

**VISUAL C# 2005 KULLANILARAK ÇOK BOYUTLU
GÖRÜNTÜLER İÇİN JPEG2000 STANDARDINI DESTEKLEYEN
GÖRÜNTÜ İŞLEME UYGULAMASI**

**Pamukkale Üniversitesi
Fen Bilimleri Enstitüsü
Yüksek Lisans Tezi
Bilgisayar Mühendisliği Anabilim Dalı**

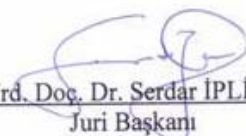
Evgin GÖÇERİ

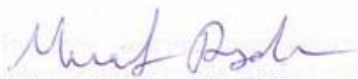
Danışman: Yrd. Doç. Dr. A. Kadir YALDIR


**Ağustos 2006
DENİZLİ**

YÜKSEK LİSANS TEZİ ONAY FORMU


Evgin GÖÇERİ tarafından Yrd. Doç. Dr. A. Kadir YALDIR ve Herman BOYEN yönetiminde hazırlanan "**Visual C# 2005 Kullanılarak Çok Boyutlu Görüntüler İçin JPEG2000 Standardını Destekleyen Görüntü İşleme Uygulaması**" başlıklı tez tarafımızdan okunmuş, kapsamı ve niteliği açısından bir Yüksek Lisans Tezi olarak kabul edilmiştir.


Yrd. Doç. Dr. Serdar İPLİKÇİ
Juri Başkanı


Yrd. Doç. Dr. Murat AYDOS
Jüri Üyesi


Yrd. Doç. Dr. A. Kadir YALDIR
Juri Üyesi (Danışman)

Pamukkale Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun
...09.08.2006... tarih ve ...16/18... sayılı kararıyla onaylanmıştır.


Prof. Dr. Mehmet Ali SARIGÖL
Müdür

Bu tezin tasarımı, hazırlanması, yürütülmesi, arařtırmalarının yapılması ve bulgularının analizlerinde bilimsel etięe ve akademik kurallara özenle riayet edildiđini; bu çalışmanın doğrudan birincil ürünü olmayan bulguların, verilerin ve materyallerin bilimsel etięe uygun olarak kaynak gösterildiđini ve alıntı yapılan çalışmalara atfedildiđini beyan ederim.

İmza: 

Öğrenci Adı Soyadı: Evgin GÖÇERİ

TEŞEKKÜR

Hayatımdaki herseyi borçlu olduğum sevgili annem ve babama, varlıklarıyla beni şanslı bir kardeş yapan ablam Emine, kardeşlerim Esin ve Numan'a, emeklerinin karşılığını hiçbir zaman ödeyemeyeceğim değerli danışman Hocam Yrd. Doç. Dr. A. Kadir Yıldır ve bugüne kadar beni destekleyen diğer tüm hocalarıma, arkadaşlarıma, arkadaşça ve sıcakkanlı davranışlarından dolayı Herman Boyen, Kristen Bauwens, Jan Boutsen ve Turan Aşçıođlu'na, komşularım Turgut Saygıner, İlknur Şurupçu ve bu güzel ülkede bana öz ablam ve abim gibi davranıp, yardımlarını hiç bir zaman esirgemeyen sevgili Hilal Yazıcı ve Mahmut Koçaslan'a sonsuz tesekkürler.

ÖZET

VISUAL C# 2005 KULLANILARAK ÇOK BOYUTLU GÖRÜNTÜLER İÇİN JPEG2000 STANDARDINI DESTEKLEYEN GÖRÜNTÜ İŞLEME UYGULAMASI

Göçeri, Evgin

Yüksek Lisans Tezi, Bilgisayar Mühendisliği ABD

Tez Yöneticisi: Yrd. Doç. Dr. A. Kadir YALDIR

Ağustos 2006, 66 Sayfa

Sayısal ortamda her gün büyük miktarda bilgi depolanıp, işlenir ve iletilir. Bu bilgilerin çoğunun sayısal görüntüler olması nedeniyle çeşitli görüntü işleme problemlerine ek olarak depolama ve iletim gereksinimleri de her geçen gün artmaktadır.

Görüntü sıkıştırma bir sayısal görüntünün gösterilmesinde gereksinim duyulan veri miktarının azaltılmasına yönelik problemleri ortaya koymaktadır. Artan çoklu ortam (multimedia) teknolojisi kullanımı ile görüntü sıkıştırma yeni özelliklerin yanı sıra yüksek performansa da gereksinim duymaktadır. Bu gereksinimleri karşılamak için görüntü kodlama alanında JPEG2000 olarak bilinen yeni bir standart geliştirilmiştir.

Bu tez çalışmasında kenar belirleme ve görüntüdeki gürültü ve olumsuz etkileri azaltmak için sayısal filtreleme gibi önemli görüntü işleme tekniklerini sağlayan bir uygulama geliştirilmiştir. Geliştirilen uygulamanın en önemli özelliklerinden bir tanesi diğer görüntü tiplerine ek olarak, JPEG2000 standardını desteklemesi ve sadece doğal, bilimsel, uzaktan algılamalı görüntüler üzerinde değil aynı zamanda çok boyutlu tıbbi görüntüler üzerinde de çalıştırılabilmesidir.

Uygulamanın diğer bir özelliği ise Lifting Scheme ve Dalgacık tiplerinin kodlanması sayesinde sadece iki boyutlu (2D) tıbbi görüntülerin sıkıştırılması değil aynı zamanda, zaman bilgisi ile iki boyutlu (2D in time) görüntüler, üç boyutlu (3D) görüntüler ve hatta zaman bilgisi ile üç boyutlu (3D in time) görüntüler üzerinde sıkıştırma işleminin gerçekleştirilebilmesidir.

Uygulamamızın son önemli özelliği ise diğer görüntü işleme programlarından farklı olarak kodlama için günümüzdeki en son teknoloji olan Microsoft Visual Studio 2005'in kullanılmış olmasıdır.

Anahtar kelimeler: Görüntü İşleme, Kenar Belirleme, Sayısal Filtreler, Lifting Scheme, Dalgacıklar, JPEG2000, Çok Boyutlu Tıbbi Görüntüler, 3D Görüntüler, 2D Görüntüler, Visual Studio 2005

Yrd. Doç. Dr. A. Kadir YALDIR

Yrd. Doç. Dr. Murat AYDOS

Yrd. Doç. Dr. Serdar İPLİKÇİ

ABSTRACT

DEVELOPING AN IMAGE PROCESSING APPLICATION USING VISUAL C# 2005 SUPPORTING JPEG2000 STANDARD FOR MULTIDIMENSIONAL IMAGES

Göçeri, Evgin

M. Sc. Thesis in Computer Engineering
Supervisor: Asst. Prof. Dr. A. Kadir YALDIR
August 2006, 66 Pages

Everyday, an enormous amount of information is stored, processed and transmitted digitally. Much of this information is procured in the shape of digital images and therefore, in addition to several image processing problems, the storage and communications requirements are becoming immense.

Image compression addresses the problem of reducing the amount of data required to represent a digital image. With the increasing use of multimedia technologies, image compression requires high performance as well as new features. To meet these needs in the specific area of still image encoding, a new standard has been developed known as JPEG2000.

In this thesis we implemented an application which implies important image processing techniques such as edge detection and digital filtering in order to improve the imperfections that is the presence of noise and inadequate contrasts and contours. One of the main features of the application is to support the JPEG2000 standard, in addition to other image types, and the implementation does not only apply to natural, scientific and remote sensing images, but also to multi-dimensional medical images.

Another feature of the application is the implementation of Lifting Scheme and Wavelet types, so that we can not only use 2D medical images to compress with wavelets, but also 2D-images in time, 3D-images and even 3D-images in time for compression with our application.

The last important feature of our application is that, unlike other image processing programs, we used the latest technology, which is Microsoft Visual Studio 2005 (Visual C# 2005) for the implementation.

Keywords: Image Processing, Edge Detection, Digital Filters, Lifting Scheme, Wavelets, JPEG2000, Multi-dimensional medical images, 3D-images, 2D-images, Visual Studio 2005

Asst. Prof. Dr. A. Kadir YALDIR
Asst. Prof. Dr. Murat AYDOS
Asst. Prof. Dr. Serdar İPLİKÇİ

İÇİNDEKİLER

Sayfa No

Yüksek Lisans Tezi Onay Formu	i
Bilimsel Etik Sayfası	ii
Teşekkür	iii
Özet.....	iv
Abstract.....	vi
İçindekiler	viii
Tablolar Dizini.....	x
Şekiller Dizini	xi
Simgeler ve Kısaltmalar Dizini	xii
1. GİRİŞ.....	1
2. GÖRÜNTÜ İŞLEME TEKNİKLERİ.....	3
3. DALGACIKLAR	7
3.1 Giriş.....	7
3.2 Altbant (Subband) Kodlama	10
4. JPEG2000	13
4.1 JPEG2000 Nedir?.....	13
4.2 Niçin Görüntüleri Sıkıştırmaya Gereksinim Duyulur?	14
4.3 Niçin Başka Bir Standart?.....	14
4.4 JPEG2000 ile Hedeflenenler.....	16
4.5 JPEG2000'nin Bölümleri.....	16
4.5.1 Çekirdek kodlama sistemi.....	17
4.5.2 Uzantılar.....	17
4.5.3 Hareketli JPEG2000.....	18
4.5.4 Uyumluluk	18
4.5.5 Referans yazılımı	18
4.5.6 JPM (Karışık raster içerik).....	19
4.5.7 (İptal edilen bölüm).....	19
4.5.8 (JPSEC) güvenli JPEG2000	19
4.5.9 JPIP (Etkileşimli araçlar, API'ler ve protokoller).....	20
4.5.10 3D ve kayan ondalıklı veri.....	20
4.5.11 JPWL (Kablosuz).....	20
4.5.12 ISO tabanlı dosya biçimi.....	22
4.6 Kodlama.....	22
4.6.1 Ön işleme	23
4.6.2 Renk dönüşümü	24
4.6.3 Dalgacık dönüşümü	26
4.6.4 Orantılama.....	26
4.6.5 Gömülü blok kodlama.....	27
4.6.6 Oran kontrolü.....	29
4.6.7 Bit dizin organizasyonu	30
4.7 Kod Çözümleme	31

4.8 JPEG2000'in Özellikleri.....	32
4.8.1 Bir kere sıkıştır, pek çok yolla aç.....	32
4.8.2 İlgili bölgesi (ROI).....	32
4.8.3 Hatadan korunma.....	35
4.9 Sonuçlar	37
5. MICROSOFT VISUAL STUDIO 2005	38
5.1 .NET Framework'un Temelleri	38
5.2 Visual Studio 2005'in Getirdikleri	39
5.2.1 IDE'deki gelişmeler	39
5.2.2 Windows forms.....	41
5.2.3 .NET compact framework	41
5.2.4 Visual studio 2005 araçları (VSTO).....	43
5.2.5 Akıllı istemci	43
5.2.6 Derlemek ve hataları düzeltmek	43
6. GELİŞTİRİLEN YAZILIM HAKKINDA	45
6.1 Giriş.....	45
6.2 Median Filtre.....	46
6.3 Thresholding	47
6.4 İskeletleştirme	48
6.5 Parlaklık Düzeltme.....	49
6.6 Adaptive Smoothing	50
6.7 Bağlantılı Parça Etiketleme.....	51
6.8 Fourier Dönüşümü	52
6.9 Morfoloji	53
6.10 Renk Filtreleme.....	55
6.11 Kenar Belirginleştiriciler.....	56
6.12 Convolution Filtreler.....	58
6.13 İki Görüntünün Kombinasyonu	60
7. SONUÇ	63
KAYNAKLAR	64
ÖZGEÇMİŞ.....	66

ŞEKİLLER DİZİNİ

		Sayfa No
Şekil 3.1	İleri dalgacık dönüşümü	8
Şekil 3.2	Daubechies D4 ileri dalgacık dönüşümü	10
Şekil 3.3	Altbant kodlama	11
Şekil 4.1	JPEG2000 Standardının sıkıştırılma yöntemi	13
Şekil 4.2	JPEG2000'in kodlanması ve kod çözümlenmesi	21
Şekil 4.3	JPEG2000'in kodlama aşamaları	22
Şekil 4.4	Ön işleme aşamaları	23
Şekil 4.5	Bölümleme	23
Şekil 4.6	Görüntü kalitesi üzerine bölümlemenin etkisi	24
Şekil 4.7	Renk dönüşümü	25
Şekil 4.8	Dalgacık dönüşümü	25
Şekil 4.9	Orantılama yapısı	26
Şekil 4.10	Orantılama örneği	27
Şekil 4.11	Altbantların kod bloklara bölünmesine örnek	27
Şekil 4.12	Kod bloklar	28
Şekil 4.13	MSB düzleminden LSB düzlemine kod blokları	28
Şekil 4.14	Gömülü blok kodlama	28
Şekil 4.15	Bit dizin organizasyonu	30
Şekil 4.16	Kod dizin organizasyonu	31
Şekil 4.17	Kod çözümlenme	31
Şekil 4.18	Bir kere sıkıştır, pek çok yolla aç	32
Şekil 4.19	İlgi bölgesi	33
Şekil 4.20	İlgi bölgesi maskesinin kaydırılmış katsayıları	34
Şekil 4.21	İlgi bölgesi – örnek	34
Şekil 4.22	0.25 bpp'de sıkıştırılmış görüntü sonuçları	36
Şekil 4.23	0.125 bpp'de sıkıştırılmış görüntü sonuçları	36
Şekil 5.1	.NET mimarisi	38
Şekil 6.1	Görüntü işleme yazılımı	45
Şekil 6.2	Median filtre	48
Şekil 6.3	Threshold işlemi	49
Şekil 6.4	İskeletleştirme	50
Şekil 6.5	Parlaklık düzeltme	50
Şekil 6.6	Adaptive smoothing	52
Şekil 6.7	Bağlantılı parça etiketleme	53
Şekil 6.8	Fourier dönüşümü	54
Şekil 6.9	Morfolojik işlemler	54
Şekil 6.10	Renk filtreleme	57
Şekil 6.11	Sobel kenar belirginleştirici	58
Şekil 6.12	Canny kenar belirginleştirici	58
Şekil 6.13	Keskinleştirme	60
Şekil 6.14	Kenarlar	61

TABLolar DİZİNİ

	Sayfa No
Tablo 4.1 Sıkıştırılmamış görüntüler	14
Tablo 4.2 Bölümlemenin görüntü kalitesi üzerindeki etkisi	24

SİMGELER VE KISALTMALAR DİZİNİ

ADC	Analog Sayısal Dönüştürücü
ASP	Aktif Sunucu Sayfaları
CLR	Çalışma Zamanlı Ortak Dil
CWT	Sürekli Dalgacık Dönüşümü
DWT	Ayrık Dalgacık Dönüşümü
FCL	Framework Sınıf Kütüphaneleri
FEC	İleri Hata Düzeltme
FFT	Hızlı Fourier Dönüşümü
FTP	Dosya Transfer Protokolü
HH	Yatay Ve Düşey Yönde Yüksek Geçirgen
HL	Yatay Yönde Yüksek Geçirgen, Düşey Yönde Alçak Geçirgen
ICT	Terlenebilir Renk Dönüşümü
IDE	Bütünleşik Geliştirme Ortamı
ISO	Uluslararası Standartlaştırma Organizasyonu
JPM	JPEG2000 Karışık Raster İçerik
JPIP	JPEG2000 Etkileşimli Araçlar Ve Protokoller
JPSEC	Güvenlikli JPEG2000
JPWL	Kablosuz JPEG2000
LL	Yatay Ve Düşey Yönde Alçak Geçirgen
LH	Yatay Yönde Alçak Geçirgen, Düşey Yönde Yüksek Geçirgen
LSB	En Az Baskın Bit
MSB	En Çok Baskın Bit
MJ2	Hareketli JPEG2000
MJP2	Hareketli JPEG2000
PPM	Ana Paket Bölümünün Korunması
PPT	Paket Bölümlerinin Korunması
ROI	İlgi Bölgesi
SOP	Paket Başlangıcı
QMF	Dörtlü Ayna Filtresi
XML	Genişletilebilir Yapısal Dil
YcrCb	Parlaklık (Luminance), Kırmızı Renklilik, Mavi Renklilik

1. GİRİŞ

Görüntü işleme sayısal olarak depolanmış piksellerden oluşan resimler üzerinde çeşitli işlemlerin yapılması anlamına gelmektedir. Sayısal görüntü işleme çeşitli nedenlerden dolayı programlamada bir meydan okuma alanı olmuştur. Bu nedenlerden birincisi, sayısal görüntü işleme konusunun ilk grafiksel işletim sisteminin oluşturulmasına kadar beklemek durumunda kalması ve bilgisayar tarihinin sonlarında ortaya çıkmış olması olarak belirtilebilir. İkincisi, sayısal görüntü işleme özellikle gerçek zamanlı uygulamalar için çok dikkatli iyileştirmelerin yapılmasını gerektirir. Son olarak ise sayısal görüntü işleme iki boyutlu ortamlarda çalışmayı gerektirdiğinden bu durum sayısal filtreler ile çalışmayı karmaşık hale getirmektedir.

