ya uygulanabilse de bu tez çalışması görüntü filigranlama işlemine odaklanmaktadır. Ayrıca bu çalışma sonuçları video görüntülerine de uygulanabilir.

İyi bir görüntü filigranlama yöntemi algılanamaz, gürbüz ve güvenli olmalıdır. Algılanamazlık, filigranların algısal olarak göze çarpmayan olması gerektiği anlamına gelmektedir. Gürbüzlük, farklı türde saldırılara maruz kaldıktan sonra filigranları doğru bir şekilde çıkarma yeteneğini göstermektedir. Sinyal işleme saldırıları (sıkıştırma, filtreleme ve gürültü ekleme) ve geometrik saldırılar (ölçekleme, döndürme, kesme, kırpma ve rastgele bükme) olmak üzere iki tür saldırı bulunmaktadır [55]. Güvenli terimi ise gizli anahtarı bilmeden yetkisiz filigran kod çözmeye karşı direnci ifade etmektedir.

Son on yılda, çeşitli görüntü filigranlama yöntemleri geliştirilmiştir. Filigran uygulamalarına ait hem yazılım hem de donanım ortamında çeşitli çalışmalar mevcuttur [56]. Yazılım ile filigranlama bilgisayar ortamında kolay bir şekilde uygulanabilir [57,58]. Ancak bu yaklaşım yavaş kalmaktadır ve gerçek zamanlı uygulamalar için uygun değildir [59].

Donanım üzerinde gerçekleştirilen filigran uygulamalarında çoğunlukla, özel tasarlanmış olan alan programlanabilir kapı dizileri olan FPGA'lar üzerinde gerçekleştirilmektedir. Donanım uygulamalarının yazılım uygulamalarına göre avantajları çoktur. Hızlı işlem performansı, gürbüzlük, düşük güç tüketimi ve maliyet gibi kıstaslar en önemli avantajlarıdır. Hızlı ve düşük maliyetli olması sebebiyle çoğunlukla elektronik cihazların içerisine bir modül olarak kolaylıkla entegre edilebilmektedir.

Bu avantajlar ele alındığında, gerçek zamanlı uygulamaların filigranlama işleminin donanım üzerinde yapılması teknolojide önemli rol oynamaktadır. Filigran gömme işlemi için donanım veya yazılım platformunda uygulanabilir tasarımlar mevcuttur. Donanım filigranlaması daha iyi bir yaklaşımdır çünkü filigranın gerçek zamanlı olarak gömülmesine yardımcı olur ki bu yazılım filigranlamasından daha güvenli kabul edilmektedir [60].

Filigranlama algoritmaları, bir yazılım uygulamasında, algoritmanın işlemleri bir mikroişlemci üzerinde çalışan bir kod olarak gerçekleştirilmektedir [61]. Bu tür bir uygulamaların, ana dezavantajı, sistem hızını ve donanım performanslarını iyileştirmek için sınırlı araçlar olmasıdır [62]. Bu tür bir uygulamanın donanım üzerinde gerçekleşmesi ile algoritmanın işlemleri tamamen özel olarak tasarlanmış bir devrede uygulanmasıdır. Bu, donanım alanı ve tüketimin azalması ve esas olarak hızın artması gibi avantajları beraberinde sağlamaktadır [61-63].

Filigranlamanın donanım üzerinde uygulanması, düşük güç tüketimi, daha az alan kullanımı ve güvenilirlik açısından yazılım uygulamasına göre avantaj sunmaktadır [64]. Donanım uygulaması, Uygulamaya Özel Entegre Devreler (ASIC) veya Alan Programlanabilir Kapı Dizileri (FPGA) üzerinde özel olarak tasarlanmış devrelerde gerçekleştirilmektedir. Gerçek zamanlı uygulamalar, kompakt sistemler ve güç tüketimi gibi kriterlerde önemli bir etki ortaya koymaktadır.

Dijital veri işleme konusunda FPGA'lar, son derece hızlı, geliştirilmesi kolay ve yüksek alan kapasitesine sahip cihazlardır [65]. FPGA'lar, yanıt süresinin çok önemli bir rol oynadığı gerçek zamanlı sistemlerde de kullanılır. Standart CPU veya GPU'lar yanıt süresi tahmin edilemez sistemlerdir. Yanıt süresini belirli bir aralıkta tutmak için gerçek zamanlı işletim sistemleri kullanılır. Hızlı yanıt süresinin (milisaniye veya daha az) gerekli olduğu işlemlerde yine de yetersiz kalmaktadır.

Bu sebeple, istenen algoritmanın, her zaman aynı yanıt süresini sağlaması için donanımda uygulanması daha uygun olacaktır. Ayrıca bir donanım seçeneği olan FPGA'lar, donanım konfigürasyonunun değişebileceği ve bu değişikliklere göre ayarlanabilen bir devre sistemine sahiptir. Bunun sonucunda istenilen işlemler herhangi bir zamanda istenildiği gibi değiştirilebilir ve aynı donanım yapısı üzerinde tekrar kullanılabilir.

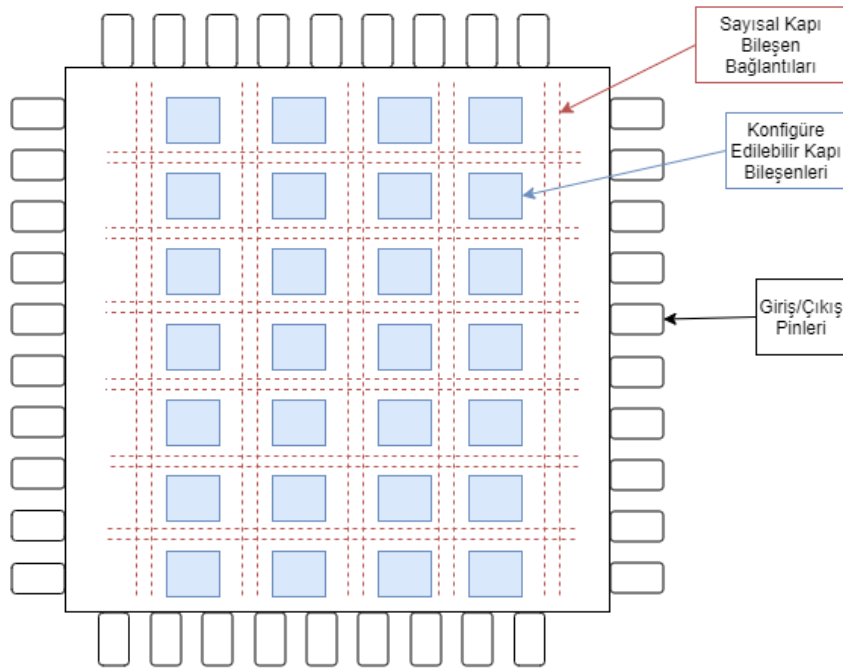
DWT, görüntü dönüşümleri, görüntü filtreleme, veri tanımlama vb. alanlarda yaygın olarak kullanılmaktadır. Tez çalışmasında, donanıma ait özel fonksiyonlar (FPGA üzerinde paralel hesaplama) ile filigranlama işleminin uygulaması gösterilmektedir. Bu alanda yapılabilecek çeşitli çalışmalarda, FPGA tabanlı sistemlerin kullanılabilmesi gösterilmektedir. FPGA tabanlı mimariler ile donanım tasarlanarak, yazılım gibi GPU ve CPU işlemcilerinin hız faktörüne bağımlı kalmaksızın çözümler geliştirilebilmektedir. Yazılım tabanlı çözümler yerine, donanım tabanlı çözümler ortaya konulması ile hız faktörü en önemli oranda kendini göstermektedir.

Donanım geliştirme süreçleri, ortaya çıkan devrenin işlevsel olması ve tasarımcıların ihtiyaçlarını karşılamasına yönelik mimari planlanması, tasarlanması ve yürütülmesini içermektedir. FPGA programlamanın nihai amacı, güç verildiğinde belirli çıktılar üreten bir donanım devre tasarımı oluşturmaktır. Tüm FPGA'lar, programlama sırasında donanım tanımlama dillerini (HDL) kullanmaktadır.

FPGA programlama, yüksek seviye bir yazılımı andırmaktadır. Aradaki fark, yazılım programlamada kullanılan komutların ardışık olarak çalışacak şekilde bir bilgisayar ortamında işlemesi, FPGA programlamanın ise elektriksel programlama aracılığı ile donanım içeriği oluşturmasıdır. FPGA programlamada kullanılan donanım tanımlama dilleri donanımın paralel olarak gerçekleştirilmesine yönelik ifadeler de içermektedir.

Çok Yüksek Hızlı Tümeleşik Devre Donanım Tanımlama Dili (VHDL), bir devrenin çıkışını belirleyen özel tasarımlarla elektronik otomasyonu yapılandırmak için kullanabileceğiniz bir donanım programlama dillerinden biridir. VHDL, bir FPGA'nın donanımını tanımlar ve kullanıcıların gerçek donanım oluşturmadan modellemesine ve simüle etmesine olanak tanımaktadır.

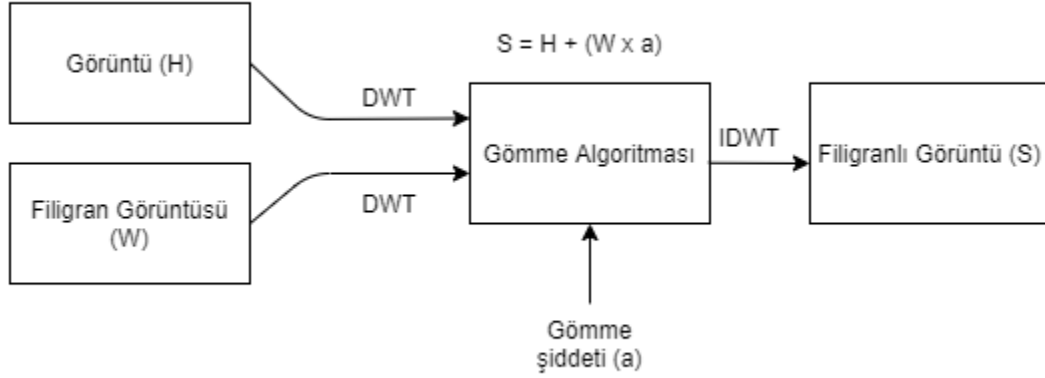
Verilog Dili, elektronik sistemleri tanımlarken elektronik donanımın modellenmesinde kullanabileceğiniz başka bir yaygın HDL'dir. Verilog daha az kod satırı gerektirir ve genellikle C yazılım kodlama diline benzetmektedir. FPGA'ların genel yapısı Şekil 3.1'de olduğu gibi gösterilmektedir.



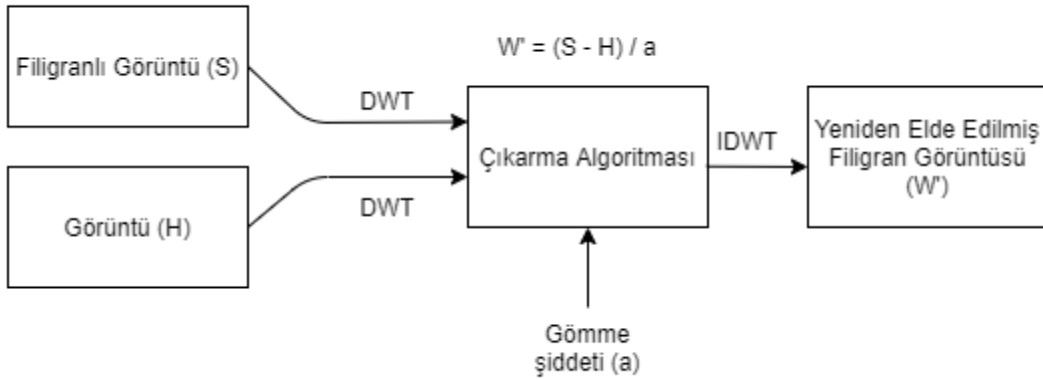
Şekil 3.1 FPGA'nın İç Yapısı

FPGA'ların için, programlanabilir mantık blokları (konfigüre edilebilir kapı bileşenleri) ve bu blokların arasında bulunan ara bağlantılarından (sayısal kapı bileşen bağlantıları) oluşmaktadır. Kullanıcıların ihtiyacı doğrultusunda mantık blokları düzenlenebilmektedir. FPGA'nın anlamı olan alanda programlanabilir terimi, bu mantık bloklarının ve ara bağlantı bileşenlerinin yeniden programlanabilir olmasıdır. Bu mantık bloklarını kullanan giriş ve çıkış bağlantıları mevcuttur.

Düşük maliyet ve tasarım aşamasında kullanıcıya esneklik getirmesi sebebiyle teknolojiye giderek yaygınlaşmaktadır. FPGA içerisinde yapılacak tasarım, şematik tasarım yani devre elemanlarının elle yerleştirilerek aralarındaki bağlantıların şematik ile yapılması veya donanım tanımlama dilleri ile uygulanabilmektedir. Bu tez kapsamında iki programlama dili de kullanılmıştır. Tezde gerçekleştirilen çalışma Şekil 3.2 ve Şekil 3.3’de görüldüğü gibi *Gömme* ve *Çıkarma* işlemi olarak iki aşamada gerçekleştirilmiştir.