Bu çalışmada en basit seviyeden başlanarak matris convolution gibi en karmaşık seviyeye kadar önemli görüntü işleme teknikleri ve sayısal filtreleme işlemleri açıklanmıştır.

Bu tez çalışmasının en önemli katkısı, gerçekleştirilen uygulamanın hem bilinen görüntü tiplerini hem de JPEG2000 standardını destekliyor olmasıdır. Bu uluslararası standart ile görüntü kodlama sistemi sadece etkinlik için değil aynı zamanda ölçeklenebilirlik, bilgisayar ağlarındaki iletkenlik ve taşınabilir (mobil) araçlar için iyileştirilmiş görüntü sıkıştırma tekniklerindeki gelişmeleri gösterir. JPEG2000 yeni ve güçlü bir araç olmakla birlikte sayısal görüntüleme de Internet uygulamalarının bir parçası haline gelmiştir.

Bu nedenle bu tezde JPEG2000 standardının sadece bölümlene (tiling), çoklu komponent dönüşümü, dalgacık (wavelet) dönüşümü, orantılama (quantization) ve entropi kodlama ile mimarisi değil aynı zamanda standardın ilgi bölgesinin (region-of-interest) kodlanması, ölçeklenebilirlik, görsel ağırlıklama (visual weighting), hata direnci ve dosya biçimlendirme yönleri gibi en önemli özellikleri de belirtilmiştir.

Bu tezin diğ er bir önemli katkısı ise uygulamanın iki boyutlu (2D) görüntüler, zaman bilgisi ile iki boyutlu (2D in time) görüntüler, üç boyutlu (3D) görüntüler ve hatta zaman bilgisi ile üç boyutlu (3D in time) görüntüler üzerinde sıkıştırma işlemlerinin yapılabilmesi olmasıdır. Bu görüntüler ayrıca dört boyutlu (4D) görüntüler olarakta adlandırılabilir. Seçilen dalgacık tipleri Daubechies dalgacık olarak adlandırılan 2’den 20’ye kadar olan katsayılı dalgacık tipleridir. Dalgacıklar veriyi her dönüşüm seviyesindeki her boyut için eşit büyüklükteki düzleştirilmiş (smoothed) ve detaylı parçalara böler. Her boyut için verinin boyutunu azaltmak amacıyla 2 faktörlü standart olmayan ayırma yöntemi kullanılmış olup bu işlem sadece sıkıştırılmış veri üzerinde tekrarlandı. Dönüşüm zamanını azaltan bu işlem standart ve standart olmayan ayırma yöntemi arasındaki en önemli farktır. Orjinal veriyi, sıkıştırılmış veriyi oluşturan en sondaki düzleştirilmiş (smoothed) parça ile ve diğ er detaylı parçalar ile tekrar oluşturabiliriz. Bu dalgacık tekniğ i yüksek geçirgen (high-pass), alçak geçirgen (low-pass), bant-geçiş (band-pass) ve notch filtreler yapmak için kullanılan sıfır doldurma tekniğ i ile birlikte kullanılır. Bu yaklaşım standart ayırma yaklaşımından daha hızlıdır.

Geleneksel görüntü işleme uygulamaları ile gerçekleştirdiğ imiz uygulama arasındaki diğ er bir fark ise kullanılan platformdur ki, uygulamamızda Microsoft Visual Studio 2005 kullandık. Bu teknoloji her çeşit uygulama için .NET Framework tabanlı çok geniş çözümler sağlar ve ayrıca uygulama geliştirmeyi oldukça kolaylaştırır.

Bu çalışmada takip ettiğ imiz yaklaşım şu şekildedir: Birinci bölümde tez çalışması ile ilgili giriş yapıldıktan sonra, ikinci bölümde görüntü işleme hakkında gerekli altyapısal bilginin verilmesi ve görüntü işleme uygulamalarında kullanılan önemli tekniklerin açıklanmasına yer verilmiştir. Üçüncü bölümde, dalgacık dönüşümlerine ilişkin bilgiler verilmiş, dalgacık dönüşüm tekniklerinden uygulamada gerçekleştirilen Daubechies ve Lifting Scheme için açıklamalarda bulunulmuştur. Dördüncü bölümde, JPEG2000 standardının görüntü sıkıştırma teknolojisi, kodlama ve kod çözümlenme (encoding and decoding) işlemleri ile tanımlanarak gerekli açıklamalarda bulunulmuştur. Ayrıca, JPEG2000’in özellikleri ve hatalardan korunma yöntemlerine de yer verilmiştir. Beşinci bölümde, proje için uygulama geliştirmede platform olarak kullanılan son teknoloji, Visual Studio 2005 ile birlikte gelen yenilikler ve özellikleri hakkında bilgi verilmiş ve son bölümde de gerçekleştirilen uygulamanın tanıtımı ve özelliklerinin açıklanmasına yer verilmiştir.

2. GÖRÜNTÜ İŞLEME TEKNİKLERİ

Bilindiği gibi, sayısal görüntü görsel bilginin ayrı bir formda görüntülenmesi için bitlerin sonlu sayılar ile gösterildiği bir dizi gerçel ve karmaşık sayılardan ibarettir. Fiziksel görüntünün uygun sayısal forma dönüşümü, örnekleme ve orantılama (quantization) işlemlerinin gerçekleştirildiği analog ve sayısal dönüştürücü kullanılarak yapılır. Bu nedenle görüntüler resimler olarak düşünülebilir fakat en geniş anlamda bir görüntü her çeşit iki boyutlu veriyi ifade eder.

Görüntü işleme, görüntü üzerinde yapılan değişiklikleri ifade eder ve bilgisayarlar kullanılarak sayısal ortamda gerçekleştirilir. Sayısal görüntü işleme bir görüntünün ve özelliklerinin değiştirilmesi için kullanılan çok çeşitli teknikleri içerir. En basit düzeyde, görüntü işleme görüntüdeki piksellerin fiziksel olarak yerlerinin değiştirilmesi ile gerçekleştirilir.

Farklı görüntü işleme algoritmaları geniş alt sınıflara ayrılarak incelenebilir. Farklı görevler ve problemler için farklı algoritmalar mevcuttur ve bu görevleri şu şekilde ayırmak mümkündür;

Görüntü İyileştirme (Image Enhancement): Özel bir uygulama için daha uygun bir sonuç üretmek amacıyla görüntüde yapılan işlemler iyileştirme (enhancement) olarak adlandırılır. Örneğin; görüntüdeki bulanıklığın giderilmesi veya keskinleştirme, kenarların belirginleştirilmesi, görüntüdeki zıtlığın veya parlaklığın artırılması veya gürültünün kaldırılması gibi uygulamalardır.

Bu gruptaki teknikler nokta işlemleri olarak bilinir. Çünkü görüntüde piksel piksel değişiklik yapmak için nokta işlemleri kullanılır. Her piksel değeri bir önceki piksel'in değerine bağlı olan yeni bir piksel ile yer değiştirilir. En önemli nokta işlemleri thresholding, adaptive thresholding, contrast stretching ve histogram eşitleme olarak belirtilebilir.

Görüntü Tamiri (Restoration): Bilinen bir nedenden dolayı zarar görmüş olan görüntüde düzeltme işlemleri yapılabilir. Örneğin, düzgün hareketten dolayı oluşan bulanıklığın kaldırılması, optik bozulmaların kaldırılması ve periyodik etkilenmelerin kaldırılması gibi.

Görüntü Bölümleme: Bir görüntüdeki elementlerin veya nesnelerin gruplandırılması, sınıflandırılması bu kategoride gerçekleştirilir. Bölümleme bir görüntüdeki çizgileri, daireleri veya arabalar, yollar, binalar gibi belirli şekilleri ele alınıp incelenmesi için yapılan bir gruplandırma işlemidir.

Görüntü Sıkıştırma: Sayısal bir görüntüyü sıkıştırmak için ihtiyaç duyulan bellek miktarını azaltmak amacıyla yapılan işlemlerdir.

Görüntü Manipulasyonu (Image Manipulation): Görüntü analizi, desen tanıma veya görüntü üzerinde yapılan watermarking gibi işlemler bu grupta incelenir.

Görüntü işleme için çeşitli teknikler mevcuttur ve en önemlileri morfolojik ve sayısal filtreleme teknikleridir.

Morfolojik işlemler genellikle gürültüyü azaltmak, sınırları belirginleştirmek veya görüntüdeki iskeleti belirlemek gibi işlemler için kullanılır.

Görüntü işlemede kullanılan filtreleme teknikleri ya yüksek ya da alçak frekansların geçişlerini engellemek amacıyla kullanılan tekniklerdir. Yüksek frekansların geçişini engellemek görüntüdeki bulanıklık veya düzleştirme (smoothing) işlemlerini gerçekleştirmek için yapılır. Alçak frekans geçişlerinin engellenmesi ise genellikle görüntüdeki kenarların belirginleştirilmesi veya iyileştirme (enhancement) için yapılır. (Goceri ve Boyen 2006)

Filtreleme teknikleri ya frekans ya da uzaysal (spatial) ortamda gerçekleştirilir. Uzaysal (spatial) ortam görüntüdeki pikseller topluluğunu ifade eder ve spatial ortam metotları ise doğrudan bu pikseller üzerinde yapılan işlemleri belirtir.

Görüntü işleme için uzaysal maskeler kullanıldığında bu işlem uzaysal filtreleme (spatial filtering) ve maskeler ise uzaysal filtreler (spatial filters) olarak adlandırılır.

Uzaysal filtreler, Doğrusal (Linear) ve Doğrusal Olmayan (Non-Linear) Uzaysal Filtreler olarak şu şekilde belirtilir:

- Alçak geçiren filtreler
- Yüksek geçiren filtreler
- Bant-geçiş (band-pass) ve Notch filtreler
- Gaussian Smoothing
- Kenar Belirginleştirme Filtreleri
- Doğrusal Olmayan Filtreler

Görüntü işleme için frekans ortamında yapılan filtreleme Fourier Dönüşümü kullanılarak yapılır. Fourier Dönüşümü bir görüntüyü sinus ve cosinus parçalarına ayırmak için kullanılan önemli bir görüntü işleme aracıdır.

Frekans ortamındaki bu filtreleme teknikleri Hızlı Fourier Dönüşümü (Fast Fourier Transformation - FFT) ve Walsh Dönüşümü ve ayrıca Hugh Dönüşümü ile sağlanır.

Hızlı Fourier Dönüşümü, Ayrık Fourier Dönüşümü (Discrete Fourier Transformation - DFT) hesaplamasında kullanılan diğer bir yöntemdir. Diğer yöntemler ile aynı sonucu üretmesinin yanı sıra daha etkili ve hesaplama zamanı daha azdır.

Hızlı Walsh Dönüşümü, Hızlı Fourier Dönüşümü ile tamamen aynı bir dönüşümdür fakat bu dönüşüm işlemlerinde gerçel veriler yerine karmaşık (sinus ve cosinus) veriler kullanılır.

Hugh Dönüşümü, görüntüdeki düz çizgiler, daireler ve elips şekiller gibi basit şekillerin bulunmasında faydalı bir dönüşümdür bu nedenle genellikle şekli belirginleştirmek amacıyla kullanılır.

Fourier açılımındaki en büyük dezavantaj sadece frekans verisinin olması, zaman bilgisini içermemesidir. Bu nedenle bir görüntüdeki bütün frekanslar gösterilebilir fakat bu frekans değerlerinin ne zaman elde edildiği belirtilemez. Bu problemi çözmek için geçmiş yıllarda zaman ve frekans bilgisini aynı anda gösteren çeşitli çözümler üretilmiştir.

Bu probleme bulunan en son çözüm ise dalgacık dönüşümü tekniğinin kullanılmasıdır. Bu tekniğin arkasındaki fikir sinyali çeşitli bölümlere ayırmak ve ayrı ayrı analiz

etmektedir. Bir sinyalin bu şekilde analiz edilmesi ile o sinyaldeki frekansların ne zaman ve nerede oluştuğuna dair daha fazla bilgi alınabileceği açıktır. Dalgacık dönüşümü zaman ve frekans gösterimini destekler.

Dalgacık, bölümlene (decomposition) tekniklerine dayanan sinyal işlemeden türetilmiştir. Bu tekniğin gücü görüntü sıkıştırma üzerinedir. Bir sonraki bölümde bu teknikler hakkında daha fazla bilgi verilmektedir.

3. DALGACIKLAR

3.1 Giriş

Dalgacıklar, veri veya fonksiyonların gösteriminde kullanılan belirli matematiksel gereksinimleri karşılayan fonksiyonlardır.

Dalgacıklar şu iki ana sebepten dolayı kullanılır:

- Dalgacıklar bir sinyaldeki özelliklerin belirlenmesi için gerekli olan zaman ve frekans verilerini içerir.
- Dalgacıklar çoklu-çözünürlük analizine dayanır. Bir dalgacık bölümlenme farklı çözünürlük düzeylerinde bir sinyalin analiz edilmesine imkan tanır.

Görüntü sıkıştırma için Ayrık Dalgacık Dönüşümü (Discrete Wavelet Transformation - DWT) kullanılır bu nedenle Sürekli Dalgacık Dönüşümü (Continues Wavelet Transformation - CWT) ve Ayrık Dalgacık Dönüşümü'nün açıklanmasında yarar vardır.

Sürekli dalgacık dönüşümü, analiz için kullanılan ölçeğin değiştirilerek zamana bağlı olarak analiz penceresinin sinyal üzerinde kaydırılıp çarpma işleminin yapılması ve bu işlemler sonucunda elde edilen verilerin toplanması ile hesaplanır. (WEB_2 2006)

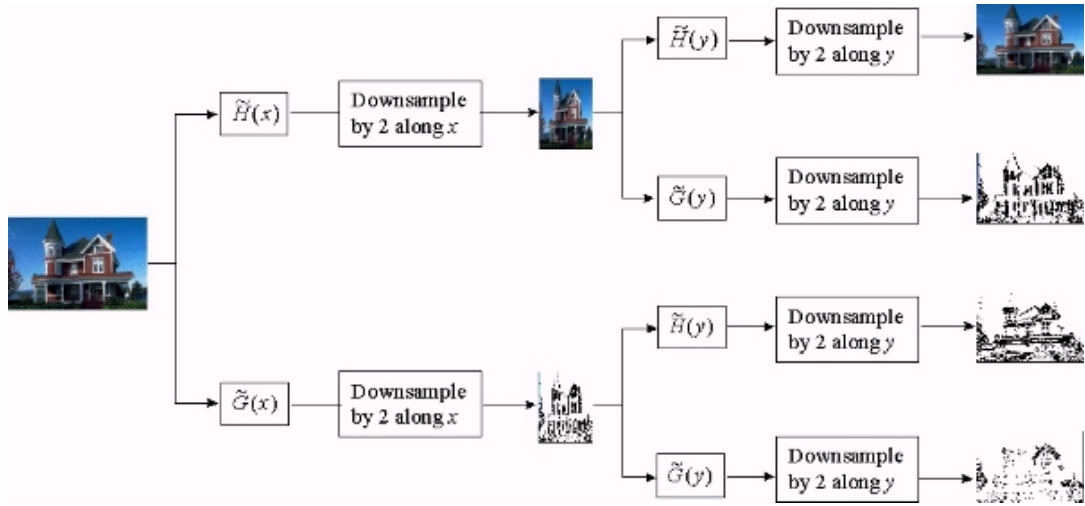
Ayrık Dalgacık Dönüşümünde sinyali analiz etmek için farklı frekanslar kullanılır. Sinyal yüksek frekansı analiz etmek için bir dizi yüksek geçirgen filtreden geçirilir, sonra alçak frekansı incelemek için bir dizi alçak geçirgen filtreden geçirilir. (Goceri ve Boyen 2005)

Görüntüler iki boyutlu veriler olduğu için görüntü sıkıştırma işleminde 2D Ayrık Dalgacık Dönüşümü (DWT) kullanılır. 2D Ayrık Dalgacık Dönüşümünde veri ilk önce satırların sonra da sütunların dönüşümü yapılabilecek şekilde ayrılabilir bir dönüşümdür. (WEB_3 2006)

Dalgacık dönüşümlerinin uygulanması ileri yönlü veya geri yönlü olabilir. Bu ileri yönlü ve ters yönlü dönüşümler bir çift Dörtlü Ayna Filtresi (Quadrature Mirror Filters-QMFs) ile uygulanabilir. Bu nedenle dalgacık tabanlı görüntü sıkıştırma altbant

(subband) kodlama olarak görülebilir. Her QMF çifti bir alçak geçiren filtre (H) ve sinyal genişliğini ikiye bölen bir yüksek geçiren filtreden (G) ibarettir.

Şekil 3.1 bir görüntünün iki ayrı, bir boyutlu (1D) dönüşüm ile gerçekleştirilen, iki boyutlu (2D) ileri yönlü dalgacık dönüşümünü göstermektedir. Görüntü $f(x,y)$ ilk önce x boyutu üzerinde filtrelenir. Böylece alçak geçiren görüntü $fL(x,y)$ ve yüksek geçiren görüntü $fH(x,y)$ elde edilir. X boyutunca fL ve fH 'in band genişliği f 'in yarısına indiği için filtrelenen her görüntü üzerinde veri kaybı olmadan aşağı örnekleme (downsample) yapılabilir. Aşağı örnekleme diğer filtrelenen değerlerin iptal edilmesi ile yapılır. Sonraki adım ise hem fL hem de fH 'nin y boyutunca filtrelenmesidir. Bu işlem sonunda dört görüntü elde edilir: fLL , fLH , fHL , fHH . Bu defa y boyutunca aşağı örnekleme yapılır. Yukardaki şekilde gösterildiği gibi 2D filtreleme yöntemi bir görüntüyü bir ortalama fLL (average) sinyal ve üç detaylı sinyale: fLH dikey görüntü özelliklerini, fHL yatay özellikleri ve fHH köşegen özelliklerini vurgulayacak şekilde ayırır. Detaylı sinyallerin boyutsal duyarlılığı bu sinyallerin içerdiği frekans aralığının kişilerce düzenlenmiş şeklidir.



Şekil 3.1: İleri yönlü dalgacık dönüşümü

İleri dalgacık dönüşümü tamamlandıktan sonra her ölçekteki ortalama ve detaylı sinyalleri oluşturan matris katsayıları elde edilir. Henüz görüntü üzerinde herhangi bir sıkıştırma yapılmamıştır; İleri yönlü dalgacık dönüşümünün her uygulaması katsayıların magnitute değerlerinin büyümesine neden olur, bu nedenle görüntü için gerekli olan

depolama alanında bir artış gözlenir. Sıkıştırma, orantılama (quantization) ve dalgacık katsayılarının kodlanması ile gerçekleştirilmektedir.

İki boyutlu (2D) ters dalgacık dönüşümünü gerçekleştirmek için ortalama ve detaylı sinyaller üzerinde ilk önce y boyutunda yukarı örnekleme (upsampling) yapılır. Yukarı örnekleme y boyutundaki her çift değere sıfır verilerek gerçekleştirilir. Yukarı örnekleme sinyallerin tekrar biraraya getirilmesinde gerekli olan bant genişliğini korumak için yapılır.

Yukarı örneklemeden sonra sinyaller y boyutunca filtrelenir ve eklenir. Bu işlem daha orjinal görüntüyü elde etmek için sonra x boyutu için tekrarlanır. (Michael vd 1994)

Daha fazla boyutlu görüntüler söz konusu ise dalgacık dönüşümü ayrı ayrı her boyutta gerçekleştirilir. Çünkü dalgacık dönüşümü, matris değerlerinin birleşiminden dolayı dalgacık dönüşüm matris değerlerinin çarpımı şeklinde yazılabilir. (WEB_5 2001)

Gerçekleştirilen uygulamada diğerlerinden daha gelişmiş ve daha iyi frekans özellikleri olan Daubechies dalgacık dönüşüm algoritmaları kullanılmıştır. Aşağıda Daubechies dalgacık dönüşüm tiplerinden biri olan D4 için ölçekleme fonksiyon katsayı değerleri gösterilmektedir.

$$h_0 = \frac{1 + \sqrt{3}}{4\sqrt{2}} \quad h_1 = \frac{3 + \sqrt{3}}{4\sqrt{2}} \quad h_2 = \frac{3 - \sqrt{3}}{4\sqrt{2}} \quad h_3 = \frac{1 - \sqrt{3}}{4\sqrt{2}} \quad (3.1)$$

D4 için dalgacık fonksiyon katsayı değerleri ise şu şekildedir:

$$g_0 = h_3; \quad g_1 = -h_2; \quad g_2 = h_1; \quad g_3 = -h_0 \quad (3.2)$$

Daubechies D4 ölçekleme fonksiyonu;

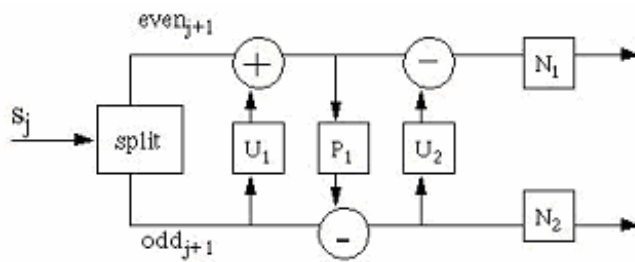
$$\begin{aligned} a_1 &= h_0 s_{2i} + h_1 s_{2i+1} + h_2 s_{2i+2} + h_3 s_{2i+3} \\ a[i] &= h_0 s[2i] + h_1 s[2i+1] + h_2 s[2i+2] + h_3 s[2i+3]; \end{aligned} \quad (3.3)$$

Daubechies D4 dalgacık fonksiyonu;

$$\begin{aligned} c_1 &= g_0 s_{2i} + g_1 s_{2i+1} + g_2 s_{2i+2} + g_3 s_{2i+3} \\ a[i] &= g_0 s[2i] + g_1 s[2i+1] + g_2 s[2i+2] + g_3 s[2i+3]; \end{aligned} \quad (3.4)$$

Dalgacık Lifting Scheme çeşitli avantajları olan diğer bir algoritma olup daha etkin olmakla birlikte D4 versiyonundaki gibi geçici bir diziye ihtiyaç duymaz. Aşağıdaki şekilde belirtildiği gibi ters dönüşüm ile ileri dönüşüm arasındaki fark ekleme işlemleri yerine çıkarma işlemlerinin yapılmasıdır. (Walker 1999)

Lifting Scheme, Daubechies'nin D4 dönüşümünün farklı bir versiyonudur. Lifting Scheme dalgacık dönüşümleri güncelleme (update) ve tahmin (predict) aşamalarından oluşur. Bu durumda bir normalizasyon adımı eklenmiştir. Bir ileri dönüşüm adımı Şekil 3.2'de gösterilmektedir.



Şekil 3.2: Daubechies'nin D4 ileri yönlü dalgacık dönüşümü

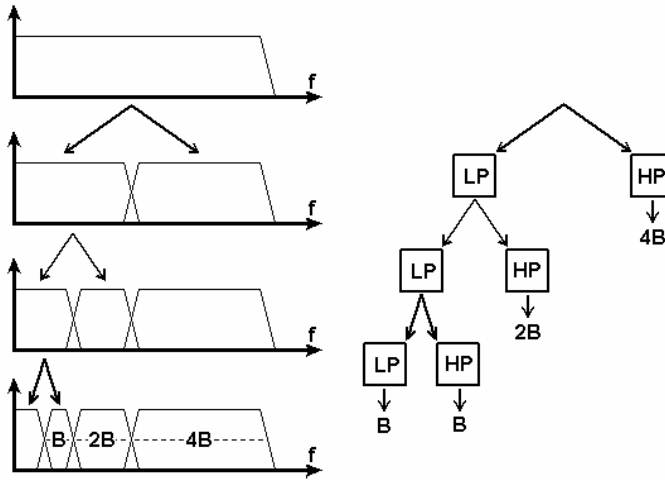
3.2 Altbant (Subband) Kodlama

Eğer dalgacık dönüşümü filtre bankası olarak düşünülürse, sinyallerde bu filtre bankasından geçen sinyaller olarak kabul edilir. Farklı filtreleme aşamalarının çıktıları, dalgacık ve ölçkleme fonksiyon dönüşüm katsayılarıdır. Bir sinyalin filtre bankasından geçirilerek analiz edilmesi fikri yeni bir fikir olmayıp, örneğin bilgisayar görüntü uygulamalarında kullanılan eski bir yöntemdir.

Şekil 3.3'de bir alçak geçiren bant ve çift bant genişliği ile bir dizi bant-geçiş ayrılması işlemi tekrarlı tayf (spectrum) sonrası görülmektedir.

Altbant (subband) kodlamada gereksinim duyulan filtre bankası çeşitli şekillerde gerçekleştirilir. Çeşitli bant-geçiş (band-pass) filtrelerinin oluşturulması için bir yöntem ise, tayfi (spectrum) frekans bantlarına ayırmaktır. Avantajı, her bant genişliğinin serbestçe seçilmesidir. Bu durumda analiz edilecek olan sinyalin tayfi ilgi duyulan alan

üzerine örtülür. Dezavantajı ise her filtrenin ayrı ayrı tasarımılanması gerektiğidir ve bu zaman tüketen bir işlem olabilmektedir.



Şekil 3.3: Altbant kodlama

Diğer bir yöntem ise, sinyal tayfı alçak geçiren ve yüksek geçiren parçalar olmak üzere iki eşit parçaya bölmektir. Yüksek geçiren bölüm ilgilenilen en küçük detayları içerir. Sonuçta, iki bölüme sahip olmakla birlikte alçak geçiren bölüm hala bazı detayları içermektedir ve bu nedenle onu tekrar bölümleyebiliriz. Bu yöntem ile bir türetilmiş filtre bankası oluşturulur. Genellikle bandların sayısı veri miktarı veya mevcut olan hesaplama gücü ile sınırlıdır. Tayfı (spectrum) bölümlenme işlemi grafiksel olarak yukarıdaki şekilde görüntülenmektedir. Bu şemanın avantajı sadece iki filtre tasarlamak zorunda olmamız, dezavantajı ise sinyal tayf (spectrum) zaman değerinin sabit olmasıdır.

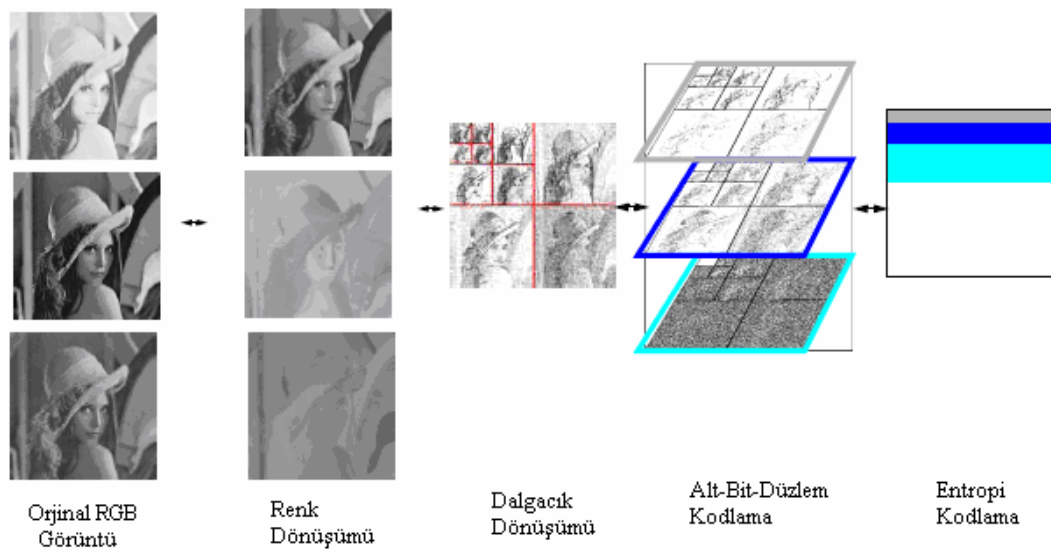
Teoride ilk bölümün bize yüksek geçirgenlik ve alçak geçirgenlik vermesine rağmen, gerçekte sinyalin sınırlı bant genişliğinden dolayı yüksek geçirgenlik bantı bant-geçiş bantıdır. Başka bir deyişle, aynı altbant (subbant) sinyali her biri bant genişliğinin iki katı kadar genişliğe sahip olan bant geçiş filtrelerinin bankasını besleyerek yapabilmektedir. Bu bölümün başında, bunun dalgacık dönüşümünün bu siyale uygulanması ile aynı olduğunu belirtilmiştir. Dalgacıklar çift bant genişliği ile bant-geçiş bantları verir ve ölçekleme fonksiyonu alçak geçirgenlik ile bizi destekler. (Marcellin vd 2000) Bu noktadan, bir dalgacık dönüşümünün altbant (subband) kodlama şeması ile aynı şey olduğu sonucu elde edebilmektedir.