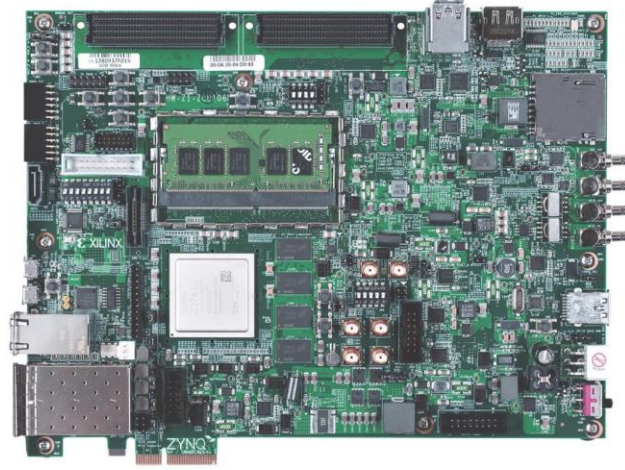


Şekil 3.2 Gömme İşleminin Temel Uygulanışı



Şekil 3.3 Çıkarma İşleminin Temel Uygulanışı

Sentezlenen kod, Xilinx firmasına ait Zynq Ultrascale+ serisi FPGA üzerinde gerçekleştirilmiştir. ZCU106 geliştirme kartı üzerinde tasarım doğrulanmıştır.



Şekil 3.4 Zynq Ultrascale+ serisi ZCU106 geliştirme kartı

Sentezlenen donanım koduna ait güç tüketimi ve sentez sonuçları Şekil 3.5’de gösterilmektedir. Donanım koduna ait kullanılan çip üzerinde oluşacak güç değeri gösterilmektedir. Bu tasarımın herhangi bir genel sistem mimarisinde çalışması durumunda, FPGA kaynaklı bu donanımın ne kadar güç çekeceği görülmektedir.

Power	
Total On-Chip Power:	0.768 W
Junction Temperature:	25.7 °C
Thermal Margin:	74.3 °C (75.1 W)
Effective θ_{JA} :	1.0 °C/W
Power supplied to off-chip devices:	0 W

Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAMs	IO	Start	Elapsed
synth_design Complete!								582	556	0.00	4	2/15/22, 9:40 PM	00:00:56
write_bitstream Complete!	4.577	0.000	0.010	0.000	0.000	0.768	0	1214	713	130.00	6	2/15/22, 9:42 PM	00:04:55

Şekil 3.5 Sentez Sonuçları

Her FPGA, belirli sayıda programlanabilir mantık, ara bağlantı, giriş çıkış ve hafıza kaynaklarına sahiptir. Derleyici uygulama, FPGA’da kod tasarımında bu kaynakları kullanır. Bir FPGA’daki programlanabilir kaynaklar, kaynaklar arasında ara bağlantılara izin verir ve çip boyunca düzenli aralıklarla bulunur. Kaynaklar arasındaki ara bağlantılar, FPGA kodunun gerektirdiği bağlantı yol süresini azaltır.

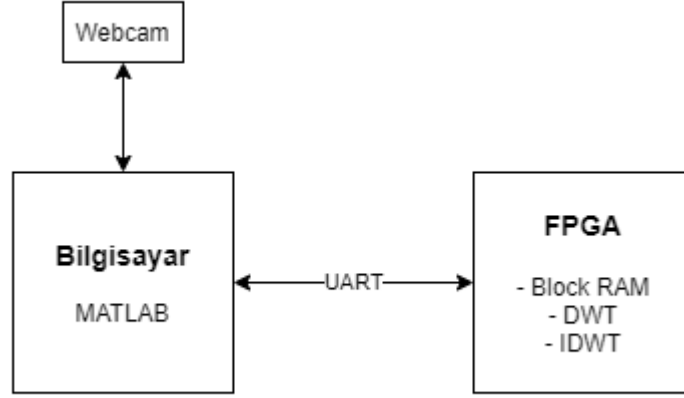
Bir FPGA'da her kaynağın birden çok örneği bulunduğundan, kaynak çakışmalarını en aza indirirken birden çok eşzamanlı işlem aynı cihaz üzerinde aynı anda çalışabilir. Tablo 1'de uygulamaya ait donanım kaynak kullanım sonuçları gösterilmiştir. Bu tablo ile donanım uygulamasının sayısal devreler üzerinde optimal çalıştığı ve bu değerlerin donanıma bağlı olarak daha da optimal olabileceği gösterilmektedir. Bu tablo, bir FPGA'daki programlanabilir kaynakların basitleştirilmiş bir örneği gösterilmektedir. Yalnızca ayarlanan sayıda programlanabilir ve bellek kaynağı sağlandığında, kullanılan LUT sayısını kontrol etmek önemlidir. Matematiksel hesaplama, bize belirli bir HDL kodu için olası LUT sayısının kaba bir tahminini verebilir.

Tablo 1 Kaynak Kullanım Sonuçları

	Kullanılan Miktar	Kaynak Miktarı	Kullanım Oranı %
<i>LUT</i>	1.214	230.400	0.53
<i>FF</i>	713	460.800	0.15
<i>BRAM</i>	130	312	41.67
<i>IO</i>	6	360	1.67
<i>BUFG</i>	1	544	0.18
<i>MMCM</i>	1	8	12.50

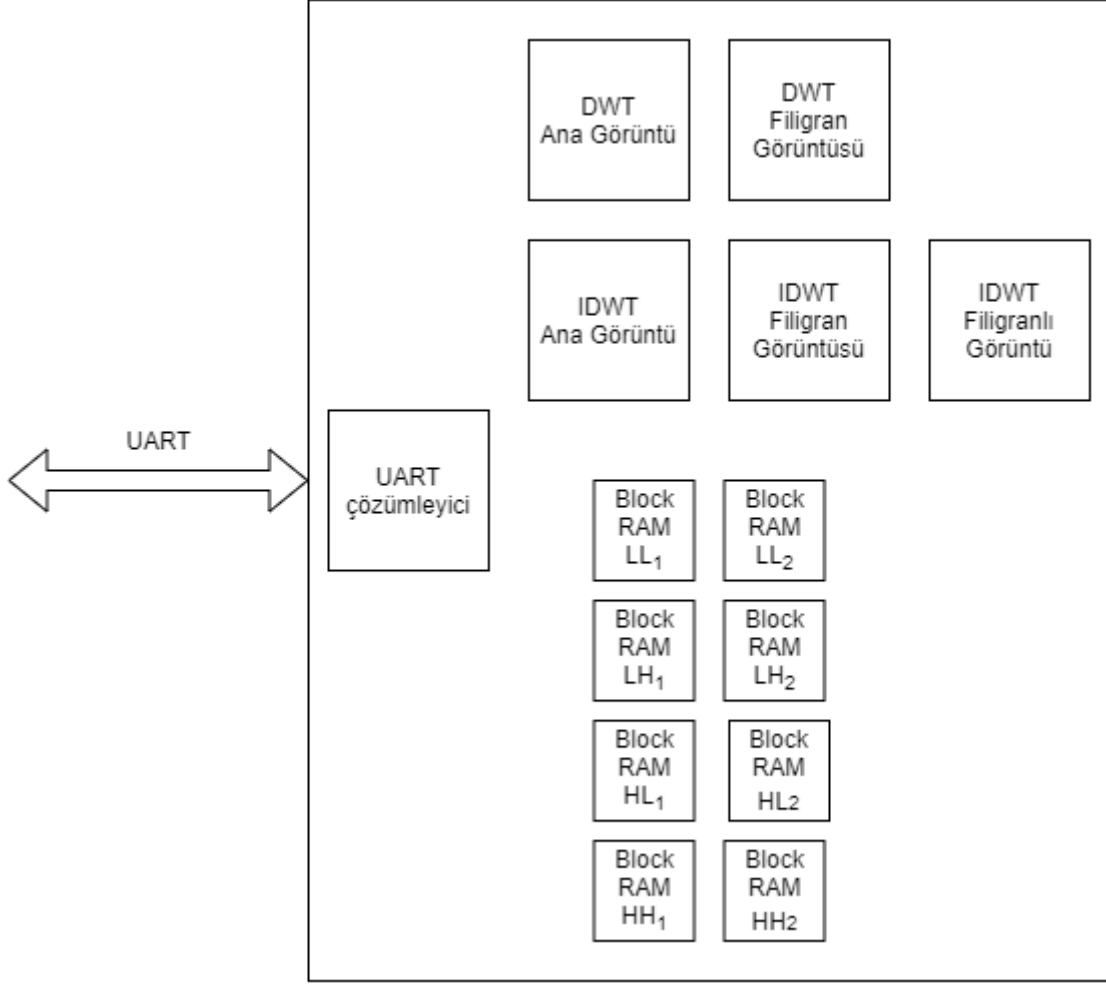
Tablo 1'de sunulan kaynak kullanım sonuçları ele alındığında en fazla kullanım oranı BRAM'dir. Bu kullanım oranının fazla olmasının sebebi ise yüksek performans ve düşük güç tüketimini optimize edebilmek için FPGA'da paralel işlem tasarımıdır. Bu optimizasyon işlemi ilerleyen bölümlerde detaylandırılmaktadır. Saat üretici olarak daha yüksek hızda işlem yapılması için FPGA içerisinde bulunan saat üretici dönüşümü kullanılmıştır ve bu dönüşüm MMCM oranı olarak gösterilmektedir.

Şekil 3.6'da tasarıma ait temel görsel verilmiştir. Alt bölümlerde ise tasarımda kullanılan tüm modüllerin detaylı uygulaması anlatılmaktadır.



Şekil 3.6 Tasarımın Genel Yapısı

FPGA donanım tasarımına ait şematik görsel Şekil 3.7’de verilmektedir. Bu görselde donanım tasarımına ait modüllerin birbiri ile olan bağlantısı görülmektedir. Bu bağlantılar UART modülü, DWT işlem modülleri, IDWT işlem modülleri ve Block RAM modüllerinden oluşmaktadır. Donanım üzerinde paralel işlem olanağı sağlamak amacı ile Block RAM ve DWT işlemlerine ait modüllerden birden çok kullanılmıştır.



Şekil 3.7 FPGA İç Tasarım Detayı

3.1. MATLAB Modülü

Bu modülde, seri haberleşme protokollerinden en çok kullanılan UART arayüzünün kullanımı gerçekleştirilmiştir. MATLAB içerisinde bulunan 'serial' komutu ile öncelikli olarak seri haberleşme yapılacak olan port bilgisi seçilmektedir.

MATLAB'da bulunan 'fwrite' ve 'fread' komutları ile verinin gönderilmesi ve alınması sağlanmaktadır. 300x300 boyutlarında MATLAB tarafından oluşturulmuş bir ana görüntü (Lena) ile Webcam videosundan alınan gerçek zamanlı filigran görüntüsü 'fwrite' komutu ile UART üzerinden seri olarak FPGA'ya aktarılmaktadır. Bu farklı bir şekilde gerçek zamanlı bir video ile sabit bir filigran görüntüsü şeklinde de kurgulanabilir.

Filigran verisi için kolaylık olması açısından Webcam görüntüsüne alternatif olarak bir test videosu üretilmiştir. Bu test videosu 0'dan 100'e kadar sayan bir sayacın videosudur. Çalışma sırasında bu video gerçek zamanlı video verisi olarak kullanılmıştır. Anlık olarak bu video dosyasından elde edilen görüntüler filigran verisi olarak 300x300 boyutuna getirilerek yeniden şekillendirilmektedir. Bu filigran verisi, girdi olarak alınan görüntü ile birlikte seri hat üzerinden FPGA'ya aktarılmaktadır. Ardından veriler FPGA içerisinde bulunan Block RAM ve dönüşüm modüllerinde işlenmektedir.

3.2. Dalgacık Dönüşüm Modülü – DWT

Bu modülde, filigranın orijinal görüntüye gömülmesi işlemi bulunmaktadır.

$$LL_S = LL_H + a \cdot LL_W \quad (3.1)$$

Bu denklemde LL_H orijinal görüntünün 1.seviye LL bandı, LL_W filigran görüntüsünün 1.seviye LL bandı, LL_S ise filigranlanmış görüntüye ait olan LL bandıdır. a değeri ise filigranlama etkisini sağlamaktadır. Bu değer yüksek olduğunda görüntüde bozulmalar meydana gelmekte, düşük olduğunda ise filigranlı görüntüden filigranın çıkartılması zorlaşmaktadır. Bu değer testler sonrasında 0,03 olarak belirlenmiştir. Gömme şiddeti değeri olan a nın 0,03, 0,003 ve 0,0003'e düşürdükçe, algısal görünmezlik kapsamında elde edilen sonuçların iyiye gittiği görülmektedir.