Bir sonraki bölümde, dalgacık dönüşümleri kullanılarak geliştirilen JPEG2000 standardının özellikleri ve kodlama aşamaları açıklanmaktadır.

4. JPEG2000

4.1 JPEG2000 Nedir?

JPEG2000 Uluslararası Standartlar Organizasyonu (ISO)'nun bir bölümü olan Joint Photographic Expert Grup (JPEG) tarafından geliştirilen görüntü sıkıştırma standardıdır. Şekil 4.1'de JPEG2000 Standardı için sıkıştırma yöntemi gösterilmektedir.



Şekil 4.1: JPEG2000 Standardının sıkıştırılma yöntemi

JPEG2000 taşınabilir sayısal kameralardan gelişmiş ön-baskı, tıbbi görüntüleme ve diğer anahtar sektörler kadar çok geniş bir kullanım alanına sahiptir. Orjinal JPEG görüntü sıkıştırma standardı internet, sayısal kameralar, yazıcı ve tarayıcı cihazlarında içeren çok farklı uygulama alanlarından geniş kabul almıştır. Görüntü işleme modern çoklu-media iletişimlerinde merkezi rol oynar ve sıkıştırılmış görüntüler internet trafiğini azaltır.

JPEG2000 dalgacık teknolojisine dayalı “state-of-the-art” sıkıştırma tekniklerini kullanan bir çeşit görüntü kodlama tekniğine sahiptir.

JPEG2000 ile çeşitli çözünürlükleri, kaliteleri, parçaları veya uzaysal bölgeleri sıkıştırılmış dosyayı açmadan görüntülemek mümkündür.

JPEG2000 “state-of-the-art” sıkıştırma tekniğini sağlamak için dalgacık dönüşümü ve bit-düzlem entropi (entropy) kodlayıcıyı kullanır. (Athanasios vd 2001)

4.2. Niçin Görüntüleri Sıkıştırmaya Gereksinim Duyulur?

Görüntülerin çoğunlukla sıkıştırılmasına gereksinim duyulmaktadır. Bunu açıklamak için farklı tiplerdeki sıkıştırılmamış görüntüler için depolama büyüklüğünü, veri iletim bant genişliğini, ve çeşitli tiplerdeki sıkıştırılmamış görüntüler için gerekli olan iletim zamanını gösteren aşağıdaki Tablo 4.1’e bakalım. Tablo 4.1’deki değerlerden anlaşılıyor ki, görüntüler fazla depolama alanı, geniş veri iletim bant genişliği, ve uzun iletim zamanına ihtiyaç duyar. Günümüzdeki teknoloji ile bu probleme tek çözüm, görüntünün depolanması ve iletiminden önce sıkıştırılmasıdır. Alıcı tarafta ise sıkıştırılan görüntü sıkıştırılmamış hale getirilebilir. (David vd 2002)

Tablo 4.1: Sıkıştırılmamış görüntüler

Görüntü Tipi	Boyutları	Bits/Pixel	Sıkıştırılmamış Büyüklük	İletim Bant Genişliği	İletim Zamanı (28.8K modem ile)
Gri Tonlu	512 x 512	8 bpp	262 KBytes	2.1 Mbit/image	1 min 13 sec
Renkli	512 x 512	24 bpp	786 KBytes	6.29 Mbit/image	3 min 39 sec
Tıbbi	2048 x 1680	12 bpp	5.16 MBytes	41.3 Mbit/image	23 min 54 sec
Yüksek Yoğunluklu	2048 x 2048	24 bpp	12.58 MBytes	100 Mbit/image	58 min 15 sec

4.3 Niçin Başka Bir Standart?

JPEG yıllardır kullanılan bir standart olmakla birlikte artık günümüzün gelişen gereksinimlerini karşılayamamaktadır. Bugünün sayısal görüntülemesi sadece kalite açısından değil aynı zamanda görüntü büyüklüğü bakımındanda tamamen isteğe bağlıdır. (Frey ve Suesstrunk 2000, Parulski ve Rabbani 2000, Boliek vd 2000)

Teknoloji geliştikçe JPEG günümüzdeki ihtiyaçları karşılayamaz hale gelmektedir. JPEG için uygulama alanının genişliği bazı karışıklıkları da beraberinde getirmektedir.

Bu standard üzerinde bir takım iyileştirmeler yapılsada köklü bir değişiklik yapılmadıkça ilerleme sağlanamayacağı açıktır.

Ricoh Innovations firması tarafından CREW algoritmasının JPEG-LS içinde kullanmak üzere kabul edilmesinden sonra 1996 yılında JPEG2000 çalışma grubu kuruldu.

JPEG2000 grubu tarafından oluşturulması beklenen standart mevcut standardın hatalarını, eksikliklerini şu şekilde belirtmiştir:

Az sayıdaki bitlerin sıkıştırılmasındaki yetersizlik: JPEG orta ve yüksek düzeydeki bit oranları için kabul edilebilir bir sonuç verirken, düşük düzeydeki bit oranları için yeterli olmamaktadır. Yeni oluşturulacak olan standart mevcut standarttan daha iyi bir performans vermelidir. (WEB_1 2006)

Kayıplı ve kayıpsız sıkıştırma: Mevcut standart ile hem kayıplı hem kayıpsız sıkıştırma tek bir kod dizininde gerçekleştirilememektedir. Kayıpsız sıkıştırmanın gerekli olduğu alanlardan birisi örneğin tıbbi görüntülemelerdir ki kesinlikle veri kaybının olması istenmez. Bu nedenle yeni oluşturulacak olan standarttaki bit dizini hem kayıplı hem kayıpsız sıkıştırmayı desteklemelidir.

Büyük boyutlarda görüntüleme: JPEG, 64Kx64K'dan daha büyük boyutlardaki görüntülerin bölümlenme (tiling) işlemi yapılmadan sıkıştırılmasına izin vermez.

Tek sıkıştırma mimarisi: JPEG standardı 44 farklı tipe sahip ve bunların pek çoğu uygulamaya özgü olup her kod çözümleyicisi ile görüntülenememektedir. Farklı görüntü tipleri ve uygulamalar için sistemin iyileştirilebilmesi amacıyla açık mimariye izin verilmesi istenmektedir.

Gürültülü ortamlarda iletim: JPEG kablosuz iletişim bu kadar yaygın olmadan önce oluşturulmuştur bu nedenle kabul edilebilir bir düzeyde hata yakalama özelliğine sahip değildir. Kablosuz iletişim kanallarında veri iletiminin sağlamlığı talep edilmekte ve görüntü kalitesi için büyük önem taşımaktadır.

Bilgisayarlı görüntüleme: JPEG doğal görüntüleme için iyileştirilmiş olup bilgisayarlarda üretilen görüntüler için iyi sonuç vermemektedir.

Bütünleşik belgeler: JPEG metin görüntülemeye uygulandığında düşük bir sonuç vermektedir. Hem iki seviyeli (bi-level) hem de sürekli (continues-tone) görüntülerin sıkıştırılmasında bir kodlama standardına ihtiyaç duyulur.

Görüntü ve metin ile bütünleşik belgeleri kullanan uygulama örnekleri tıbbi görüntüler, grafik ve bilgisayarda üretilen görüntülerdir.

4.4 JPEG2000 ile Hedeflenenler

JPEG2000 çalışma grubunun amacı aşağıdaki şartları sağlayacak yeni bir görüntüleme standardı geliştirmektir:

- farklı tip ve biçimdeki görüntüleri destekleyen (iki-seviyeli, gri-tonlu, renkli, çok parçalı)
- farklı tipik özellikteki görüntüleri destekleyen (doğal, bilimsel, uzaktan algılamalı, bütünleşik gibi)
- farklı görüntüleme modellerine izin veren (istemci/sunucu, gerçek zamanlı iletim, sınırlı depolama ve bant genişliği kaynakları olan)

Bu kodlama sistemi düşük bit oranındaki uygulamalar için düşünülmüş olup oran dağılım ve özel görüntü kalite performansının mevcut standarttan daha iyi olduğu gösterilecektir.

4.5 JPEG2000'nin Bölümleri

JPEG2000, standardın bütün parçalarını ifade eder.

Bu standardın birinci bölümü (çekirdek bölüm) uluslararası standart olarak yayınlanmış olup diğer beş bölüm (2-6) tamamlanmak üzere ve diğer dört yeni bölüm (8-11) ise geliştirilme aşamasındadır.

4.5.1 Bölüm-1: Çekirdek kodlama sistemi

İsmindende anlaşılacağı gibi birinci bölüm JPEG2000'in çekirdeğini tanımlar. Bu bölüm JPEG2000 kod dizini yapısını ve görüntülerin kodlanması için gerekli adımları içerir. Standardın diğer bölümleri dosya uzantıları ve diğer çeşitleri ile ilgili olup mevcut uygulamalarda kullanılan bölüm sadece bu ilk bölümdür.

Bu bölüm, aynı zamanda JP2 olarak bilinen basit dosya yapısını tanımlar. JP2, JPEG2000 ailesindeki diğer dosya formatları ile paylaşılan genişletilmiş bir mimari kullanır.

Bu bölüm ayrıca örnekleri, teknik referansları ve patenti ISO tarafından alınan şirketlerin listesini içerir. JPEG2000 geliştirilirken dikkate alınan bir diğer unsurda lisans ödemesi yapılmadan Bölüm-1'in kodlanmasının gerçekleştirilmesidir, fakat JPEG grubu bunun için herhangi bir resmi garanti veremez.

4.5.2 Bölüm-2: Uzantılar

Bu bölüm Bölüm-1 için bazı uygulamalarda yararlı olabilecek çeşitli uzantıları tanımlar. Bu bölüm;

- katsayı miktarlarını belirten değerleri ve dalgacık bölümlene formlarının daha esnek olması,
- ilgi bölgesinin kodlanması için bir alternatif yöntem,
- JP2 tabanlı olan fakat katmanları, hareketli görüntüleri ve genişletilmiş renk uzayını ve daha fazlasını destekleyen yeni bir dosya biçimi; JPX tanımlaması,
- fotoğraf görüntüleme için zengin veri kümesi (DIG35 özelliklerine dayalı) gibi özellikleri içerir.

Bölüm-2'de yer alan uzantıların çoğu birbirleri üzerinde bağımsız olarak işlem yapabilirler. Bunun için mekanizmalar hem kod dizini hem de JPX dosya biçim düzeyinde sağlanmaktadır. Standardın diğer bölümü hareketli JPEG2000 ile ilgili olup uzantıları yine bu bölümde tanımlanmaktadır.

4.5.3 Bölüm-3: Hareketli JPEG2000

Çerçeve çerçeve JPEG2000 MPEG tipindeki dosyalar kadar etkili olamamakla birlikte MPEG-4 ve Quicktime ile karşılaştırıldığında oldukça yüksek kaliteli güncelleme standardıdır.

Bölüm-3 MJ2 (veya MJP2) olarak adlandırılan dosya biçimini tanımlar. MJ2 içerçeve kodlamayı içermez, her çerçeve JPEG2000 kullanılarak bağımsız olarak kodlanır. Beklenen uygulamalar ise,

- Sayısal kameralar ile çekilen video görüntülerin depolanması,
- Yüksek kaliteli çerçeve tabanlı görüntü kaydetme ve güncelleme
- Sayısal sinema
- Tıbbi ve uydu görüntüleme işlemleri olarak belirtilebilir.

4.5.4 Bölüm-4: Uyumluluk

Uyumluluk olarak adlandırılan bu bölümünde kod çözümlemede kullanılan üç sınıf tanımlanır. Çünkü, kodlama yapan kişiler (encoders) kod çözümleme işleminde (decoding) bu standartta nelere gerek duyulduğunu bilmek için standardı incelemek isteyeceklerdir.

Bu bölüm, ilk bölümdeki kodlama ve kod çözümleme işlemleri için test yordamlarının tanımlandığı uyumluluk ile ilgilidir. Bu bölümdeki test dosyaları hem çıplak kod dizinini hem de JP2 dosyaları içerir.

4.5.5 Bölüm-5: Referans yazılımı

Bu bölümde amaç oluşturulacak olan standardın kullanımının kolaylaştırılmasıdır.

Bu bölüm bir kısa metin belgesi ve ilk bölümün kodlanması için gereken iki kaynak kod paketinden ibarettir. Bu iki kod paketlerinden biri C programlama dili diğeri ise Java ile yazılmış olup her ikisinde açık kaynak kodludur.

4.5.6 Bölüm-6: JPM

Bu bölüm belge görüntülemek için Karışık Raster İçerik (Mixed Raster Content) kullanan JPM dosya formatını tanımlar.

JPM her sayfada çeşitli nesnelere olan çok sayfalı belgeleri depolayabilir. JPM, JPEG2000 ailesinin bir üyesi olmasına rağmen diğer kodlama ve sıkıştırma teknolojilerininide sağlar.

4.5.7 Bölüm – 7

Bu bölüm geliştirilmemiş, iptal edilmiştir.

4.5.8 Bölüm-8: JPSEC (Güvenlikli JPEG2000)

Bölümü bir görüntünün çeşitli bölümlerini güvenli hale getirmek için yapılan işlemleri içerir.

Sayısal ortamda yapılan çalışmalar kolaylıkla kopyalanıp, yasal veya yasal olmayan yollardan üzerinde bir takım işlemler yapılabilir. Güvenlik bu nedenle görüntüleme uygulamalarında önemli bir özelliktir.