DWT, dijital görüntü işleme, dijital video işleme ve sinyal işleme için en yaygın kullanılan frekans dönüşümlerinden biridir. Bu modül bir uzamsal alan dijital verisini frekans alanına dönüştürmektedir. DWT kullanılan yöntemlerin en önemli avantajı, diğer frekans alanı dönüşümlerine göre insan görme sistemine göre en iyi seçenek olmasıdır. DWT, görüntü analizi ve işleme için güçlü bir tekniktir ve görüntü sıkıştırma, gürültü giderme, özellik çıkarma ve örüntü tanıma gibi alanlarda yaygın uygulamaları bulunmaktadır.

DWT işleminin temel fikri, bir görüntüyü farklı uzamsal alan ve bağımsız frekans bölgelerinin bir alt görüntüsüne ayrıştırmaktır. DWT, görüntü ve ses gibi sinyalleri dalgacık katsayıları açısından analiz etmek ve temsil etmek için kullanılan matematiksel bir araçtır. Bir sinyali, sinyaldeki yerel varyasyonları yakalayan küçük dalga benzeri fonksiyonlar olan bir dizi dalgacıklara ayrıştıran bir dönüşüm türüdür. DWT, bir sinyale, sinyali farklı frekans bantlarına ayıran bir dizi filtre uygulayarak çalışmaktadır.

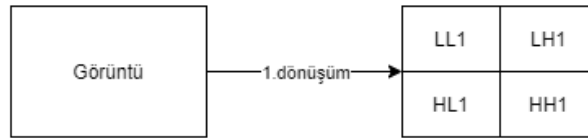
DWT ile görüntü önce farklı çözünürlük seviyelerinde alt bantlara ayrıştırılır ve her bir alt bant görüntünün farklı frekans bileşenleri hakkında bilgi içermektedir. Ayrıştırma işlemi, görüntüyü bir dizi düşük frekanslı ve yüksek frekanslı alt bantlara ayıran bir dizi uygulanmayı içermektedir. Düşük frekanslı alt bant, kaba görüntü ayrıntılarını içerirken, yüksek frekanslı alt bantlar, ince görüntü ayrıntılarını içermektedir.

Ayrıştırma işlemi, istenen ayrışma düzeyine ulaşılan kadar her bir alt bantta yinelemeli olarak tekrarlanmaktadır. Her seviyede, sinyal iki alt banda ayrılır: sinyalin düşük frekans bileşenlerini içeren yaklaşım alt bantta temsil edilirken, yüksek frekans bileşenlerini içeren detay ayrıntı alt bantta temsil edilmektedir. Ayrıntı alt bandı ayrıca iki alt banda bölünür ve işlem, istenen ayrıştırma seviyesine ulaşılan kadar tekrarlanmaktadır. İstenildiğinde dönüşüm seviyesi artırılarak daha fazla bant elde edilebilmektedir. DWT işlemi, hızlı olması ve bir görüntüyü hiyerarşik alt bantlara ayrıştırması sebebiyle verimli bir araçtır. Dalgacık dönüşümünün kullanımı, esas olarak bilgi gizleme sistemi özelliklerinin kapasitesini ve gürbüzlüğü ele almaktadır.

Haar dalgacığı, Ayrık Dalgacık Dönüşümünde (DWT) kullanılan bir tür dalgacık fonksiyonudur. Adını ilk kez 1909'da öneren Macar matematikçi Alfréd Haar'dan almıştır [10]. Haar dalgacığı, görüntü ve sinyal işleme uygulamalarında yaygın olarak kullanılan, basit ve hesaplama açısından verimli bir dalgacık işlevidir. Burada, iki piksel değerinin ortalaması alınarak düşük frekans dalgacık katsayıları ve aynı iki pikselin farkının yarısı alınarak oluşturulan yüksek frekans katsayıları elde edilmektedir

Haar Dalgacık Dönüşümü, sinyalleri dalgacık katsayıları açısından analiz etmek ve temsil etmek için güçlü bir araçtır. Ortaya çıkan dalgacık katsayıları, sinyali kompakt ve verimli bir şekilde temsil etmek için kullanılabilir. Her seviyedeki katsayılar, sinyali farklı bir ölçekte ve konumda temsil eder ve en yüksek seviye, sinyalin en ince ayrıntılarını temsil etmektedir. Bu, sinyalin kalitesini önemli ölçüde etkilemeden daha düşük genliklere sahip katsayıların atılabildiği sıkıştırma gibi işlemlerde yararlı olabilmektedir. Çeşitli sinyal işleme görevleri için kullanılabilir ve görüntü ve ses sıkıştırma, gürültü giderme ve özellik çıkarmada yaygın olarak kullanılmaktadır.

Dalgacık dönüşümünün temel avantajı, bağımsız olarak işlenebilen dört alt banda (LL, LH, HL ve HH) ayrılan frekanslardır [74]. Şekil 3.8'de dalgacık dönüşümü kullanılarak görüntünün satırları ve sütunları kapsamında oluşan birinci seviye ayrıştırma göstermektedir. Ayrıştırma sonucunda dört alt bant oluşmaktadır. Görüntünün düşük frekanslı kısmı LL_1 , yatay yönde kenar özelliğine sahip yüksek frekanslı kısmı LH_1 , dikey yönde kenar özelliğine sahip yüksek frekanslı kısmı HL_1 ve köşegen yönde kenar özelliğine sahip yüksek frekanslı kısmı HH_1 temsil etmektedir [66].

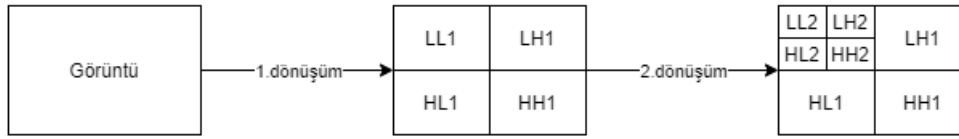


Şekil 3.8 1.Seviye Dönüşüm

DWT işlemi, görüntüyü hiyerarşik olarak ayrıştırmak için matematiksel bir araç olarak kullanılmaktadır. Orijinal görüntüyü daha düşük yakınlaştırma görüntüsü (LL), yatay (LH), dikey (HL) ve diyagonal (HH) olarak isimlendirilen dört alt banda ayırmaktadır [67].

Orijinal görüntü, daha düşük çözünürlüklü bir yaklaşım bölgesine, yatay, dikey ve çapraz frekans bantlarına ayrılır. Düşük frekans bandı, DWT'nin bir sonraki seviyesinde kullanılmaktadır. Yani ayrıştırma aynı şekilde daha düşük çözünürlüklü yaklaşım bölgesi ile devam eder. DWT'de, en önemli bilgi yüksek genliğe sahiptir ve daha az önemli olan bilgi düşük genliğe sahiptir.

Dalgacık dönüşümü kullanarak ikinci dönüştürme seviyesinden sonra, Şekil 3.9'daki gibi görüntü boyutunun 1/8'i büyüklüğünde dört alt bant elde edilmektedir. LL₂ bandı: düşük frekanslar, ayrıntılar ise; yatayda LH₂, dikeyde HL₂ ve köşegende HH₂ olarak gerçekleşmektedir [67].



Şekil 3.9 2.Seviye Dönüşüm

Bu tez kapsamında kullanılan yöntemde; filigran, orta frekansları içeren LH₂ alt bandına yerleştirilmektedir. Bu genelde yatay ve dikey bileşenlere karşılık gelmektedir. Bu aşamanın sonunda, filigranlı görüntüyü oluşturmak için IDWT yani ters DWT işlemi uygulanmaktadır.

Filigranlama uygulamalarında, daha düşük ayrıştırma seviyeleri, daha yüksek ayrıştırma seviyelerine kıyasla daha düşük enerji oranına sahip oldukları için görüntü değişikliğine karşı daha savunmasızdır. Bu enerji denklemi aşağıdaki gibi tanımlanır.

$$E_k = \frac{1}{N_k M_k} \sum_i \sum_j |I_k(i, j)| \quad (3.2)$$

k'nin ayrıştırma seviyesi olduğu yerde, I_k karşılık gelen alt bandın katsayılarını belirtir ve N_k ve M_k alt bant boyutlarıdır. Aynı seviyedeki alt bantların enerjileri karşılaştırıldığında, yatay detayda (HL_k) enerji birikiminin önemli ölçüde daha fazla olduğu görülebilmektedir. Dikey ve diyagonal detaylar, dolayısıyla bu alt bandın görüntü modifikasyonuna karşı daha gürbüz olduğunu göstermektedir. Yaklaşık görüntü (LL_k) orijinal görüntünün enerjisinin en yüksek kısmına sahip olsa da filigranın bu kısma gömülmesi görüntü kalitesini düşürmektedir. Bu nedenle, aynı anda görüntü kalitesinin korunmasını sağlamak ve gürbüzlük sunmak için, filigranı yerleştirmek için her seviyedeki yatay alt bant yani LH_k en iyi alan olarak seçilebilmektedir.

DWT katsayılarının büyüklükleri, her ayrıştırma seviyesinde en düşük bantlarda (LL) daha büyüktür. Filigranın daha yüksek seviyeli alt bantlara gömülmesi, filigranın sağlamlığını arttırmaktadır. Ancak, görüntünün görsel doğruluğu kaybolabilmektedir.

DWT tabanlı yöntemler, DCT yöntemiyle karşılaştırıldığında daha doğru görselleştirme sonuçları ortaya koymaktadır. Ayrıca, insan görsel sistemine daha az duyarlı alanlar kullandığında, görüntü kalitesini düşürmeden filigranların gömülmesi için daha fazla fırsat sağlayacağından ve bu yöntemin gürültüye karşı en güçlü yöntem olduğundan da literatürde yer almaktadır [73]. Ek olarak, DCT tabanlı dönüşümlerle karşılaştırıldığında, DWT görüntülerin bloklara ayrıştırılmasını gerektirmediğinden daha az görsel artefakt üretir.

DWT, çok çeşitli uygulamalar için kullanılabilen güçlü bir istatistiksel yöntemdir. DWT, çok sayıda uygulamada kullanılmaktadır ve DFT uygulamaları yerine tercih edilmektedir. DWT, görüntü sıkıştırma, özellik çıkarımı, görüntü tanımlama ve tıbbi görüntü teknolojisi alanlarında da kullanılabilir.

Bir $x[n]$ sinyalinin Haar DWT işlemi için denklemler şunlardır:

$x[n]$ bir N-noktası sinyali ise, $x[n]$ 'nin DWT'si şu şekilde verilmektedir:

- İlk olarak, aşağıdaki denklemleri kullanarak $x[n]$ 'nin ortalaması ve farkı hesaplanmaktadır:

$$a_0[k] = \frac{x[2k] + x[2k+1]}{\sqrt{2}} \quad (3.3)$$

$$d_0[k] = \frac{x[2k] - x[2k+1]}{\sqrt{2}} \quad (3.4)$$

burada $k = 0, 1, \dots, N/2-1$ ve N tek ise $N/2$ yukarı yuvarlanmaktadır.

- $a_1[k]$ ve $d_1[k]$ elde etmek için ortalama $a_0[k]$ üzerinde yukarıdaki adım tekrarlanmaktadır:

$$a_1[k] = \frac{a_0[2k] + a_0[2k+1]}{\sqrt{2}} \quad (3.5)$$

$$d_1[k] = \frac{a_0[2k] - a_0[2k+1]}{\sqrt{2}} \quad (3.6)$$

burada $k = 0, 1, \dots, N/4-1$ ve $N/2$ tek ise $N/4$ yukarı yuvarlanmaktadır.

- $a_2[k]$ ve $d_2[k]$ elde etmek için $a_1[k]$ üzerinde yukarıdaki adım tekrarlanmaktadır:

$$a_2[k] = \frac{a_1[2k] + a_1[2k+1]}{\sqrt{2}} \quad (3.7)$$

$$d_2[k] = \frac{a_1[2k] - a_1[2k+1]}{\sqrt{2}} \quad (3.8)$$

burada $k = 0, 1, \dots, N/8-1$ ve $N/4$ tek ise $N/8$ yukarı yuvarlanmaktadır.

İstenilen ayrışma seviyesine ulaşılan kadar bu işleme devam edilmektedir. $a_J[k]$ katsayıları, J seviyesindeki yaklaşıklık katsayılarını temsil ederken, $d_J[k]$ katsayıları, J seviyesindeki detay katsayılarını temsil etmektedir.

Ters DWT ise yukarıdaki verilen işlemi tersine çevirerek hesaplanabilmektedir:

- İlk olarak, J-1 seviyesinde sinyali elde etmek için J seviyesinde, $d_{J-1}[k]$ detay katsayılarının ters dönüşümünü hesaplanmaktadır:

$$a_{J-1}[2k] = \frac{d_{J-1}[k] + a_J[k]}{\sqrt{2}} \quad (3.9)$$

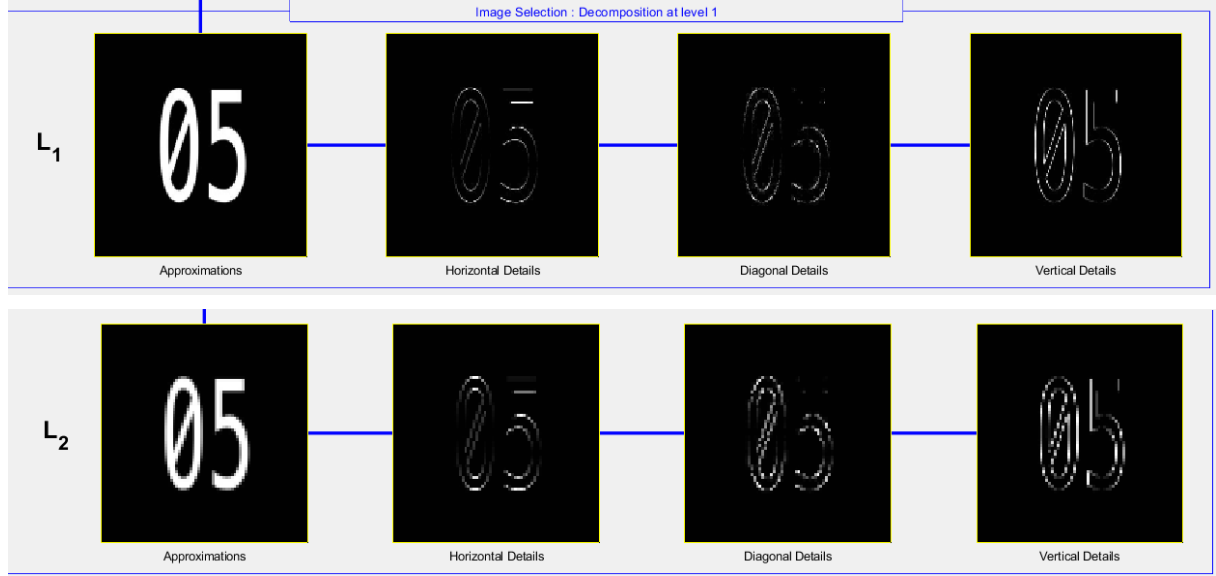
$$a_{J-1}[2k + 1] = \frac{d_{J-1}[k] - a_J[k]}{\sqrt{2}} \quad (3.10)$$

burada $k = 0, 1, \dots, N/2^{(J-1)}-1$.

Orijinal $x[n]$ sinyali elde edilene kadar yukarıdaki adımlar tüm J seviyeleri için tekrarlanmaktadır.

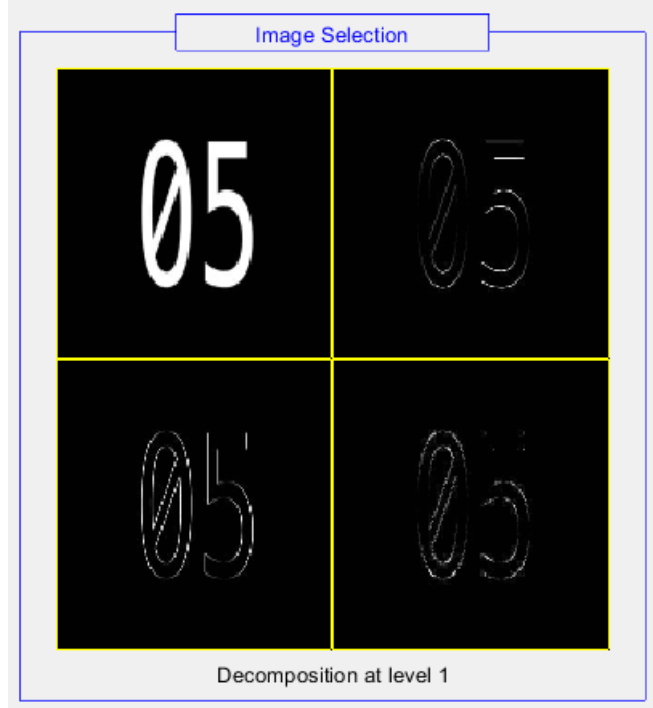
Bu denklemler, Haar dalgacıyı kullanan DWT'nin temel uygulamasını temsil etmektedir. Diğer dalgacık türleri kullanılabilir ve uygulama, verimlilik ve sayısal kararlılık için optimize edilebilmektedir.

Aşağıda Şekil 3.10’da filigran görüntüsüne ait 1.seviye ve 2.seviyeye ait yatay dikey ve köşegen detaylarına ait dönüşüm sonuçları gösterilmektedir.



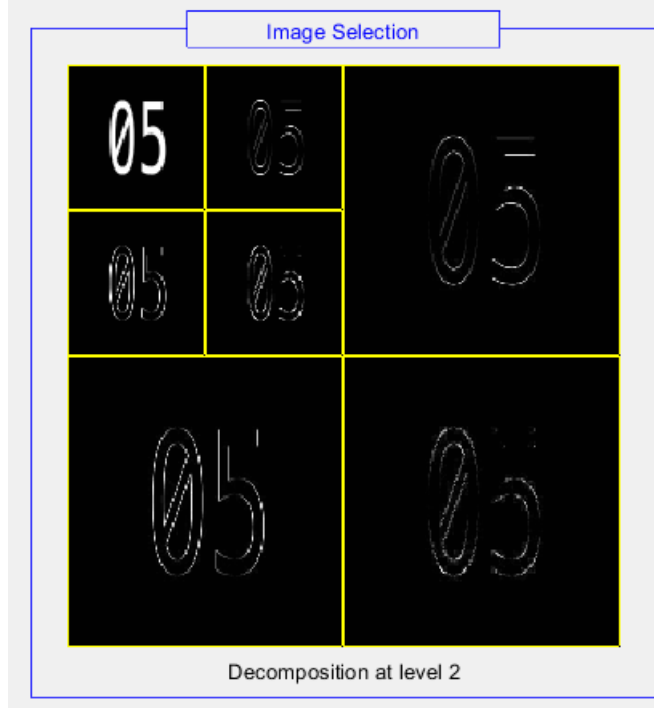
Şekil 3.10 Filigran Görüntüsünün 1. ve 2. seviye Yatay, Dikey ve Köşegene ait Dönüşüm Sonuçları

Şekil 3.11’de gösterilen görüntüye ait 1. seviye dönüşüm alt bantları yani LL, LH, HL ve HH bileşenlerini göstermektedir. Dalgacık dönüşümü, görüntüyü farklı frekanslarda dört alt banda, yani yaklaşık görüntü (LL_k), yatay (HL_k), dikey (LH_k) ve köşegen (HH_k) ayrıntılara ayırır; burada k, ayrıştırma seviyesini göstermektedir.



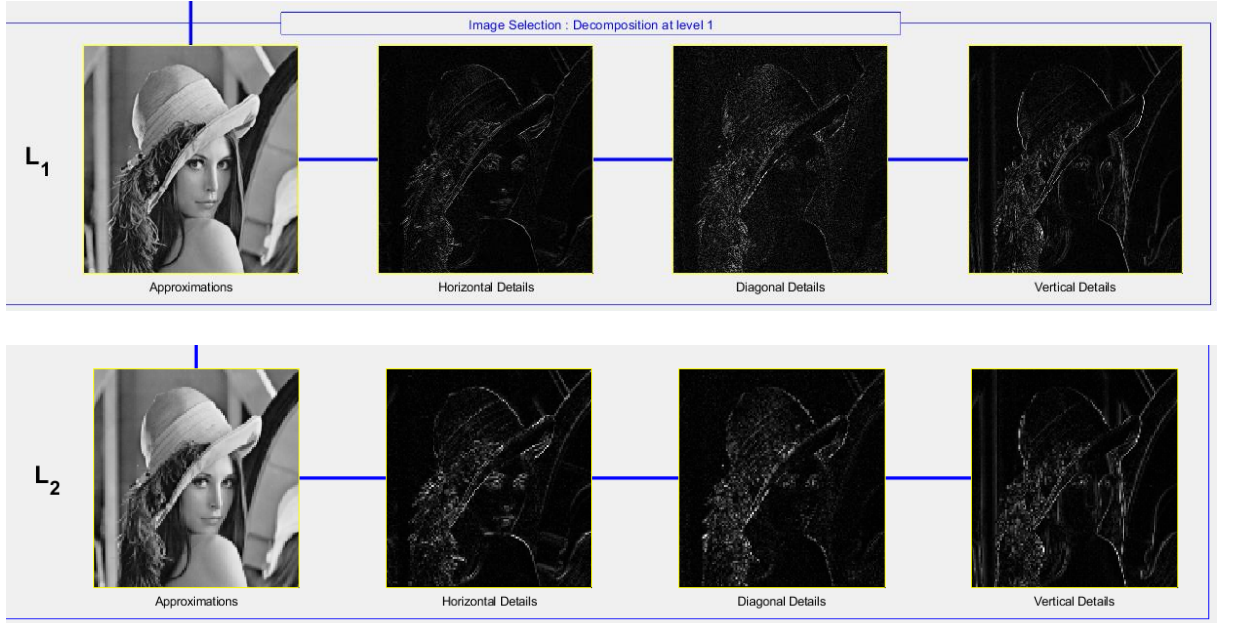
Şekil 3.11 Filigran görüntüsünün 1. seviye Dönüşüm Sonuçları

Düşük çözünürlüklü yaklaşım bölgesi bilgisi DWT ile ayrıştırılırsa, alt düzey frekans bölgeleri elde edilir. Şekil 3.12’de ikinci kez DWT ayrışımından sonra elde edilen sonuç gösterilmektedir. Orijinal görüntü LL_1 , HL_1 , LH_1 , HH_1 frekans bölgelerine ayrıştırılmaktadır. Bir görüntünün daha düşük çözünürlüklü yaklaşım bölgesi, orijinal görüntünün en önemli bilgisini içermektedir. Düşük frekans bölgesi olan LL_1 ; ayrıca LL_2 , HL_2 , LH_2 ve HH_2 'nin alt seviye frekans bölge bilgilerine ayrıştırılabilmektedir. Bunu yaparak, orijinal görüntü k seviyeli dalgacık dönüşümü için ayrıştırılabilmektedir.



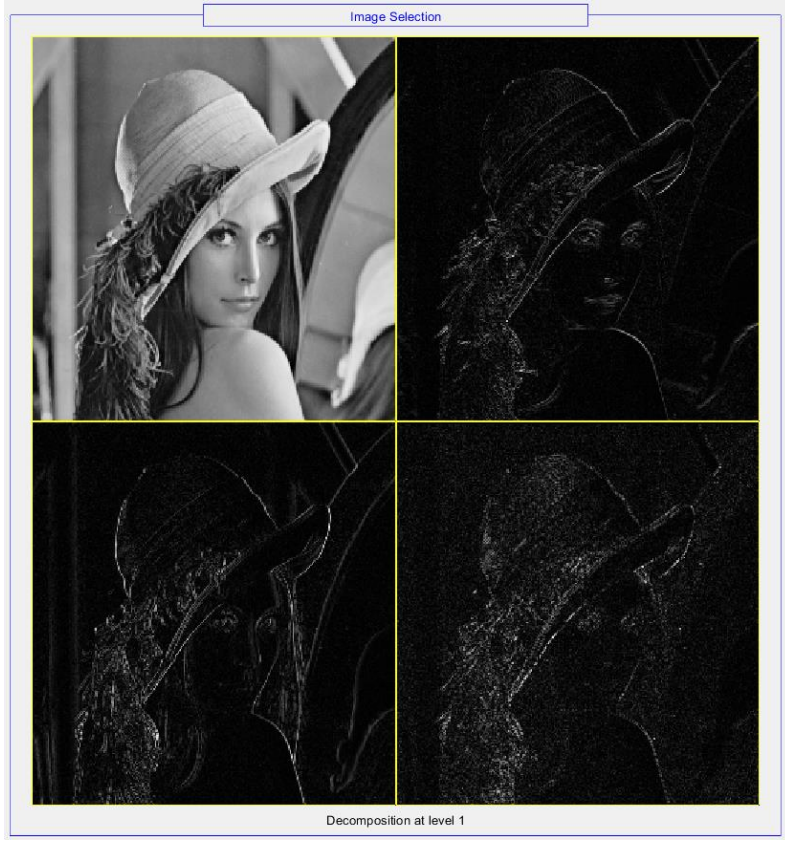
Şekil 3.12 Filigran görüntüsünün 2. seviye Dönüşüm Sonuçları

Şekil 3.13’de ana görüntüye ait 1.seviye ve 2.seviyeye ait yatay dikey ve köşegen detaylarına ait dönüşüm sonuçları gösterilmektedir.