Aşağıdaki örnekler uygulamalardaki JPSEC kullanımını gerektirebilecek durumları belirtmektedir:

- Kaynak kullanım izni: JPSEC, herhangi bir kaynağın belirlenmesi ve kullanım için gereken onaylanma işlemine izin verir.
- Veri bütünlüğü: JPSEC bütünlük doğrulamaya izin verir.
- Koşullu erişim: JPSEC bir görüntüdeki koşullu erişime izin verir. Böylelikle bir kullanıcı görüntüdeki izin verilen alanı yüksek çözünürlüğe ihtiyaç duymadan düşük çözünürlük ile de görebilir.

- İsim hakkı koruması: JPSEC, görüntülerin isim haklarının (copyright) korunmasına izin verir ve bu sahiplik tanımlanma mekanizmasını içerir.

İçeriği korumak için bu teknikler sayısal imzalama, şifreleme, anahtar üretim ve yönetimini içerir.

4.5.9 Bölüm-9: JPIP (Etkileşimli araçlar, API'ler ve protokoller)

Bu bölüm için çalışma grubu tarafından üzerinde en çok durulan, en önemli bölüm olduğu söylenilebilir. JPEG2000 standardı rastgele erişim ve kod diziminin tekrar düzenlenmesi gibi esneklikler sağlar.

Bu bölümün temel parçası JPIP olarak adlandırılan istemci/sunucu protokolüdür. JPIP kodlaması HTTP'nin üst seviyesinde gerçekleştirilmektedir. JPIP sunucu tarafından geri döndürülen görüntü verisi için çeşitli dosya biçimlerini ele alır.

Bu bölüm, ilk bölümdeki teknolojinin uygulamasına odaklanmış olmasına rağmen, JP2'ya ek olarak ikinci bölümdeki bazı dosya biçimlendirme uzantılarını destekler. Ayrıca JPEG2000 dosyalarını indekslemek için bazı yeni dosya formatları bu bölümde tanımlanır.

4.5.10 Bölüm-10: 3D ve kayan ondalıklı veri

Bu bölüm, üç boyutlu verilerin kodlanması ile ilgilidir.

4.5.11 Bölüm-11: JPWL (Kablosuz)

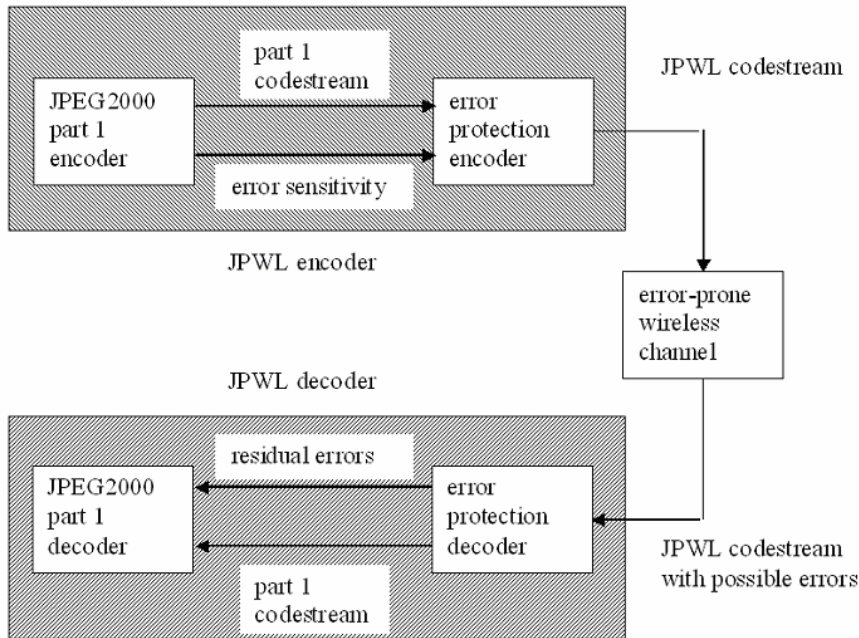
JPEG2000 görüntülerinin kablosuz uygulamalarda iletimin hatalara karşı dayanıklı olması için geliştirilmiştir. Bölüm-1'de tanımlanan çekirdek kodlama sistemi performansı arttırmak için gürültülü kanallarda hata yakalama araçlarını içerir. Fakat bu araçlar sadece hataları tespit edebilmektedir, iletim hatalarının düzeltilmesi durumu söz konusu değildir. Ayrıca bu araçlar kod dizisinin en önemli bölümü olan görüntü dosyasının başlık bölümüne uygulanamamaktadır. Bu nedenlerden dolayı kablosuz iletimlerde etkili sonuç vermemektedir.

Bu sorunu çözmek, kablosuz ortamlarda görüntü iletimini etkinleştirmek için JPEG2000 Kablosuz veya Bölüm-11 olarak adlandırılan bu bölüm geliştirilmiş ve çekirdek kodlama sistemi olarak adlandırılan ilk bölümdeki elementler kullanılmıştır.

JPWL şu üç fonksiyonel özelliği destekler:

- iletim hatalarına karşı kod dizininin tanımlanması,
- hata iletiminde kod dizinindeki farklı parçaların duyarlılık derecelerinin tanımlanması,
- kod dizinindeki hataların bulunduğu yerlerin tanımlanması.

Şekil 4.2 JPEG2000 için kodlama ve kod çözümü aşamalarını göstermektedir.



Şekil 4.2: JPEG2000 için kodlama ve kod çözümü aşamaları

Hatalardan korunmak için kullanılan kodlayıcı (encoder) ve kod çözümleyici (decoder) iletim hatalarına karşı bir kod dizinini korumayı amaçlayan tekniklerden ibarettir. Hatalardan korunmak için kullanılan kod dizininde fazlalık veya ek veriler eklenerek düzenlemeler yapılır. Özellikle, kod dizininin en kritik parçası olan başlık bölümünde hatalardan korunmak için yedek alınır. Hatalardan korunmak için yapılan kodlamanın çözümlenmesi işleminde hataların sıklığı tespit edilip, düzeltilir. Hata duyarlılık alıcıları hata iletiminde farklı kod dizin parçalarının duyarlılık derecesini

tanımlar. Bu bilgi JPEG2000'in birinci bölümü kullanılarak üretilip görüntüdeki hatalardan kaçınmak için kullanılır.

Artan hata tanımlayıcıları kod dizinindeki artan hataların yerlerini tanımlamada kullanılır.

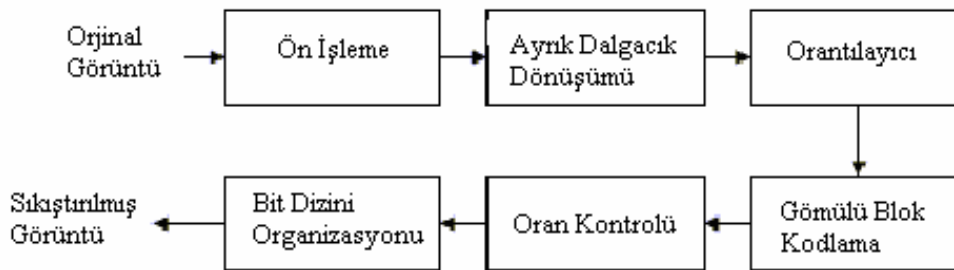
Hata düzeltme amaçlı kullanılan kod çözümleyicileri ile düzeltilemeyen hatalar artan hatalar olarak adlandırılır. Bu bilgi tipik olarak hatalardan korunmak için yapılan kod çözümleme işlemi sırasında elde edilir ve genellikle ilk bölümdeki kod çözümleyici tarafından bozulmuş kodların düzeltilmesinde kullanılır.

4.5.12 Bölüm-12: ISO tabanlı dosya biçimi

Bu bölüm, Bölüm-3'deki hareketli JPEG2000 dosya biçimini MPEG-4'den alır. Geliştirilme aşamasında bazı parçalar hem MPEG-4 hem de JPEG2000 içinde olmak üzere iki defa tanımlanmıştır. Bu nedenle bazı yerleri basitleştirmek için ortak bölüm her iki parçadan da kaldırılır ve tek bir döküman olarak her iki bölümde de tekrar işlenir.

4.6 Kodlama (Encoding)

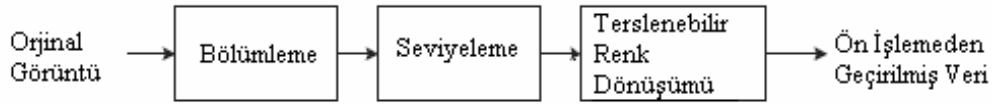
The JPEG2000 standardı hem kayıpsız hem kayıplı sıkıştırmaya olanak sağlar. JPEG2000 sıkıştırma standardı aşağıdaki Şekil 4.3'de gösterilen aşamaları içermektedir. (Athanasios vd 2000)



Şekil 4.3: JPEG2000'in kodlama aşamaları

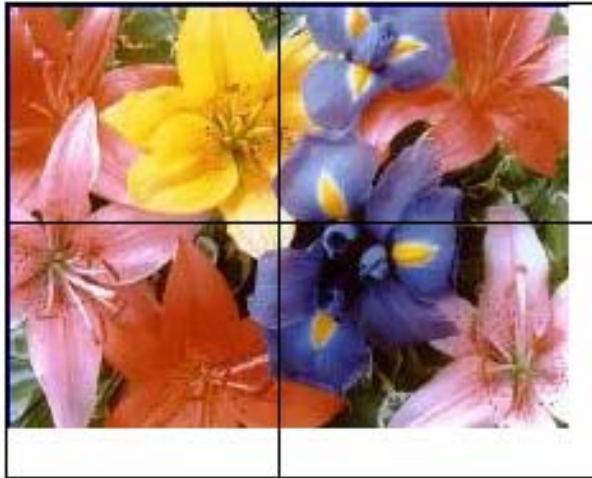
4.6.1 Ön işleme (Pre-Processing)

İlk aşama olarak ön işleme gerçekleştirilir. Ön işleme Şekil 4.4’de gösterildiği gibi gerçekte üç alt aşamayı kapsar. Bu aşamaların doğru bir şekilde gerçekleştirilmesi ayrık dalgacık dönüşümünün kusursuz bir şekilde yapılmasını sağlar.



Şekil 4.4: Ön işleme aşamaları

Kodlayıcı için gerekli bellek miktarı kodlanmış olan görüntünün boyutundan daha geniş olabilir. Bu sorunu çözmek için, JPEG2000 seçenekli bölümlenmeye (tiling) olanak sağlar. Bölümlenmede girdi olarak kullanılan görüntü Şekil 4.5’deki gibi eşitgen eşit parçalara ayrılır.



Şekil 4.5: Bölümlenme

Dalgacık dönüşümü, orantılama, entropi kodlama gibi bütün işlemler görüntü parçaları üzerinde bağımsız olarak gerçekleştirilir. Her parça bağımsız olarak sıkıştırılır.

Bölümlenme işlemi görüntü kalitesini etkiler. Bununla ilgili veriler Tablo 4.2’de verilmiştir.

Tablo 4.2: Bölümlemenin görüntü kalitesi üzerindeki etkisi

Görüntü Kalitesinin Bölümleme Üzerindeki Etkisi			
Bölümleme	Bölümlemesiz	Bölümlerin	Bölümlerin
Bit Oranı(b/p)		Boyutu	Boyutu
		128x128	64x64
0.125	24.75	23.42	20.07
0.25	26.49	25.69	23.95
0.5	28.27	27.79	26.80

4.6.2 Renk dönüşümü

Renk Dönüşümü sıkıştırma etkisini artırır ve görsel olarak orantılamaya izin verir. Renk dönüşümü ile ifade edilen, kodlama etkinliğinin artırılması için RGB verilerinin YCrCb (Y: Luminance, Cr: Chrominance Kırmızı, Cb: Chrominance Mavi statik olarak birbirlerine daha az bağlı ve dolayısı ile sıkıştırma daha iyi olmaktadır.) verilerine dönüştürülmesidir.

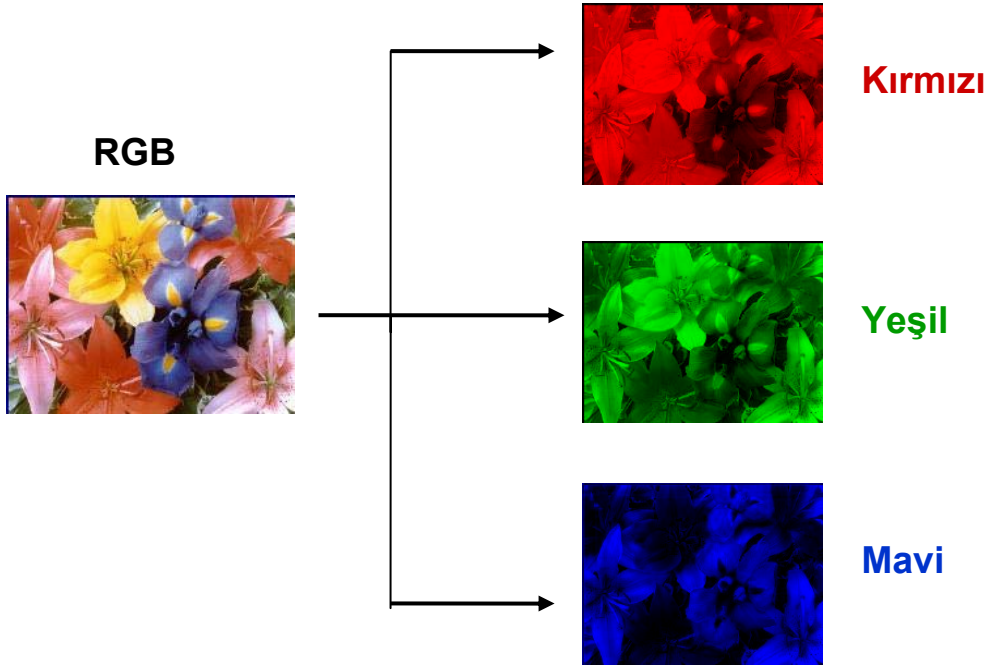
JPEG2000 sıkıştırma genellikle renkli görüntüleri sıkıştırmak için kullanılmaktadır. Renkli görüntüler genellikle RGB biçiminde olmaktadır.

RGB biçimindeki bir görüntüde kırmızı, yeşil ve mavi olmak üzere üç parçanın birleşimi yer alır. Ayrık dalgacık dönüşümünün JPEG2000’da yapılması için her katman ayrı ayrı dönüştürülür.

Fakat Y, Cr ve Cb renk değerleri R, G ve B renk değerlerine göre statik olarak birbirlerine daha az bağlı olması nedeniyle bağımsız olarak yapılan sıkıştırma daha etkili olmaktadır. Bu nedenle JPEG2000’da RGB değerlerinin YCrCb değerlerine dönüştürülmesi, yani renk dönüşüm işlemi gerçekleştirilir.

Bağımsız renk dönüşüm işlemleri sıkıştırma performansını arttırmaktadır. Eğer görüntü sadece gri renklerden oluşuyor ise renk düzlemi ayrı olarak depolanır.

Şekil 4.7’de orjinal resimde yer alan renklerden kırmızı, yeşil ve mavi renkleri için renk dönüşümü gösterilmektedir.

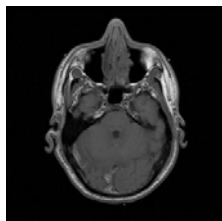


Şekil 4.7: Renk Dönüşümü

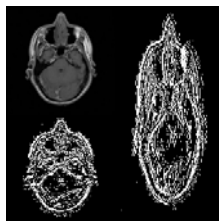
4.6.3 Dalgacık dönüşümü

Dalgacık dönüşümü entropi kodlama işleminden önce bölümlere uygulanır. Dalgacık dönüşüm işleminin faydası dönüştürülen verinin genellikle daha düşük entropi sergilemesi ve bundan dolayı daha çok sıkıştırılabilir olmasıdır. Özellikle, dalgacık dönüşümü, bir bölümü dört alt-bant olarak ayırdığı için kaynak modelleme her alt-bant için düşünülebilir ki JPEG2000 ile yapılan işlem budur. (WEB_3 2006)

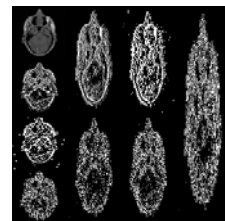
Dalgacık filtreleri her alt-bantta farklı frekanslar olacak şekilde depolanacak biçimde tasarlanır. Şekil 4.8’de bir ve iki aşamalı DWT için örnek gösterilmektedir.



Orjinal Görüntü



1 Aşamalı DWT



2 Aşamalı DWT

Şekil 4.8: Dalgacık Dönüşümü

Hem kayıplı hem de kayıpsız sıkıştırma için standart tarafından farklı dalgacık filtrelerinin sayısına izin verilir. Kayıplı filtreler genellikle daha iyi sonuç verir fakat kayan noktalı işlemleri içerir. Kayan noktalı yaklaşımdan dolayı bu filtreler kullanılarak girilen sinyalin tekrar düzgün bir şekilde elde edilmesi garantilenemez. (Gaetano 2004)

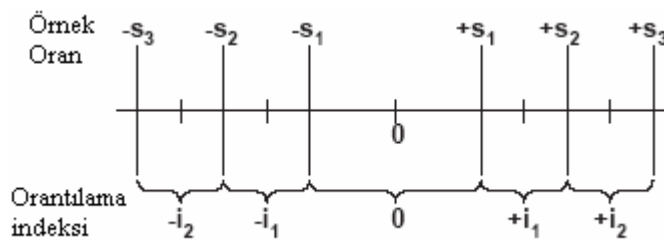
JPEG2000 her görüntü bölümünü ayırmak için ayrı dalgacık bölümlenmeyi kullanır. Ayrı dalgacık dönüşümü yüksek geçirgen ve alçak geçirgen filtreler ile ön işlem den geçirilmiş görüntü bölümünün her satır ve sütununun filtrelenmesi ile gerçekleştirilir. (Karen 2005)

4.6.4 Orantılama

Dalgacık katsayıları eşit aralıklı miktarlayıcılar kullanılarak orantılanır. Her alt-bant b için, basit bir orantılama sabiti bütün katsayıları orantılamak üzere aşağıdaki eşitliğe göre kullanılır.

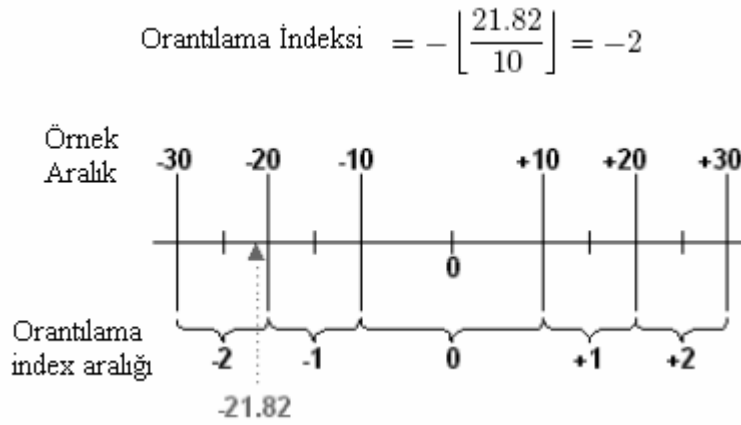
$$q = \text{sign}(y) \left\lfloor \frac{|y|}{\Delta_b} \right\rfloor \quad (4.1)$$

Y miktarlayıcı için girdi, $\text{sign}(y)$ ise y 'nin işaretini ifade eder iken, b adım büyüklüğü ve q sonuç miktarlayıcı indeks değeridir. Ölü bölge, orantılama oranının yaklaşık 0 yani $2b$ olduğu anlamına gelir bu ise daha fazla sıfırlı sonuç elde edilmesi demektir. Şekil 4.9'de orantılama yapısı ile ilgili bir örnek gösterilmektedir.



Şekil 4.9: Orantılama yapısı

Orantılama şöyle bir örnek ile açıklanabilir. Verilen orantılama adım aralığı 10 ve kodlama girdi değeri -21.82 ise miktarlayıcı indeks Şekil 4.10'deki denklemde gösterildiği gibi belirtilir.

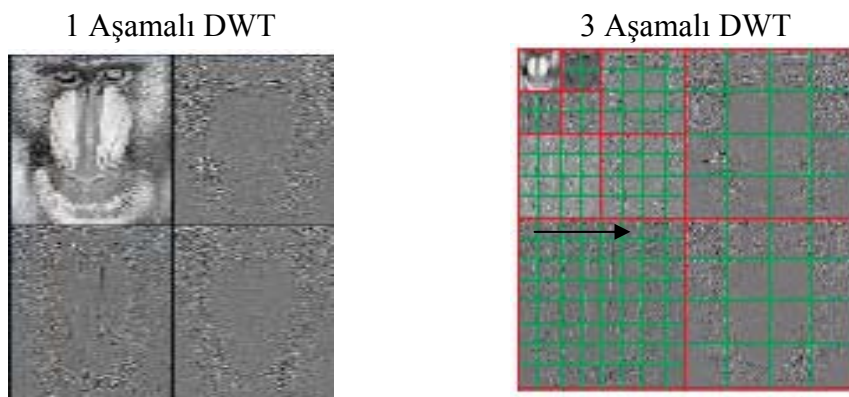


Şekil 4.10: Orantılama için örnek

4.6.5 Gömülü blok kodlama

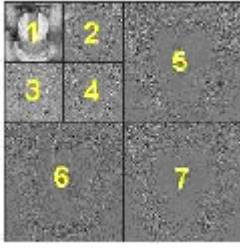
Gömülü blok kodlama tartışılmadan önce, gömülü blok kodlamada kullanılan kod blokları fikri açıklanmalıdır. JPEG2000'ta kodlama yapılmadan önce her bölümün altbandı eşit büyüklükteki göreceli küçük kod bloklarına ayrılır. (64x64 veya 32x32 boyutlarında)

Kod bloklar esnek bit dizisi organizasyonu için kullanılır. Örnek kod blok parçalama Şekil 4.11'de gösterilmektedir.



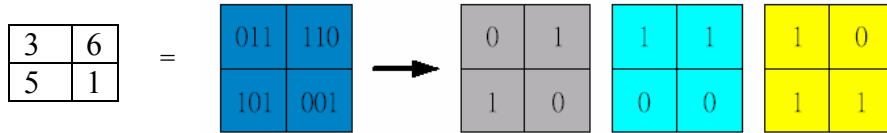
Şekil 4.11: Altbantların kod bloklara bölünmesi

JPEG2000'de her kod bloğu Şekil 4.12'da gösterildiği gibi bağımsız olarak kodlanır.



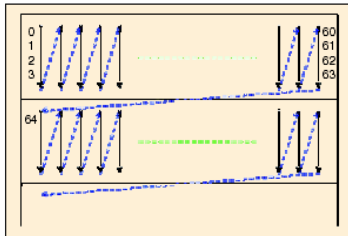
Şekil 4.12: Kod bloklar

Kod blokları Şekil 4.13'deki baskın (most significant) bit düzleminde başlanarak en az baskın (least significant) bit düzlemine kadar kodlanır.



Şekil 4.13: MSB düzleminde LSB düzlemine kod bloklar

Tarama deseni (scan pattern) olarak adlandırılan bir özel kod blok, bir kod bloktaki her bit düzlemi için üç kodlama geçişini yapmak amacıyla kullanılır.



Şekil 4.14: Gömülü blok kodlama

Bit düzlemindeki her katsayı biti üç kodlama geçiş türünden sadece biri ile kodlanır. Bunlar;

1- Değerlilik Yayılma Geçiş (Significance Propagation Pass)

- Eğer bir bit değeri sıfır (insignificant = 0) fakat şekiz komşusundan en az bir tanesinin değeri 1 (significant = 1) ise o bit kodlanır.
- Eğer bit aynı zamanda 1 ise, onun bayrağı 1 olarak ayarlanır ve sembolün işareti kodlanır.

2- Magnitude Refinement Pass

Değerlilik yayılma geçişi ile kodlanmayan ve değeri 1 olan örnekler kodlanır.

3- Temizleme Geçişi (Clean-up Pass)

Önceki iki kodlama geçişi ile kodlanmış bütün bitler kodlanır. MSB düzlemi için ilk geçiştir.

MSB düzleminde sadece temizleme geçişi gerçekleştirilir. Her bit düzlemi için üç kodlama geçişi de gerçekleştirilir.

4.6.6 Oran kontrolü

Oran kontrolü, belirli bir orana kadar bir görüntünün kodlanabilirliği. Kod dizini oran hedef bit değerine erişilene kadar kontrol işlemi yapılarak değiştirilir.

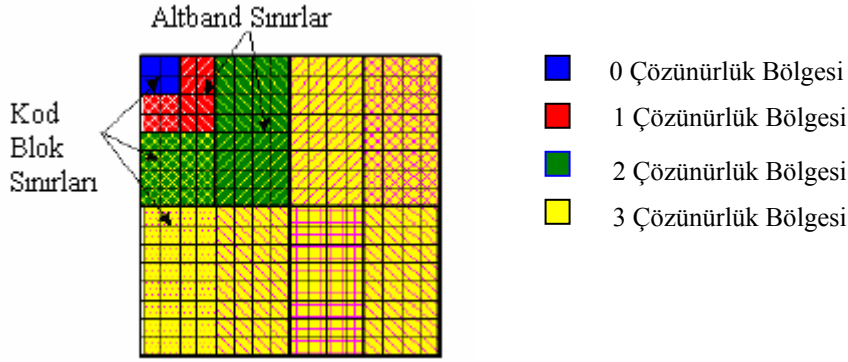
İlk önce görüntü sıkıştırılır, son işlem bütün sıkıştırılmış kod bloklarından geçirilir ve hedef bit oranına erişimi kısaltmak için her bloğun gömülü bit dizininin genişliğine karar verilir.

İdeal kısaltma stratejisi hedef bit oranına erişilirken bozulmayı en aza indirgeyen bir stratejidir.

Kod blokları her bit dizinine kısaltma işlemi uygulanabilecek şekilde bağımsız olarak sıkıştırılır.

4.6.7 Bit dizini organizasyonu

Bit dizini organizasyonunda, bit düzlem kodlama geçişlerinden geçen sıkıştırılmış veri, paketlere ayrılır. Bir paket bir bölümdeki (tile) her bölge (precinct) için oluşturulur. Bir bölge esas olarak bir çözünürlük seviyesindeki kod bloklarının gruplanmasıdır. Bölge bir parçadaki çözünürlük seviyesini $2P_x$ $2P_y$ örneklerini de içeren (P_x, P_y) boyutlarında ve eşit büyüklükteki dikdörtgen parçalara ayırır. Bölgelerin boyutları 2'nin katı büyüklükte olup kod blok parçalarının boyutlarını sınırlandırır. Bölge bölümlenmesi için bir örnek Şekil 4.15'te gösterilmektedir:



Şekil 4.15: Bit dizin organizasyonu

Her bölge boş paket olsa bile bir paket oluşturur. Bir paket başlık ve sıkıştırılmış veriden oluşmaktadır. Başlık kod dizin organizasyonunu içerir. Paketlere tek bir kod dizini oluşturulacak şekilde çarpma işlemi uygulanır.

Bölüm-1’de konumları bölge numarasını belirten aşağıda ifade edildiği gibi paketleri sıralamak için beş farklı yol vardır:

Kalite: katman, çözünürlük, parça, konum

Çözünürlük 1: çözünürlük, katman, parça, konum

Çözünürlük 2: çözünürlük, konum, parça, katman

Konum: konum, parça, çözünürlük, katman

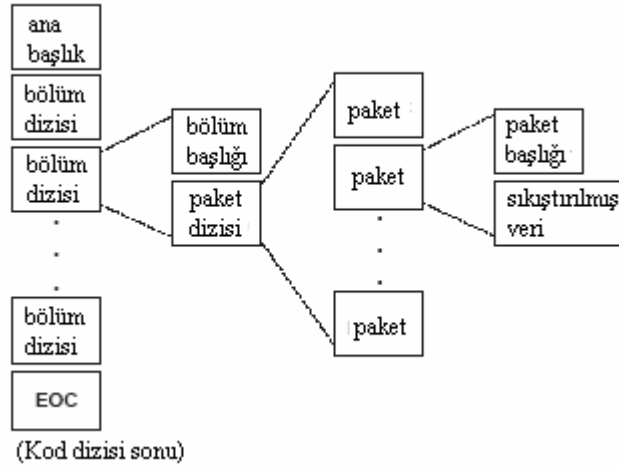
Parça: parça, konum, çözünürlük, katman

Sıralama mekanizmaları en çok baskın olandan en az baskın olana doğru gerçekleştirilir.

Örneğin; kalite yönteminde paketler ilk önce katman olacak şekilde sıralanır, ikinci olarak çözünürlük, üçüncü sırayı parça ve dördüncü sırayı konum almaktadır.

Ek olarak, bu standart için Bölüm-2’de kullanıcı tanımlı ilerlemeleri belirtmek mümkündür.

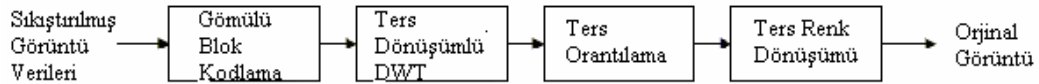
Şekil 4.16’te bir kod dizini organizasyonu gösterilmektedir.



Şekil 4.16: Kod dizini organizasyonu

4.7. Kod çözümü

Kod çözümü temel olarak kodlamanın tam tersini yapar. Şekil 4.17’de kod çözümü yapısı gösterilmektedir.



Şekil 4.17: Kod çözümü

Kod dizini başlık bölümünde belirtilen ilerleme sırasına göre kod çözümleyici tarafından ele alınır.

Paketlerdeki katsayılara kod çözümü ve diğer işlemler ile birlikte ters yönlü renk dönüşümü aşağıdaki ifade ile uygulanır.