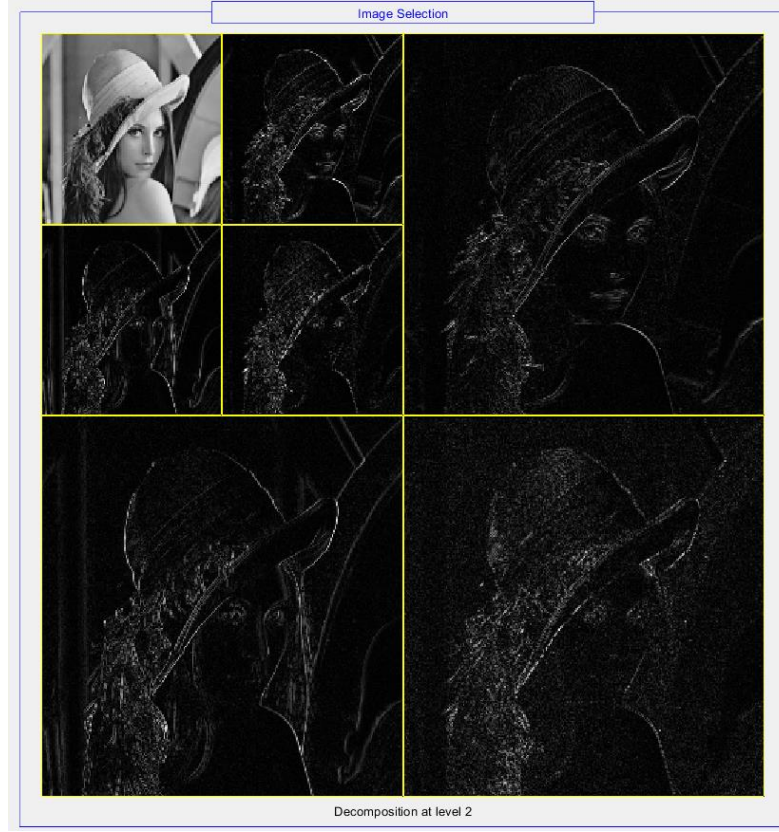


Şekil 3.13 Ana Görüntünün 1. ve 2. seviye Yatay, Dikey ve Köşegene ait Dönüşüm Sonuçları

Elde edilen dört bant, Şekil 3.14’de gösterilen LL, LH, HL ve HH’dir. Şekil 3.15’de iki kez DWT ayrışımından sonra iki boyutlu ana görüntünün sonuçlarını gösterilmektedir.



Şekil 3.14 Ana (Lena) görüntünün 1. seviye Dönüşüm Sonuçları



Şekil 3.15 Ana (Lena) görüntünün 2. seviye Dönüşüm Sonuçları

DWT, yüksek performans ve düşük güç tüketimi sağlayabildiği için FPGA tabanlı sinyal işleme uygulamaları için popüler bir algoritma haline gelmiştir. DWT algoritması, dalgacık ayrıştırma ve yeniden oluşturma işlemlerini gerçekleştirmek için kullanılan toplayıcılar, çarpanlar ve kaydırıcılar gibi donanım modülleri kullanılarak bir FPGA üzerinde uygulanabilmektedir. FPGA, işlem süresini büyük ölçüde hızlandırabilen dalgacık katsayılarının hesaplanmasını paralel hale getirmek için de kullanılabilir. Çoğu yazılım uygulamalarında bu işlemler zaman alırken, donanım uygulamalarının gerçekleştirilmesiyle bu zaman kısıtı ortadan kalkmaktadır. FPGA kullanımı ile donanım tekrar tekrar programlanabilir ve daha optimize ve işlevli şekilde çalışabilmektedir.

DWT algoritmasını FPGA üzerinde uygulamanın yolu, dalgacık ayrıştırma ve yeniden yapılandırmanın her aşamasının ayrı bir ardışık düzen aşamasında gerçekleştirildiği bir mimari kullanmaktır. Bu, her verinin işlenmesi diğer verinin işlenmesiyle çakışabileceğinden, yüksek verim ve düşük gecikme sağlamaktadır.

Yüksek performans ve düşük güç tüketimi için bir FPGA'da Haar DWT uygulanmasını optimize etmenin birkaç yolu bulunmaktadır. Bunlar:

1. **Ardışık Düzen:** Ardışık düzen, farklı verilerin işlenmesini üst üste bindirmek için kullanılabilir, bu da verimi artırabilmekte ve gecikmeyi azaltabilmektedir. DWT'nin her aşaması, her aşamada bir veriyi işleyerek ayrı bir ardışık düzen aşamasında uygulanabilmektedir. Bu, DWT'nin işlem hızını büyük ölçüde artırabilmektedir.

2. **Parallelleştirme:** Dalgacık ayrıştırma ve yeniden oluşturma işlemleri, çoklu işlem birimleri kullanılarak bir FPGA üzerinde paralelleştirilebilmektedir. Bu, tek bir veri için işlem süresini azaltabilmekte ve sistemin verimini artırabilmektedir.

3. **Sabit nokta aritmetiği:** DWT'nin uygulanması için gereken donanım kaynaklarını azaltmak için kayan nokta aritmetiği yerine sabit nokta aritmetiği kullanılabilir. Sabit noktalı aritmetik ayrıca kayan noktalı aritmetikten daha hızlı ve güç açısından daha verimli olabilmektedir.

4. **Verimli bellek kullanımı:** DWT, dalgacık katsayılarını ve giriş ve çıkış verilerini depolamak için önemli miktarda bellek gerektirmektedir. Bir FPGA'da bellek kullanımını optimize etmek için, veriler çip dışı bellek yerine çip içi bellekte saklanabilir. Bellek erişimi, veri ön getirme, veri sıkıştırma ve veri arabelleğe alma gibi teknikler kullanılarak da optimize edilebilmektedir.

5. **Donanım optimizasyonu:** DWT'nin donanım tasarımı, güç tüketimini azaltmak ve performansı artırmak için optimize edilebilmektedir. Düşük güçlü toplayıcılar ve çarpanlar gibi düşük güçlü bileşenlerin kullanılması, DWT'nin güç tüketimini azaltabilmektedir. Yüksek hızlı seri bağlantılar gibi yüksek hızlı ara bağlantıların kullanılması sistemin veri çıkışını artırabilmektedir.

Bu tez kapsamında birden çok örneği aynı anda işlemek için birden çok paralel DWT modülü kullanılmaktadır. Bu şekilde verim artırılmıştır. Kullanılan hafıza modülleri için çip içi bellek (BRAM) kullanılmaktadır. Bu çip içi bellek kullanımı sayesinde optimize bir tasarım ortaya konulmuştur.

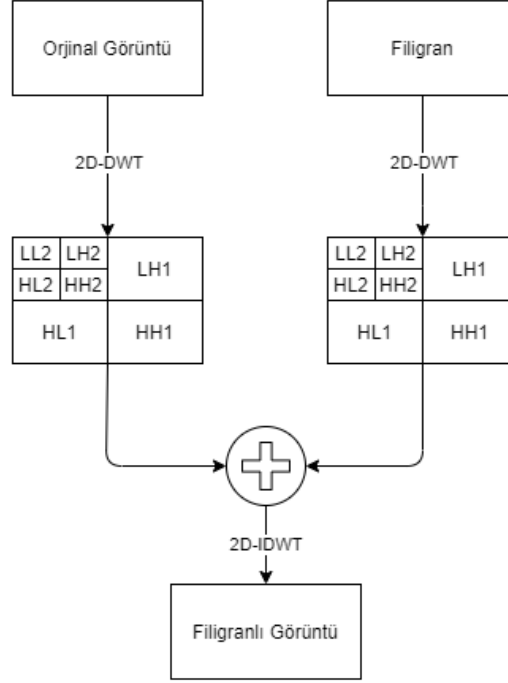
Genel olarak, bu teknikler, yüksek performans ve düşük güç tüketimi için DWT'nin bir FPGA'da uygulanmasını optimize etmek için kullanılabilir. Bununla birlikte, DWT'nin optimal uygulaması, uygulamanın özel gereksinimlerine ve mevcut donanım kaynaklarına bağlıdır.

Haar dalgacığı, dalgacık ayrıştırma ve yeniden yapılandırma için kullanılan bir çift filtre ile tanımlanmaktadır. Ayrıştırma filtresi $[1, 1]$ olarak tanımlanır ve yeniden oluşturma filtresi $[1, -1]$ olarak tanımlanmaktadır. Bu filtreler, giriş sinyalinin veya görüntünün ayrıntılarını temsil eden farklı ölçeklerde bir dalgacık katsayıları dizisi oluşturmak için kullanılmaktadır.

Haar dalgacığı, onu birçok uygulamada yararlı kılan bazı önemli özelliklere sahiptir. Bunlar:

1. Ortogonalite: Haar dalgacığı ortogonal bir dalgacık fonksiyonudur, yani mükemmel yeniden yapılandırma koşulunu karşılamaktadır. Bu özellik, sinyal veya görüntünün doğruluğunun kritik olduğu uygulamalarda önem taşımaktadır.
2. Kompakt destek: Haar dalgacığı kompakt bir desteğe sahiptir, bu da yalnızca sonlu bir aralıkta sıfır olmadığı anlamına gelmektedir. Bu özellik onu hesaplama açısından verimli kılar ve dalgacık katsayılarını depolamak için gereken bellek miktarını azaltmaktadır.
3. Çoklu çözünürlük: Haar dalgacığı, çoklu çözünürlüklü analiz için kullanılabilir, bu da bir sinyali veya görüntüyü farklı ölçeklerde analiz etmek için kullanılabileceği anlamına gelmektedir. Bu özellik, görüntü sıkıştırma, gürültü giderme ve özellik çıkarma gibi uygulamalarda kullanışlı hale gelmiştir.

Genel olarak, Haar DWT, FPGA tabanlı sinyal işleme uygulamaları için uygun bir algoritmadır. Yüksek performans ve düşük güç tüketimi için optimize edilebilir. Yapılan uygulamaya ait temel mimari Şekil 3.16'da gösterildiği gibidir.



Şekil 3.16 Filigranlama İşlemi Mimarisi

Ses, görüntü, video ve metin üzerinde yapılabilen filigranlama yöntemleri mevcut olmasına rağmen literatürde olan çalışmaların büyük çoğunluğu görüntü üzerine yapılmaktadır [20,38,59,68]. Video filigranlama konusunda yapılan araştırmalar, görüntü filigranlama konusunda gerçekleştirilen yöntemlerden yola çıkılarak gerçekleştirilmektedir.

Video filigranlama, yayın izlenme ve gerçek zamanlı video filigranlama için bir uygulama olarak çalışması yapılmıştır [69]. Bu çalışma ile FFT tabanlı filigranlama üzerine durulmuştur. Kayan dalga aritmetiği kullanılmıştır.

Haar dalgacık tabanlı gerçek zamanlı video filigranlama için bir FPGA prototiplemesi sunulmuştur [70]. Bu çalışmada DWT tabanlı filigranlama tasarımı sunulmuştur. Yapılan çalışma ile filigran gömme işleminin gerçek zamanlı videolar üzerinden gerçekleştirildiği gösterilmiştir. Gömülen filigranın sonucunda elde edilen filigranlı görüntüye bakıldığında algısal görünürlüğü konusunda sorunlar görülmüştür.

[71] çalışmasında, 5 farklı sayısal filigranlama yöntemi DCT tekniği ile donanım üzerinde gerçekleştirilmiştir. Çalışmada biyometrik parmak izi verisi gibi veriler kullanılarak sayısal filigranlama sisteminin uygulaması gerçekleştirilmiştir.

[72] çalışmasında, hybrid bir method yani DCT ve DWT'nin birlikte kullanıldığı bir video tabanlı uygulama gerçekleştirilmiştir. Bu çalışma ile PSNR ve MSE değerlerinin video kalitesine bağlı olarak çerçeve görüntülerinden nasıl hesaplandığı gösterilmektedir.

3.3. Ters Dalgacık Dönüşüm Modülü – IDWT Haar

Bu modülde, filigranın orijinal görüntüden çıkarılması işlemi bulunmaktadır.

$$LL_{W_e} = (LL_S - LL_H)/a \quad (3.11)$$

LL_{W_e} filigranlı görüntüden çıkarılan filigranın LL bandını ifade etmektedir. Orijinal filigranın LH,HL ve HH bant sonuçları ile birlikte oluşturulduğunda, ters dalgacık dönüşümü işleme alındığında filigran elde edilmiş olmaktadır.

Ters Ayrık Dalgacık Dönüşümü (IDWT), dalgacık katsayılarından orijinal sinyali veya görüntüyü yeniden oluşturma işlemidir. DWT'nin tersi işlemidir.

IDWT, DWT'de kullanılan dalgacık ayrıştırma filtrelerinin tersi olan bir dizi dalgacık rekonstrüksiyon filtresi kullanılarak gerçekleştirilebilmektedir. Yeniden oluşturma filtreleri tipik olarak alçak geçiren filtre için [1, 1] ve yüksek geçiren filtre için [1, -1] olarak tanımlanmaktadır. Bu filtreler, farklı ölçeklerdeki dalgacık katsayılarından orijinal sinyali veya görüntüyü yeniden oluşturmak için kullanılmaktadır.

IDWT doğrusal bir işlemdir ve matematiksel olarak şu şekilde ifade edilebilmektedir:

$$x[n] = \left(\frac{1}{\sqrt{2}}\right)^J * w_J[n] + v_J[n] \quad (3.12)$$

Burada $x[n]$, yeniden yapılandırılmış sinyal veya görüntüdür, $w_J[n]$ ve $v_J[n]$, DWT'nin J'inci seviyesindeki dalgacık katsayılarıdır ve J, DWT'nin seviye sayısıdır. $\left(\frac{1}{\sqrt{2}}\right)^J$ terimi, yeniden oluşturulan sinyal veya görüntünün orijinal sinyal veya görüntü ile aynı enerjiye sahip olmasını sağlayan bir normelleştirme faktörüdür.

IDWT, DWT'ye benzer bir yaklaşım kullanılarak uygulanabilmektedir. Dalgacık katsayıları, önceki seviyede yeniden oluşturulmuş sinyali veya görüntüyü oluşturmak için önce dalgacık yeniden yapılandırma filtreleri ile örneklenir ve dönüşüme tabi tutulmaktadır. Bu işlem, orijinal sinyal veya görüntü yeniden oluşturulana kadar DWT'nin tüm seviyeleri için tekrarlanmaktadır.

IDWT aynı zamanda ortogonalite ve mükemmel rekonstrüksiyon gibi birçok sinyal ve görüntü işleme uygulamasında avantaj sağlayan bazı önemli özelliklere de sahiptir. IDWT, diğer uygulamaların yanı sıra görüntü sıkıştırma, gürültü giderme ve yeniden yapılandırma için kullanılabilir.

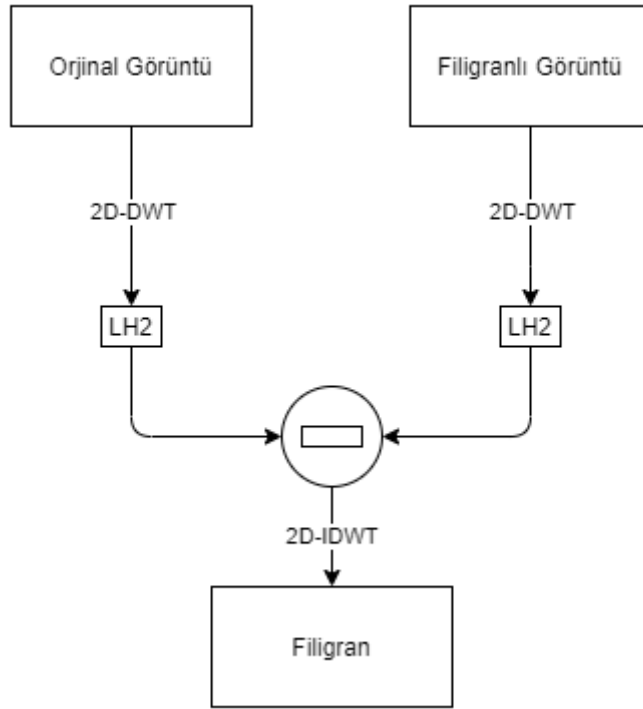
IDWT, dalgacık katsayılarından orijinal görüntüyü yeniden oluşturmak için kullanılmaktadır. Yeniden oluşturma işlemi, orijinal görüntüyü yeniden oluşturmak için düşük frekanslı ve yüksek frekanslı alt bantları birleştiren dalgacık katsayılarına ters dalgacık dönüşümünün uygulanmasını içermektedir.

Ters Haar dalgacıđı, onu birçok uygulamada yararlı kılan bazı önemli özelliklere sahiptir. Bunlar:

1. Ortogonallık: IDWT ortogonal bir dönüşümdür, yani orijinal sinyal veya görüntünün enerjisini korumaktadır. Bu özellik, yeniden oluşturulmuş sinyal veya görüntünün doğruluğunun ve aslına uygunluğunun kritik olduğu uygulamalarda önemlidir.
2. Mükemmel Rekonstrüksiyon: IDWT, mükemmel rekonstrüksiyon koşulunu karşılar; bu, yeniden oluşturulan sinyal veya görüntünün, bir ölçeklendirme faktörüne kadar orijinal sinyal veya görüntüyle aynı olduğu anlamına gelmektedir. Bu özellik, dalgacık dönüşümü sırasında bilgi kaybı olmamasını sağlamaktadır.
3. Çoklu Çözünürlük Analizi: IDWT çoklu çözünürlüklü analiz için kullanılabilir, bu da bir sinyali veya görüntüyü farklı ayrıntı seviyelerinde yeniden oluşturmak için kullanılabileceđi anlamına gelmektedir. Bu özellik, farklı ayrıntı düzeylerinin farklı sıkıştırma oranları kullanılarak sıkıştırılabildiđi görüntü sıkıştırma gibi uygulamalarda kullanışlıdır.
4. DWT'nin Ters İşlemi: IDWT, Ayrık Dalgacık Dönüşümünün (DWT) ters işlemidir. DWT, orijinal sinyali veya görüntüyü farklı ölçeklerde dalgacık katsayılarına ayrıştırırken, IDWT orijinal sinyali veya görüntüyü bu dalgacık katsayılarından yeniden oluşturmaktadır.
5. Hesaplama Verimliliği: IDWT, hesaplama açısından verimlidir ve DWT'ye benzer hızlı bir algoritma kullanılarak uygulanabilmektedir. Bu özellik, onu gerçek zamanlı sinyal ve görüntü işleme uygulamaları için uygun hale getirmektedir.

Genel olarak IDWT, sinyal ve görüntü işleme uygulamalarında güçlü bir araçtır. Sinyal ve görüntü sıkıştırma, gürültü giderme, özellik çıkarma ve diğer birçok uygulama için kullanılabilir. IDWT, DWT ile birlikte, sinyal ve görüntü analizi için güçlü bir teknik olan ayrık dalgacık dönüşümünün temelini oluşturmaktadır.

Yapılan işlemlere ait temel mimari Şekil 3.17’de gösterildiği gibidir.

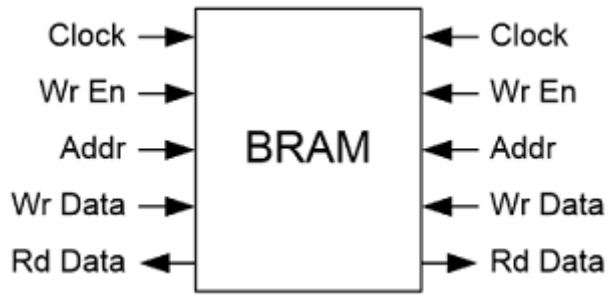


Şekil 3.17 Filigranla Çıkarma İşlemi Mimarisi

IDWT işleminin sonucunda verilerde kurcalama yapıp yapılmadığı anlaşılabilir. Örneğin filigranlanmış bir gerçek zamanlı video verisinde, alıcı taraf bu filigranlı video verisini orjinal veri ile birlikte IDWT işlemine sokabilir. Bu IDWT işlemi sonucunda, elde edilen filigran verisinin, çerçeve görüntülerinden (frame) çıkarılma yapıp yapılmadığı kolayca tespit edilebilir. Bu sayede gerçek zamanlı video verileri üzerinde kurcalama saldırısı yapıp yapılmadığı tespit edilebilmektedir. İlerleyen bölümlerde bu çalışma ile ilgili sonuçlar yer almaktadır.

3.4. Block RAM Modülü (BRAM)

Bu modülde, Xilinx'in sunmuş olduğu hazır bir hafıza bloğudur. Kullanıcının konfigürasyonuna bağlı olarak ayarı değiştirilebilir özelliktedir. Arayüz modeli olarak "Native", hafıza tipi olarak da "Simple Dual Port RAM" seçilmiştir. Bu seçim sayesinde Block RAM'e Dalgacık dönüşümü yapılan veriler, ters Dalgacık dönüşümü yapılan veriler ve orijinal görüntüler yazılabilecek ve okunabilecektir. Kullanılan hafıza modülüne ait yapı Şekil 3.18'deki gibidir.

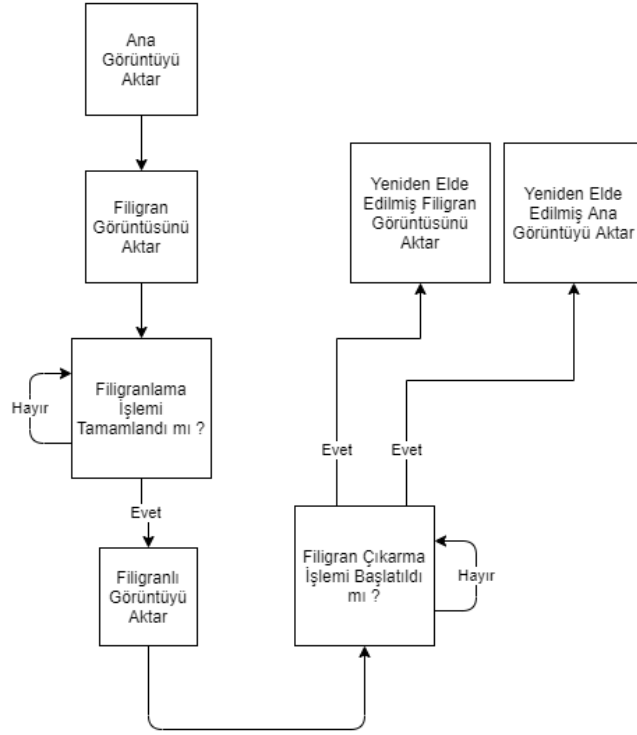


Şekil 3.18 Block RAM Yapısı

3.5. Sonuçların Değerlendirilmesi

Bu kısımda elde edilen sonuçlara ait değerlendirmeler yapılmaktadır.

Şekil 3.19'da gösterilmiş olan sistem mimarisi, bu çalışmada yapılan detayları göstermektedir.



Şekil 3.19 Sistem Tasarımı

















Bu tez çalışmasında, ilk olarak sabit bir ana görüntü seçilerek, bu görüntü üzerine akan filigran videosunun gömülmesi işlemi gerçekleştirilmiştir. Bu uygulama sayesinde, verilerdeki *kurcalama saldırılarına* karşı filigranlama işleminin öneminin gösterilmesi amaçlanmıştır.

Tablo 2’de ilk görülen veri ana görüntü olarak yani Lena kullanılmıştır. Sağ tarafında yer alan veriler ise akan filigran videosundan birer örnek görüntü verilmiştir. Tablo 3’de ise gerçek zamanlı akan video verileri üzerinde yapılan işleme ait görüntüler verilmiştir.









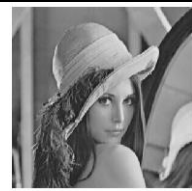
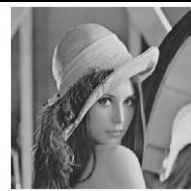
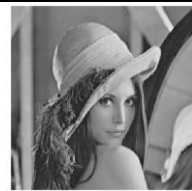





Ana görüntü ve filigran görüntüsü verileri UART arayüzü ile MATLAB tarafından FPGA tarafına aktarılmıştır. FPGA tarafında işlenen veriler sonucunda filigranlı görüntü önce FPGA üzerinde bulunan Block RAM’e kaydedilmiş, hafızaya kayıt edilmesinin ardından UART ile MATLAB ortamına aktarılmıştır. MATLAB da gösterimi yapılmıştır.

Filigranlı veriler FPGA içerisinde bulunan Block RAM'e kaydedilmiştir. Filigranlı veri ile birlikte ana veri ve filigran verisine IDWT işlemi uygulanmıştır. Filigranlı görüntüden IDWT işlemi ile filigran ve ana görüntü çıkarıldığında ise görseller elde edilmektedir. Bu görsellerden filigran görüntüsünün tekrardan elde edilmesi ve ana görüntünün tekrar elde edilmesine ait sonuçlar yer almaktadır. Bu işlemler sırasında FPGA'da bulunan Block RAM modülünde; ana görüntü, filigran görüntüsü ve filigranlı görüntüye ait tüm verileri tutulmaktadır. Bu sebeple yeniden elde etme işlemini hem ana görüntü hem de filigran görüntüsü için yapılabilmektedir.

Tablo 2 Kurcalama Saldırılarına ait Örnek Çalışma

Ana Görüntü: 	Filigran Görüntüsü: (1.çerçeve örneği) 	Filigran Görüntüsü: (2.çerçeve örneği) 	Filigran Görüntüsü: (3.çerçeve örneği) 	Filigran Görüntüsü: (4.çerçeve örneği) 	Filigran Görüntüsü: (5.çerçeve örneği) 
Filigranlı Görüntü Sonuçları:					
Yeniden Elde Edilen Filigran Görüntüsü:					

Tablo 3 Kurcalama Saldırılarına ait Gerçek Zamanlı Akan Videodan Örnek Çalışma

















Ana Görüntü: 	Filigran Görüntüsü: (1.çerçeve örneği) 	Filigran Görüntüsü: (2.çerçeve örneği) 	Filigran Görüntüsü: (3.çerçeve örneği) 	Filigran Görüntüsü: (4.çerçeve örneği) 	Filigran Görüntüsü: (5.çerçeve örneği) 
Filigranlı Görüntü Sonuçları:					
Yeniden Elde Edilen Filigran Görüntüsü:					

Bu çalışmaya ek olarak, sabit bir filigran görüntüsü ve akan bir videodan örneklenen ana görüntü seçilerek ile filigranlama işlemi de gerçekleştirilebilir. Bu uygulama sayesinde filigranlama işleminin, medya verilerindeki *kimlik doğrulama* sorunu hakkında öneminin gösterilmesi amaçlanmıştır. Tablo 4’de ilk yer alan filigran görüntüsü olarak Lena verilmiştir. Sonrasında ise akan videodan birer örnek görüntü verilmiştir. Tablo 5’de ise gerçek zamanlı akan video verileri üzerinde yapılan işleme ait görüntüler verilmiştir.

