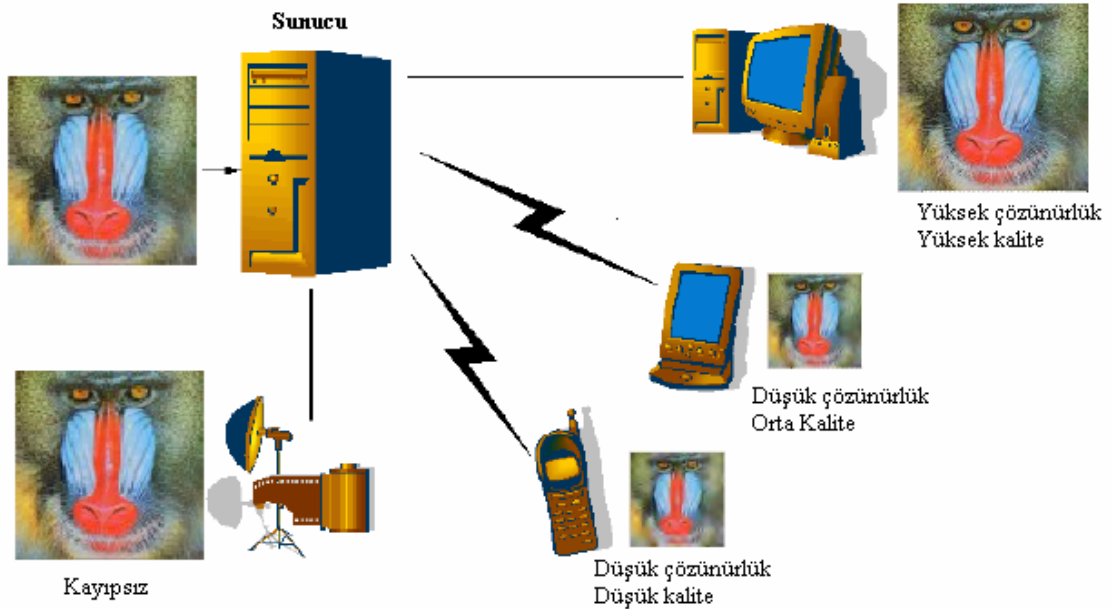
$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.0 & 0.0 & 1.4021 \\ 1.0 & -0.3441 & -0.7142 \\ 1.0 & 1.7718 & 0.0 \end{bmatrix} \begin{bmatrix} Y \\ C_r \\ C_b \end{bmatrix} \quad (4.1)$$

Bazı ters dönüşümü yapılamayan sıkıştırılmalarda, sıkıştırılan görüntünün tekrar açılması durumunda veri kaybı olabilir ve sonuç görüntü orijinaline benzemez.

4.8 JPEG2000'nin Özellikleri

4.8.1 Bir kere sıkıştır, pek çok yolla aç

JPEG2000 ile kullanılabilir en büyük çözünürlük ve en büyük görüntü kalitesine karar verilir. Sıkıştırılmış veri her görüntü kalitesi ve boyutunda açılabilir. Sıkıştırılmış görüntü üzerinde sadece belirli bir bölgedeki veriye rastgele erişim yapmak mümkündür. İstenilen bir görüntünün elde edilmesi için sadece gerekli bitlerin üzerine konumlanıp, sıkıştırılmamış hale getirilmesi ve kod çözümü işleminin yapılması mümkündür. Bu JPEG2000 standardının önemli bir özelliğidir. Şekil 4.18'de sıkıştırılmış bir görüntünün birden fazla yolla açılması gösterilmektedir.



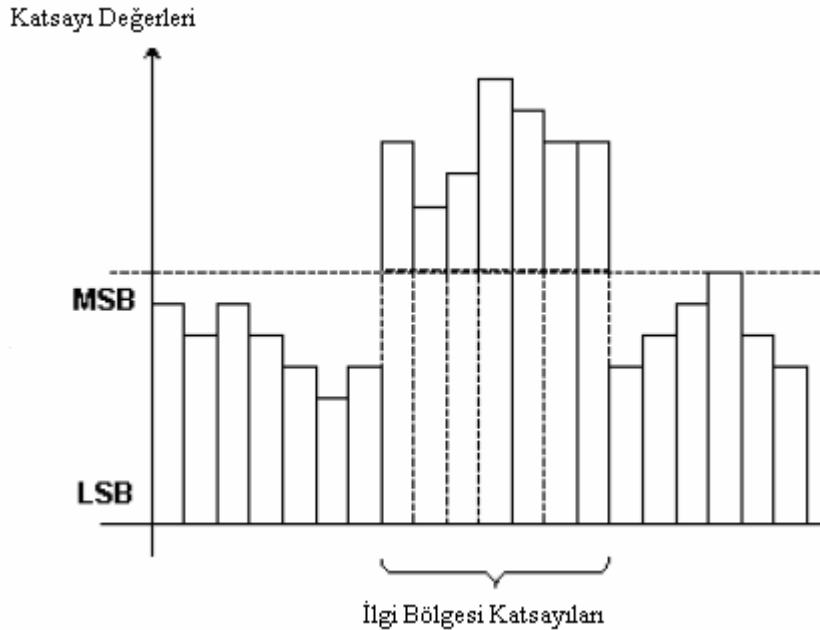
Şekil 4.18: Sıkıştırılmış görüntünün birden fazla yolla açılması

4.8.2 İlgi bölgesi

İlgi bölgesi kodlamasında seçilen bir bölge daha yüksek kalite ile kodlanırken görüntüdeki diğer bölgeler daha düşük bir kalitede kodlanır. Bu ilgi bölgesi statik veya dinamik bir bölge olabilir. Dinamik olması durumunda ilgi bölgesi tanımlaması verinin iletimi esnasında gerçekleşmekte iken statik olması durumunda ise bölge tanımlaması kodlama zamanında gerçekleştirilir.

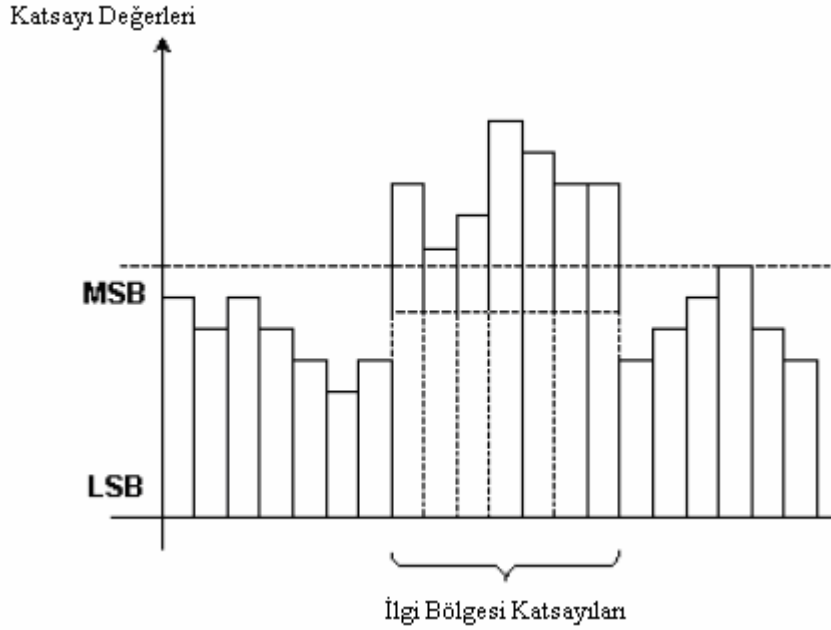
İlgi bölgesi alanlarının belirlenmesinde maskelerden yararlanılır. Bu maskeler ilgi bölgelerinin oluşturulmasında hangi dalgacık katsayılarının kullanılacağını belirler. İlgi bölgesi maskeleri kod çözümleyiciye maske katsayılarının yukarı kaydırılması ile gösterilir. Kod çözümleyici tarafında ilgi bölgesi maske katsayıları bulunup tekrar aşağı kaydırılır. JPEG2000 ilgi bölgesi maske katsayılarını yukarı kaydırmak için iki yöntem kullanılmasına imkan sağlar, bunlar en büyük kaydırma ölçeklemesi ve genel ölçeklemedir.

En büyük kaydırma ölçeklemesi standardın ilk bölümünde desteklenir. Bu bölümde, ilgi bölgesi maske katsayıları en az baskın bit değerleri (LSB) en baskın değerlerden daha yüksek olacak şekilde yukarı kaydırılır. İlgi bölgesi maskesi değerlerinin kodlayıcıya gönderilmesine gerek olmadığı için ilgi bölgesi maskesini bulmak kolaydır, sadece ilgi bölgesi ve şeklin iletilmesi yeterlidir. Bu metod belirli bir şekle sahip olmayan ilgi bölgesi şekillerini destekler. Şekil 4.19 ilgi bölgesi katsayılarının yukarı kaydırılmış şeklini göstermektedir.



Şekil 4.19: İlgi bölgesi

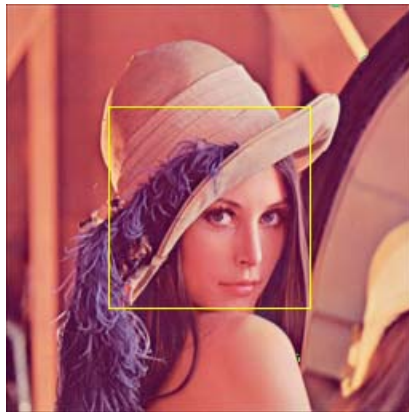
JPEG2000 standardındaki Bölüm-2 genel ölçeklemeyi destekler. Genel ölçeklemede ilgi alan bölgesindeki katsayılar Şekil 4.20’de gösterildiği gibi yukarı kaydırılır. Bu yöntem her ilgi bölgesi farklı değerler ile yukarı kaydırılarak çoklu ilgi bölgelerinin görüntüdeki tek bir bölüm içinde olmasına izin verir.



Şekil 4.20: İlgi bölgesi maskesinin kaydırılmış katsayıları

Her ilgi bölgesindeki kod dizin başlığında yukarı kaydırma değerleri kaydedilir. Bu yöntem ilgi bölgesi maskesinin kolaylıkla tanımlanabilmesine izin vermediği için kod çözümleyicide mevcut olmalıdır.

Bu method aynı zamanda dikdörtgen ve elips gibi basit ilgi bölgesi şekillerini destekler. Şekil 4.21’te ilgi bölgesi için örnekler gösterilmektedir.



İlgi bölgesi belirlenmiş orjinal görüntü



Kodlanmış görüntü

Şekil 4.21: İlgi bölgesi – Örnek

4.8.3 Hatadan korunma

JPEG2000 standardının özelliklerinden biri de hatadan korunma yöntemlerinin geliştirilmesidir. Bu özellik kablosuz iletişimin öneminin artması ile daha da vazgeçilmez hale gelmiştir.

Her kod bloğun bağımsız olarak kodlanması sayesinde hatalar o kod blokları içinde sınırlı kalabilmektedir. Değişken uzunluktaki kodlamalar kullanıldığı için çeşitli bozulmalar meydana gelebilir. Ayrıca paket başlıkları oldukça kırılmalıdır.

Kod blok verilerinin korunması aşağıdaki üç farklı yöntemden biri ile gerçekleştirilebilir.

1. Bölümleme sembolleri: Her bit düzleminin sonunda özel sembol sırası kodlanır. Kod çözümü eğer yanlış sırada yapılırsa bir hata meydana gelir ve bu durum en azından son bit düzlemin bozulmasına sebep olur.

2. Düzenli tahmin edilebilir bitiş: Her kodlama geçişi sonunda özel bir tahminleme algoritması kullanılarak aritmetik kodlayıcı tamamlanır. Kod çözümleyici bitiş kodunu tekrar üretir ve eğer aynı kullanılmayan bitler bulunmazsa en son kodlama geçişinde hata meydana gelir.

3. Her iki mekanizma birlikte kullanılabilir fakat bu durum sıkıştırma etkinliğini azaltır.

Paket başlıklarının korunması için ise aşağıdaki üç farklı yöntemden biri kullanılabilir.

1. Paket başlangıcı (Start of Packet - SOP) eş zamanlama yapıcısı: Her paket sıralı- indeksli SOP işaretleyicisi tarafından öncelikle işaretlenir. Eğer bir SOP işaretleyicisi ile doğru sıradaki indeks hemen paket başlığından önce bulunmaz ise bir hata meydana gelmiştir. Bu durumda kod dizininde bir sonraki etkilenmemiş paket aranır ve kod çözümüleme işlemine oradan devam edilir.

2. PPM/PPT (Ana Paketin Korunması/Paket Bölümlerinin Korunması) işaretleyicisi: Paket başlık içeriği kod dizinindeki ana görüntüye veya bölümlerin başlıklarına taşınır ve kanallarda daha düşük bir hata oranı ile iletilir.

3. Bölgeler (Precincts): Paket başlık hatalarının küçük görüntü alanında sınırlandırılmasıdır.

Aşağıda sıkıştırılmış görüntülerden farklı çözünürlüklerde elde edilen sonuçlara ilişkin örnekler gösterilmektedir.

Örnek – I

Şekil 4.22’de, 0.25bpp’de sıkıştırılan görüntülerin tekrar açılmış hali gösterilmektedir.



Şekil 4.22: 0.25bpp’de sıkıştırılmış görüntü sonuçları

Örnek – II

Şekil 4.23’de, 0.125bpp’de sıkıştırılan görüntünün tekrar açılmış hali gösterilmektedir.



Şekil 4.23: 0.125bpp’de sıkıştırılmış görüntü sonuçları

4.9 Sonular

Yararlar: kayıplı ve kayıpsız sıkıştırma daha yüksek görüntü kalitesi ve sıkıştırma oranları, oklu özünürlüklerde dosyanın görüntülenmesi, ilgi bölgelerinin kullanılabilirliđi ile ele alınacak görüntü alanının daha yakından incelenebilirliđinin sağlanması gibi yararları vardır.

JPEG2000 görüntüleri sıkıştırma da dalgacık teknolojisini kullanır (görüntüler daha etkili sıkıştırılır). Günümüzde, JPEG2000 kayıplı sıkıştırma için bir dalgacık ve kayıpsız sıkıştırma için başka bir dalgacık kullanır fakat gelecekte diđer dalgacıklar ihtiyaç arttıkça kullanılabilir.

Internet, tıbbi görüntüleme ve sayısal fotođraflama gibi pek ok uygulamalarda kullanılabilir.

Bütün bunlardan başka, JPEG2000 mevcut sıkıştırma metotlarının büyük bir güncellemesidir ve geleceđin görüntü sıkıştırma standardı olarak görölmektedir.

Bir sonraki bölümde, yazılım geliştirme platformu olarak kullanılan Visual Studio 2005'in özellikleri üzerinde durulmaktadır.