Ana görüntü ve filigran görüntüsü verileri UART arayüzü ile MATLAB tarafından FPGA tarafına aktarılmıştır. FPGA tarafında işlenen veriler sonucunda filigranlı görüntü önce FPGA üzerinde bulunan Block RAM’e kaydedilmiş, hafızaya kayıt edilmesinin ardından UART ile MATLAB ortamına aktarılmıştır. MATLAB da gösterimi yapılmıştır. Filigranlı veriler FPGA içerisinde bulunan Block RAM’e kaydedilmiştir. Filigranlı veri ile birlikte ana veri ve filigran verisine IDWT işlemi uygulanmıştır. Filigranlı görüntüden IDWT işlemi ile filigran ve ana görüntü çıkarıldığında ise görseller elde edilmektedir. Bu görsellerden filigran görüntüsünün tekrardan elde edilmesi ve ana görüntünün tekrar elde edilmesine ait sonuçlar yer almaktadır.

Bu işlemler sırasında FPGA’da bulunan Block RAM modülünde; ana görüntü, filigran görüntüsü ve filigranlı görüntüye ait tüm verileri tutulmaktadır. Bu sebeple yeniden elde etme işlemini hem ana görüntü hem de filigran görüntüsü için yapılabilmektedir.

Tablo 4 Veri Doğrulama Kapsamında Örnek Çalışma

Filigran Görüntüsü:	Ana Görüntü: (1.çerçeve örneği)	Ana Görüntü: (2.çerçeve örneği)	Ana Görüntü: (3.çerçeve örneği)	Ana Görüntü: (4.çerçeve örneği)	Ana Görüntü: (5.çerçeve örneği)
					
Filigranlı Görüntü Sonuçları:					
Yeniden Elde Edilen Filigran Görüntüsü:					

Tablo 5 Veri Doğrulama Kapsamında Gerçek Zamanlı Akan Videodan Örnek Çalışma

Filigran Görüntüsü: 	Ana Görüntü: (1.çerçeve örneği) 	Ana Görüntü: (2.çerçeve örneği) 	Ana Görüntü: (3.çerçeve örneği) 	Ana Görüntü: (4.çerçeve örneği) 	Ana Görüntü: (5.çerçeve örneği) 
Filigranlı Görüntü Sonuçları:					
Yeniden Elde Edilen Filigran Görüntüsü:					

Ek olarak filigranlı görüntüden ana görüntü çıkarıldığında elde edilen görüntü, filigranı vermemektedir. Bu çalışmanın gerçek zamanlı akan video ile yapılan örnek çalışma sonucu Tablo 6’da gösterilmektedir. Bu çalışma ile birlikte ana görüntünün elde var olması durumunda dahi filigranlı görüntüden direkt çıkarma işlemi yapılarak filigranın elde edilemediği gösterilmektedir.

Tablo 6 Filigranlı Görüntüden Ana Görüntünün Çıkarılması İşlemi

Filigranlı Görüntü: 	Ana Görüntü: 	Filigran Görüntüsü: 	Filigranlı Görüntü - Ana Görüntü: 
--	--	--	---

4. SONUÇLAR VE ÖNERİLER

Küresel olarak gelişen teknoloji ile birlikte, bilgisayar ağları ve teknolojiler, hızlı yaygınlaşarak multimedya sistemlerinin popüleritesini arttırmıştır. Dijital kayıt ve depolama cihazları ile birlikte, verilerin çoğaltılması ve dağıtılması konusunda bu teknolojik ilerleyiş, beraberinde bilgi miktarını ve iletimini arttırmıştır. Ancak bu artış beraberinde dijital ağlar üzerinden iletilen verilerin aidiyetinin (authentication) doğrulanması ve verinin gürbüzlüğü (robustness) sağlanması gerekliliği ortaya çıkmaktadır. Dolayısı ile bu sorunları çözebilmek için özel teknikler kullanılmalıdır. Haberleşme teknolojisindeki ve bilgi alışverişindeki hızlı gelişmeler ile bilgi güvenliği alanındaki araştırmalar artmıştır. Bu nedenle, verilere dışarıdan bakıldığında göze çarpmayacak bir şekilde veri gömülmesi yani sayısal filigranlama gibi tekniklerin kullanımı zorunlu hale gelmiştir. Aynı zamanda Filigranlama ile gömülen gizli bilgiler, sıkıştırma, kırpma ve filtre saldırılarına karşı dayanıklıdır. Bu şekilde verilerde yapılabilecek herhangi bir değişikliği ortaya çıkarmak veya verilerin içerisine yerleştirilen bilgi sayesinde verinin kaynağını tespit etmek mümkün olabilmektedir.

Bu tezde, farklı filigranlama tekniklerinden ayrık dalgacık dönüşümü gözden geçirilmiş ve analiz edilmiştir. Bu tekniğin, FPGA üzerinde gerçekleşmesi test edilmiştir. Gerçek zamanlı akan veriler üzerinden kolaylıkla kullanılabilmesi gösterilmiştir. Görünür filigran kullanımda görüntülerde kırpma gibi işlemler yapıldığında filigran verisini ortadan kaldırmak mümkün iken, görünmez filigranlama uygulamalarında filigran ana görüntüye gömüldüğü için karşı saldırılara karşı gürbüz olduğu bu çalışmada gösterilmektedir.

Bu tez çalışmasında, gerçek zamanlı akan video verileri üzerine sabit bir filigran verisi gömülerek kimlik doğrulama (authentication) sorunlarına dikkat çekilmiştir. Aynı zamanda sabit bir görüntüye, gerçek zamanlı akan video verileri filigran olarak eklenerek kurcalama saldırılarına yönelik bir donanım çalışması gerçekleştirilmiştir.

Bu tez kapsamında yapılan çalışmalarda donanımda gerçekleştirilen filigranlama algoritmasının, kimlik doğrulama ve veri bütünlüğünü sağlama gerektiren IoT uygulamalarında kullanılabilmesi gösterilmiştir.

5. KAYNAKLAR

- [1] Graham, J. "Preserving the Aftermarket in Copyrighted Works: Adapting the First Sale Doctrine to the Emerging Technological Landscape", *Proceeding of Stanford Technology Law Review*, 2002.
- [2] Chang and J. C. Chuan, "An image intellectual property protection scheme for gray-level images using visual secret sharing strategy," *Pattern Recognition Letters*, vol. 23, June 2002, pp. 931-941.
- [3] R. G. van Schyndel, A. Z. Tirkel and C. F. Osborne, "A digital watermark," *Proceedings of 1st International Conference on Image Processing*, vol.2, 1994, pp. 86-90.
- [4] Langelaar, G.C., Setyawan, I., Lagendijk, R.L., "Watermarking digital image and video data a state-of-the-art overview", *IEEE Signal Processing Magazine*, 17(5), pp. 20-46.
- [5] Moulin, P., Koetter, R., "Data-hiding codes", *Proceedings of the IEEE*, 93(12), pp. 2083-2126.
- [6] Min Wu and Bede Liu, "Data hiding in image and video: Fundamental issues and solutions," in *IEEE Transactions on Image Processing*, vol. 12, no. 6, June 2003, pp. 685-695.
- [7] R. B. Wolfgang, C. I. Podilchuk and E. J. Delp, "Perceptual watermarks for digital images and video," in *Proceedings of the IEEE*, vol. 87, no. 7, July 1999, pp. 1108-1126.
- [8] N. Nikolaidis, I. Pitas, "Robust Image Watermarking in Spatial Domain", *International journal of signal processing*, 66(3), 1988, pp. 385-403.
- [9] N.Komatsu and H.Tominaga "Authentication system using concealed images in telematics" *Memories of the School of Science and Engineering*, Waseda University, 1988.
- [10] Patrick J. Van Fleet, "Discrete Haar Wavelet Transforms", *PREP - Wavelet Workshop*, 2006.
- [11] E. Koch, J. Rindfrey, E. Zhao, "Copyright Protection for Multimedia Data," *Proc. of the International Conference on Digital Media and Electronic Publishing*, Leeds, UK, December 1994.
- [12] Shieh, C.-S., Huang, H.-C., Wang, F.-H., Pan, J.-S., "Genetic watermarking based on transform-domain techniques", *Pattern recognition*, 37 (3), March 2004, pp. 555-565.
- [13] Hyun Lim, Soon-Young Park, Seong-Jun Kang and Wan-Hyun Cho, "FPGA implementation of image watermarking algorithm for a digital camera," *2003 IEEE Pacific Rim Conference on Communications Computers and Signal Processing (PACRIM 2003)* (Cat. No.03CH37490), Victoria, BC, Canada, vol. 2, 2003, pp. 1000-1003.
- [14] Tamilvanan and R. B. Selvakumar. "FPGA implementation of digital watermarking system." *International Journal of Computer Science and Mobile Computing (IJCSMC)* vol. 3.4, April 2014, pp. 1321-1327.
- [15] S. D. Roy, X. Li, Y. Shoshan, A. Fish and O. Yadid-Pecht, "Hardware Implementation of a Digital Watermarking System for Video Authentication," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 2, Feb. 2013, pp. 289-301.
- [16] Enas Dhuhri Kusuma, Thomas Sri Widodo, "FPGA Implementation of Pipelined 2D-DCT and Quantization Architecture for JPEG ImageCompression", *IEEE*, 2010.
- [17] Pankaj U. Lande, Sanjay N. Talbar, G.N. Shinde, "FPGA Implementation of Image Adaptive Watermarking Using Human Visual Model", *ICGST-PDCS Journal*, vol.9, Issue 1, Oct 2009.

- [18] R. K. Megalingam, M. M. Nair, R. Srikumar, V. K. Balasubramanian and V. S. V. Sarma, "Performance Comparison of Novel, Robust Spatial Domain Digital Image Watermarking with the Conventional Frequency Domain Watermarking Techniques," 2010 International Conference on Signal Acquisition and Processing, Bangalore, India, 2010, pp. 349-353.
- [19] S. Sowmya and R. Paily, "FPGA implementation of image enhancement algorithms," 2011 International Conference on Communications and Signal Processing, Kerala, India, 2011, pp. 584-588.
- [20] V. M. Potdar, S. Han and E. Chang, "A Survey of Digital Image Watermarking Techniques", IEEE International Conference on Industrial Informatics (INDIN) pp. 709-716.
- [21] Wasfy, Wael, and Hong Zheng. "General structure design for fast image processing algorithms based upon FPGA DSP slice." *Physics Procedia* 33, 2012, pp. 690-697.
- [22] Keyvanpour, Mohammad-Reza, and Farnoosh Merrikh-Bayat "Robust dynamic block-based image watermarking in DWT domain." *Procedia Computer Science* 3, 2011, pp. 238-242.
- [23] A. Garimella, M. V. V. Satyanarayan, R. S. Kumar, P. S. Muruges, and U. C. Niranjan, "VLSI Implementation of Online Digital Watermarking Techniques With Difference Encoding for the 8-bit Gray Scale Images," in *Proceedings of the International Conference on VLSI Design*, 2003, pp. 792-796.
- [24] S. P. Mohanty, N. Ranganathan and R. K. Namballa, "A VLSI architecture for visible watermarking in a secure still digital camera ($S/\sup 2/DC$) design (Corrected)*," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 13, no. 8, Aug. 2005, pp. 1002-1012.
- [25] S. P. Mohanty, N. Ranganathan and K. Balakrishnan, "A dual voltage-frequency VLSI chip for image watermarking in DCT domain," in *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 53, no. 5, May 2006, pp. 394-398.
- [26] D. Ziener and J. Teich, "Power signature watermarking of IP cores for FPGAs", *J. Signal Process. Syst.* 51, 2008, pp. 123-136.
- [27] V.-S. Harsh, Y. Suman and M. Anand, "Review of FPGA-based data hiding data in digital images", *Int. J. Adv. Electron. Electr. Eng.* 3, 2014, pp. 5-9.
- [28] P-P. Chu and R.-E. Jones, "Design techniques of FPGA-based random number generator", NASA Glen Research Center, Cleveland, 1999.
- [29] J.-K. Gorre and U.-N. Devi, "FPGA hardware LSB steganography technique based on the lifting scheme", *Int. J. Eng. Res. Technol.* 2, 2013, pp. 2726-2730.
- [30] H.-C. Huang, S.-C. Chu, J.-S. Pan, C.-Y. Huang and B.-Y. Liao, "Tabu search based multi-watermarks embedding algorithm with multiple description coding", *Inf. Sci.* 181, 2011, pp. 3379-3396.
- [31] Chen, M., "Image steganography based on Arnold transform", *Proceeding of Appl. Res. Comput.*, 2006, pp. 235-237.
- [32] Lin, Q.H., et al., *Secure Image Communication Using Blind Source Separation*, *Proceeding of IEEE 6th CAS Symp.on Emerging Technologies: Mobile and Wireless COMM*, 2004, pp. 261-264.

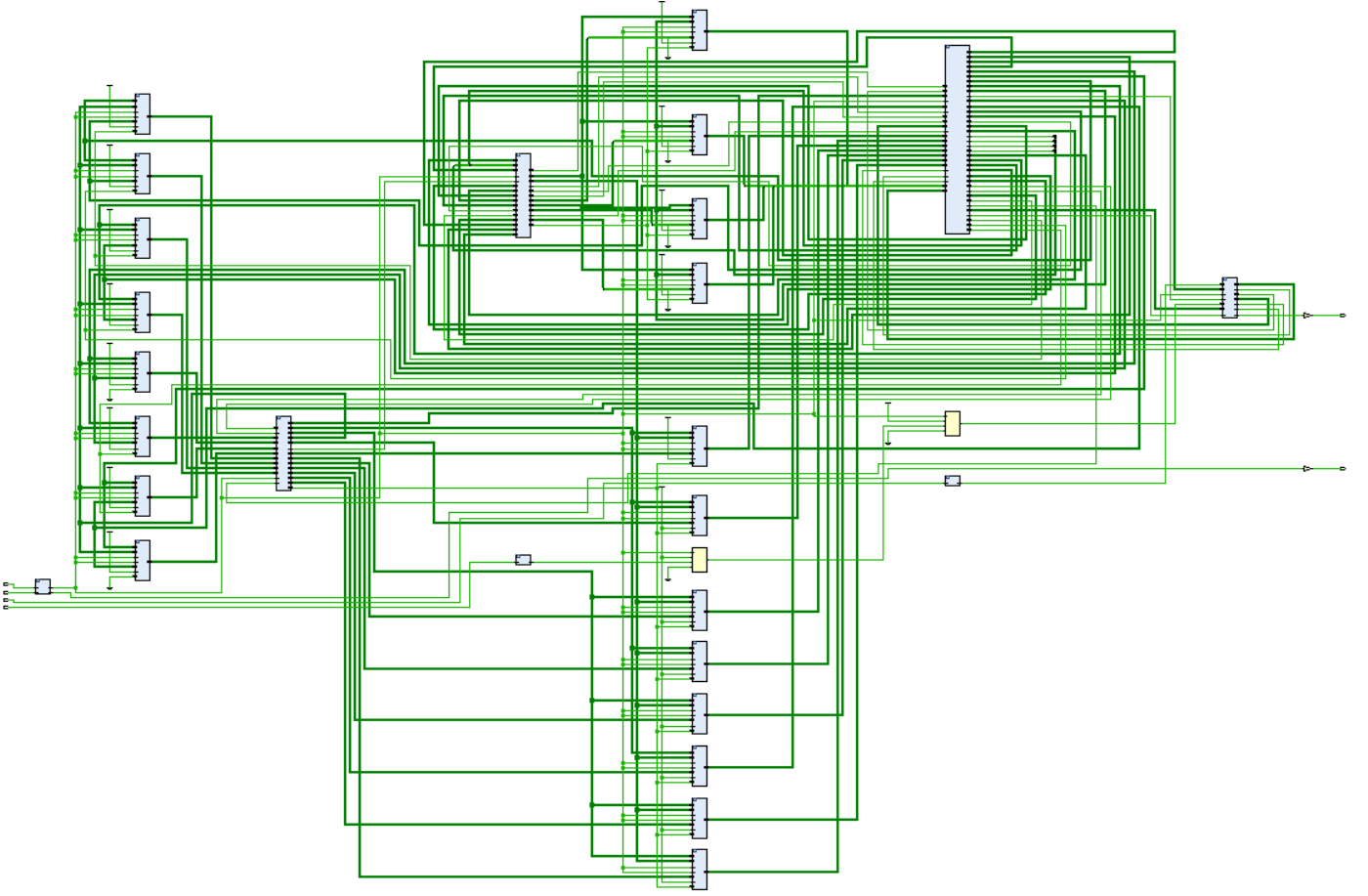
- [33] D. Singh, S.K. Singh, "Effective self-embedding watermarking scheme for image tampered detection and localization with recovery capability", *J. Vis. Commun. Image Represent.* 38, 2016, pp. 775-789.
- [34] C. Qin, C. Chang, P. Chen, "Self-embedding fragile watermarking with restoration capability based on adaptive bit allocation mechanism", *Signal Process.* 92, 2012, pp. 1137-1150.
- [35] J. Chang, B. Chen, C. Tsai, "LBP-based fragile watermarking scheme for image tamper detection and recovery," 2013 International Symposium on NextGeneration Electronics, 2013, pp. 173-176.
- [36] R. Chamlawi, I. Usman, A. Khan, "Dual watermarking method for secure image authentication and recovery," 2009 IEEE 13th International Multitopic Conference, 2009, pp. 1-4.
- [37] http://en.wikipedia.org/wiki/Enigma_Machine
- [38] M.sharkas,D.EIshafie, and N.Hamdy," A Dual Digital-Image Watermarking Technique," *World Academy of Science, Engineering and technology*,5, 2005, pp. 136-139.
- [39] FENG, J., LIN, I., TSAI, C., CHU, Y., "Reversible Watermarking: Current Status and Key Issues". vol. 2, no. 3, 2006, pp. 161-170.
- [40] Podilchuk, C.I., Delp, E.J., "Digital watermarking: algorithms and applications", *IEEE Signal Processing Magazine*, 18(4), 2001, pp. 33-46.
- [41] Tao, H., Chongmin, L.Zain, Abdalla A., "Robust Image Watermarking Theories and Techniques: A Review", *Journal of Applied Research and Technology.* 12, pp. 122-138.
- [42] M.A. Suhail,"Digital watermarking for protection of intellectual Property," University of Bradford, UK, 2005.
- [43] Lerch, A., "An introduction to audio content analysis applications in Signal Processing and Music Informatics (First edition)". Somerset: Wiley, 2012, pp. 199-200.
- [44] A. Grossmann, and J. Morlet, "Decomposition of Hardy functions into square integrable waveletsof constant shape", *SIAM J. Anal.* 15, 1984, pp. 723-736.
- [45] Stephane Mallat, "A Wavelet Tour of Signal Processing -The Sparse Way".
- [46] I. H. Witten, R. Neal, and J. G. Cleary, "Arithmetic coding for data modification," *Comm. ACM*, vol. 30, June 2009, pp. 520-540.
- [47] Shapiro, "Embedded image coding using zerotrees of WAVELET coefficients," *IEEE Trans Signal Processing.*, vol. 41, Dec. 2011, pp. 3445-3462.
- [48] S. Mallat, "A theory for multi resolution signal decomposition: The WAVELET representation," *IEEE Trans. Pattern Anal. Mach. Intell.*,vol. 11, July 2011, pp. 674-693.
- [49] Y. Xiang, I. Natgunanathan, D. Peng, W. Zhou, and S. Yu, "A dualchannel time-spread echo method for audio watermarking," *IEEE Trans. Information Forensics and Security*, vol. 7, no. 2, Apr. 2012, pp. 383-392.
- [50] Y. Xiang, D. Peng, I. Natgunanathan and W. Zhou, "Effective Pseudonoise Sequence and Decoding Function for Imperceptibility and Robustness Enhancement in Time-Spread Echo-Based Audio Watermarking," in *IEEE Transactions on Multimedia*, vol. 13, no. 1, Feb. 2011, pp. 2-13.

- [51] N. K. Kalantari, S. M. Ahadi and M. Vafadust, "A Robust Image Watermarking in the Ridgelet Domain Using Universally Optimum Decoder," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 3, March 2010, pp. 396-406.
- [52] E. Nezhadarya, Z. J. Wang and R. K. Ward, "Robust Image Watermarking Based on Multiscale Gradient Direction Quantization," in *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 4, Dec. 2011, pp. 1200-1213.
- [53] L. Wang, H. Ling, F. zou and Z. Lu, "Real-Time Compressed- Domain Video Watermarking Resistance to Geometric Distortions," in *IEEE MultiMedia*, vol. 19, no. 1, Jan. 2012, pp. 70-79.
- [54] M.-J. Lee, K.-S. Kim and H.-K. Lee, "Digital Cinema Watermarking for Estimating the Position of the Pirate," in *IEEE Transactions on Multimedia*, vol. 12, no. 7, Nov. 2010, pp. 605-621.
- [55] Nitin A. Shelke, Dr. P. N.Chatur, "A survey on various digital video watermarking schemes", *International Journal of Computer Science & Engineering Technology*, vol. 4, no. 12, Dec. 2013.
- [56] V.-P. Prachi and S.-S. Agrawal, "Implementation of digital video watermarking scheme based on FPGA", *Int. J. Electr. Electron. Comput. Syst.* 1, 2013, pp. 99-103.
- [57] A. A. Zaidan, A. W. Najj, S. A. Hameed, F. Othman and B. B. Zaidan, "Approved undetectable-antivirus steganography for multimedia information" in *PE-le, International Conference on IACSIT Spring Conf.*, Singapore, April 2009, pp. 425-429.
- [58] B. B. Zaidan, A. A. Zaidan, F. Othman, R. Z. Raji, S. Mohammed and M. Abdulrazzaq, "Quality of image vs. quantity of data hidden in the image", *Proc. 2009 Int. Conf. Image Processing, Computer Vision and Patter Recognition (ICIPCVPR'09) (CSREA Press, LA, USA, 2009)*, pp. 343-350.
- [59] P.-U. Lande, S.-N. Talbar and G.-N., Shinde, "FPGA implementation of image adaptive watermarking using human visual model", *ICGST-PDCS J.* 9, 2009, pp. 17-22.
- [60] Shoshan, Y., Fish, A., Li, X., Jullien, G., and Yadid-Pecht, O. "Vlsi watermark implementations and applications", 2008.
- [61] S. P. Mohanty, N. Ranganathan and R. K. Namballa, "VLSI implementation of invisible digital watermarking algorithms towards the development of a secure JPEG encoder," *2003 IEEE Workshop on Signal Processing Systems (IEEE Cat. No.03TH8682)*, Seoul, South Korea, 2003, pp. 183-188.
- [62] N. J. Mathai, D. Kundur and A. Sheikholeslami, "Hardware implementation perspectives of digital video watermarking algorithms," in *IEEE Transactions on Signal Processing*, vol. 51, no. 4, April 2003, pp. 925-938.
- [63] A. Basu, T. S. Das, S. K. Sarkar, A. Roy and N. Islam, "FPGA prototype of visual information hiding," *2010 Annual IEEE India Conference (INDICON)*, Kolkata, India, 2010, pp. 1-4.
- [64] S. D. Roy, X. Li, Y. Shoshan, A. Fish and O. Yadid-Pecht, "Hardware Implementation of a Digital Watermarking System for Video Authentication," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 2, Feb. 2013, pp. 289-301.
- [65] Chu Zhenyong, Weng Muyun. "FPGA design and application" [M]. Xi'an University of Electronic Science and Technology Press, 2002.

- [66] S. S. Gonge and J. W. Bakal, "Robust Digital Watermarking Techniques by Using DCT and Spread Spectrum", *International Journal of Electrical, Electronics and Data Communication* volume 1, issue 2, 2013, pp. 126-130.
- [67] Manikanda, S., "An Efficient Watermarking Algorithm Based on DWT and FFT Approach", *International Journal on Computer Science and Engineering*, 6(06), 2014, pp. 211-216.
- [68] S. P. Mohanty, "Watermarking of Digital Images", Master's thesis, Dept of Electrical Engineering, Indian Institute of Science, India, 1998.
- [69] N. J. Mathai, A. Sheikholeslami and D. Kundur, "VLSI implementation of a real-time video watermark embedder and detector," *Proceedings of the 2003 International Symposium on Circuits and Systems*, 2003. ISCAS '03., Bangkok, Thailand, 2003, pp. II-II.
- [70] Y.-J. Jeong, K.-S. Moon and J.-N. Kim, "Implementation of Real Time Video Watermark Embedder Based on Haar Wavelet Transform Using FPGA," *2008 Second International Conference on Future Generation Communication and Networking Symposia*, Hinan, China, 2008, pp. 63-66.
- [71] Aydın O., "Digital watermarking on images and its embedded system implementation", 2018, Yüksek Lisans tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Ankara.
- [72] Ünal K., "Digital Video Watermarking", 2011, Yüksek Lisans tezi, Dokuz Eylül Üniversitesi, Fen Bilimleri Enstitüsü, İzmir.
- [73] S.S.M. Mousavi, A. Naghsh, "Watermarking techniques used in medical images" survey. *J. Of Dig. Imag.* 27(6), 2014, pp. 714-729.
- [74] S. Kumar and A. Dutta, "A novel spatial domain technique for digital image watermarking using block entropy", *International Conference on Recent Trends in Information Technology (ICRTIT)*, Chennai, 2016, pp. 1-4.

EKLER

EK 1 – Donanım Tasarıma ait Şematik



EK 2 – Tez Çalışması Orjinallik Raporu