5. MICROSOFT VISUAL STUDIO 2005

Bilindiği gibi Visual Studio Windows uygulamaları, ASP web uygulamaları, XML web servisleri ve mobil uygulamalarını geliştirmek için kullanılan araçlar kümesi olarak ifade edilebilir. Visual Basic .NET, Visual C++.NET, Visual C#.NET ve Visual J#.NET gibi bütün diller karışık dil çözümlerinin oluşturulmasında aynı geliştirme ortamını (IDE-Integrated Development Environment) kullanır. Ayrıca, bu diller tarafından uygulamaların geliştirilmesini basitleştirmek için anahtar teknolojilere erişimi sağlayan .NET Framework kullanılır. (WEB_4 2006)

5.1 .NET framework temelleri

.NET Framework, farklı programlama dilleri ve kütüphanelerinin birlikte çalıştırılmasına ve Windows tabanlı uygulamalar oluşturulmasına ayrıca diğer ağ sistemleri ile bütünleşik olmasına izin veren bir geliştirme ve çalıştırma ortamıdır.

.NET Framework şu iki kısımda incelenebilir:

- **Çalışma Zamanlı Ortak Dil (Common Language Runtime - CLR)**

Uygulamanın çalıştırılması ve yönetilmesine yardımcı olacak servisleri sağlayan dil geliştirme ve çalıştırma ortamıdır.

- **Framework Sınıf Kütüphaneleri (FCL)**

Bu kısım nesne tabanlı kütüphanelerden ibarettir.

Şekil 5.1'de .NET Mimari yapısı gösterilmektedir.



Şekil 5.1: .NET Mimarisi

5.2 Visual Studio 2005'in Getirdikleri

Visual Studio 2005 ile birlikte gelen gelişmeler; IDE'deki gelişmeler, Windows formları, .NET Compact Framework, Visual Studio 2005 araçları, Akıllı istemci, Derlemek ve hataları düzeltmek başlıkları altında incelenecektir.

5.2.1 IDE'deki gelişmeler

Visual Studio 2005 çok çeşitli gelişmeleri beraberinde getirmektedir. Ortak IDE özellikleri daha kullanışlı arayüz sağlamak için basitleştirilmiştir ve böylelikle kullanıcılar tercih edilen ayarları kolay bir şekilde seçebilmektedir. Formdaki geliştirmeler görsel hizalama göstergelerinden yararlanır ve akıllı taglar ortak görevlere hızlı erişim sağlar. Bütün bunlardan başka, kod geliştirmesi tümleşik sınıf tasarımcıları ve güçlü kod analiz özellikleri ile daha kolay hale getirilir. Bu durum geliştirme işleminin hızlanmasını ve kodların daha iyi yazılmasını sağlar.

Visual Studio 2005 IDE çeşitli anahtar alanlarda büyük ölçüde basitleştirilmiştir. Bunlardan bazıları şu şekilde sıralanabilir: proje tasarımcısı adı verilen bir proje araç kutusunu kullanan proje özellik ayarlamalarına erişim yeteneği, ve tüm bir projeyi diske kaydetmek zorunda kalmadan hızlı bir şekilde bazı fikirleri deneyebilecek geçici projeleri kullanabilme yeteneği.

Visual Studio 2005 iyi tasarlanmış ve iyi bir mimariye sahip çözümleri üretebilmeyi sağlamak için aynı zamanda güçlü bir sınıf tasarımcısı ve kod analiz araçlarını destekler. Sınıf tasarımcısı görsel olarak sınıfları ve arayüzleri oluşturmaya ve kodlar ile diagramlar arasında zahmetsizce ileri geri harekete izin verir.

Ayrıca, geliştirilen ortam ayarlamalarını kolay bir şekilde dahil edip harici bırakabilme yeteneğine ve bunun yanında diğer geliştiriciler ile bu ayarları paylaşma yeteneğine de sahiptir. Bu ortamda bir kullanıcı artık mükemmel kişileştirilmiş ortamı kullanabilir ve ayarları kaydedebilir ve öyle ki bu ayarlar başka bir makineye daha sonra dahil edilebilir. Bu özellik gelişim ortamı ayarlamaları çerçevesinde birden fazla eş zamanlı bilgisayar kullanabilmek için proje geliştirmeyi ve paylaşmayı kolaylaştırır.

Akıllı tag'ler ayrıca tasarım zamanında kontroller üzerindeki özellikleri hızlı bir şekilde belirleme yeteneğini sağlar. Bu görev tabanlı akıllı tag'ler Microsoft Ofis içinde bulunanlarla benzerdir ve kisisel kontroller için özellik pencerelerini dolaşmak yerine istenilenlere daha hızlı bir şekilde ulaşılmasını sağlar.

Visual Studio 2005 birbirine göreceli olan kontrollere ve formlara genişletilmiş ölçüde destek verir. Snapline olarak adlandırılan araçlar, form üzerindeki kontroller arasındaki uygun mesafeleri tutmakla birlikte yukarı/aşağı, sol/sağ yönlerinde hizalama işlemlerini hızlı bir şekilde yapma yeteneğini sağlar.

Programlardaki hataların ayıklanması sırasında, program geliştiriciler verinin nasıl değiştiğini anlamak durumundadır. Geçmiş yıllarda, hata ayıklayanlar yerel değişkenlerin nasıl değiştiğini anlamak için izleme pencerelerini, kayıtçıları ve bellekleri içeren program verilerinin sürekli artan sayıda görüntülerini kullanarak program geliştirme verimliliğini arttırmışlardır. Visual Studio hata ayıklayıcısını kaynak kodu penceresinde doğrudan acan veri tiplerine ve araç tiplerine sahiptir ve fare imlecinin değişken üzerine götürülmesi durumunda o değişkenin değerini verir.

Visual Studio 2005 geliştirmeyi hızlı ve kolay yapmak üzere tasarlanmış bir çok yeni özelliği de destekler. Tekrar derlenebilir kod bloklarından bir kerede gözden geçirip düzeltme durumuna ve tümleşik test araçlarından güçlü seçeneklere kadar pek çok özellik her zaman mevcuttur.

Kodun hangi satırlarının değiştirildiğini görsel olarak kolayca belirlemek mümkündür veya herhangi bir oturumda kodun kaydedilip kaydedilmediği belirlenebilir.

Satır gözden geçirme işaretçileri renkli kodlanmıştır ve kodlamadaki değişikliklerin durumunu hızlı bir şekilde görmeye izin verir.

Uygulamaları oluştururken ortak işlemler için tekrar kullanılacak kod kütüphanesine erişim sahibi olmak çok faydalıdır. Sadece bu amaç için bir çok geliştirici bağımsız kütüphane tutar. Visual Studio 2005 ile dosya işlemleri, ağ erişimleri ve çok fazla sayıda ortak işlemler kod parçacıkları (kod snippets) adı verilen

kod editor penceresindeki bir hiyerarşik seçim menüsü üzerinden ulaşılan araç ile mevcut hale getirilmiştir.

5.2.2 Windows formları

Visual Studio 2005 ile Windows form uygulamaları geliştirilirken kullanıcılar için güçlü bir arayüz oluşturulması için zengin kontrol kümesine erişim hakkına sahiptir. Windows form uygulamalarına XP tema desteği verildiği gibi ToolStrips, MenuStrips gibi kontroller için de gelişmiş destek mevcuttur. Visual Studio 2005 üçüncü parça kontrollerin kullanımına izin verdiği gibi yüzden fazla kontrolü de destekler.

Uygulamalar web sitesine, ftp sunucusuna, ağ dosya paylaşımına veya yerel medyaya Visual Studio ortamında gerçekleştirilebilir. Bir defa gerçekleştirildiğinde, ClickOnce uygulamaları sunucudaki değişikliklere cevap olarak, kendilerini otomatik olarak güncelleme gücüne sahiptir ki, bu genel sistem yönetimini çok büyük ölçüde basitleştirir. Program geliştiriciler bu güncellemelerin ne zaman ve nasıl gerçekleşeceği konusunda tüm kontrolleri üzerlerinde taşırlar ve tam bir API gerçekleştirilmesinde erişim sahibidirler.

5.2.3 .NET compact framework

.NET Compact Framework bir sonraki akıllı cihaz kuşağını gerçekleştirmek için gerekli olan araçları ve ortam kütüphanesini önemli bir adım olarak geliştiricilere sunar. Bu akıllı cihazların önümüzdeki üç yıl süresince çok fazla gelişeceği göz önüne alındığında bu çalışma alanı kritik bir geliştirme konusu olarak gündemde kalacaktır. .NET Compact Framework 2.0'daki gelişmelerin anahtar noktaları kullanıcı arayüzleri kurmak, bir yeni mobil veritabanı makinesi, veri merkezli (data centric) uygulamaları üretmek ve test etmek için gerekli yeteneklerin geliştirilmesidir.

Buna ilave olarak önemli ölçüde emulator geliştirmelere, yeni çekirdek framework zenginleştirmeleri ve daha iyi komponent, kesinti (interrupt) desteği ve 3D için yönetilmiş arayüzlere sahiptir.

Visual Studio 2005'teki .NET Compact Framework aynı zamanda çok az bir koda sahip akıllı cihaz için üretilen karmaşık kullanıcı arayüzü kurmak yeteneğini değişik

yeni kontrol mekanizmaları ile sağlar. Aylık takvim ve zaman kontrollerini formlara sürükleyip bırakmak kolaydır ki bu takvim gibi bir fonksiyonellik sağlar. Buna ilaveten kullanıcı için etkileşimli (interactive) mesajları ileten bir bildirim kontrolü, dosya yönetim uygulamalarını gerçeklemek için belgeleme kontrolü ve donanımı daha fazla kontrol etmenizi sağlayan donanım düğmesi ve ekran uyumluluk bileşenlerine sahiptir.

.NET Compact Framework Visual Studio 2005 sürümündeki komponentler ve .NET Compact Framework için çok zenginleştirilmiş desteğe sahiptir. Yönetilen koddan direk bileşenlerini ve bu bileşenler üzerindeki çağrı (call) fonksiyonlarını direk olarak çağırılabilir. Bu özellik .NET Compact Framework'e dayanan uygulamaları mevcut fonksiyonlardan avantaj sağlayarak yazma yeteneğini sağlar.

.NET Compact Framework uygulamalarının geliştirilmesinde anahtar noktalardan bir tanesi de test yapmaktır. Paket PC ve akıllı telefon emulatu ru uygulamaları daha kolay derleme, test etme ve gerçekleyen zenginleştirilmiş fonksiyonlara sahiptir. Bu özellik paylaşılmış klasörleri kullanmayı ve ağ ayarları üzerinde daha iyi kontrolü mümkün kılar. Buna ilaveten emulator güçlendirilmiş performansa sahiptir ve uygulamaları daha hızlı geliştirmeyi sağlayan çoklu kaydedilmiş emulator yeteneğini sunar.

SQL Mobil 2005 özellikle eş zamanlı geliştirmeler alanında bir çok güçlü özellik içerir. Bu sayede yönetilmiş kodları kullanarak eş zamanlama gelişiminin gözlemlenmesi mümkündür ve kolay bir şekilde kullanıcıların eş zamanlama durumunu görmesi sağlanır. Aynı zamanda satır seviyesindeki eş zamanlamadan ziyade sütun seviyesindeki eş zamanlama kullanıldığında ağ üzerinde iletilen veri oranı, normal eş zamanlama durumunda iletilenden daha azdır.

Akıllı cihaz CAB projesinin geliştirilmesi ile mobil uygulamalarının gerçekleşmesi çok daha kolay bir hale gelmiştir.

Akıllı cihaz CAB projesi dosyaları çok kolay bir şekilde sürükleyip bırakmak için bir görsel editor kullanır, aynı zamanda dosya oluşturmada kayıt girişlerini mümkün kılar, ve programcılarının aynı masaüstü bilgisayarlarda kullanılan windows kurulum projesinde olduğu gibi kolay bir şekilde oluşturulmasını sağlar.

Son olarak .NET Compact Framework 2.0 yönetilmiş kodlarda yazılan oyun geliřtirmeleri için DirectX desteęini saęlar. Yönetilmiş DirectX, yönetilmemiş DirectX ile karşılaştırıldığında daha hızlı geliştirme zamanı sunar ve ayrıca daha zengin gelişim deneyimi yönetilmiş kodla saęlanır.

5.2.4 Visual Studio 2005 araçları (VSTO)

VSTO, yönetilmiş kodlar kullanılarak Microsoft Ofis sistem ile entegre olmuş çözüm gelişimleri için saęlam ortam sunar.

VSTO 2005 Microsoft Excel 2003, Microsoft 2003 ve Microsoft Infopack 2003 için bir belgeleme merkezli gelişim ortamı bulundurur, kod geliřtirmede büyük oranda zenginleşirmeler saęlar. Aynı zamanda uygulama ve fonksiyonel olarakta güçlüdür.

5.2.5 Akıllı istemci

Visual Studio 2005 akıllı istemci uygulamaları geliřtirmeyi kolaylaştırır. Akıllı istemci uygulamaları, yerel sistem kaynaklarının gücüne erişmek için üzerinde çalışmalar yapılan web tabanlı uygulama modellerine uyum saęlayan masaüstü uygulamalarının ne kadar geliřtięinin bir göstergesidir.

Akıllı istemci uygulamaları, aynı geliştirme gayreti gösterildięi halde web uygulamaları ile erişilemeyen zengin bir kullanıcı deneyimini destekler.

Akıllı istemci uygulamaları, Visual Studio içinde mevcut olan kullanıcı arayüzü kontrollerinden DataGridView, ToolStrip ve LayoutPanel gibi zengin masaüstü uygulamaları yapılmasını destekleyen, en yeni kontrollerde dahil olmak üzere hepsini kullanabilir.

5.2.6 Derlemek ve hataları düzeltmek

Visual Studio 2005 uygulama geliştirme süresince derleme ve hataları düzeltme işlemleri için güçlü seçenekler sunar. Uygulamalarınızı çalıştırmadan metotları test

edebilir, kodda deęişiklikler yaparak hataları düzeltebilir veya bilinen hata yakalama teknikleri ile çalışma zamanlı hataları izleyip düzeltebilirsiniz. Bu yeni özellikler ile daha kısa zamanda daha çok hata bulabilirsiniz.

Ayrıca, Visual Studio 2005 C# ve VB dilleri için güncelleme işlemi yapıldıktan sonra programın tekrar durdurulması ve derlenmesi işlemlerine gerek kalmadan çalıştırılmasına kaldığı yerden devam etmesini sağlayan (Edit and Continue) özelliğini de destekler. Uygulamadaki hataların yakalanması ve düzeltilmesi için bu oldukça yararlı bir özelliktir.

Uygulamadaki hataların tespit edilmesi için Visual Studio 2005 ayrıca Olağan Dışı Durum Yardımcısı olarak adlandırabileceğimiz Exception Assistant sağlar. Bu sayede, program geliştiricilere hatanın nerede oluştuęu, sebebinin ne olduęu ve bu hatadan kurtulmak için gerçekleştirilebilecek çözüm önerileri sunulur. Oluşan hata için sorunu çözmeye yönelik çeşitli dökümanlara bağlantıları da içeren bir pencere açılarak bu bilgiler görüntülenir.

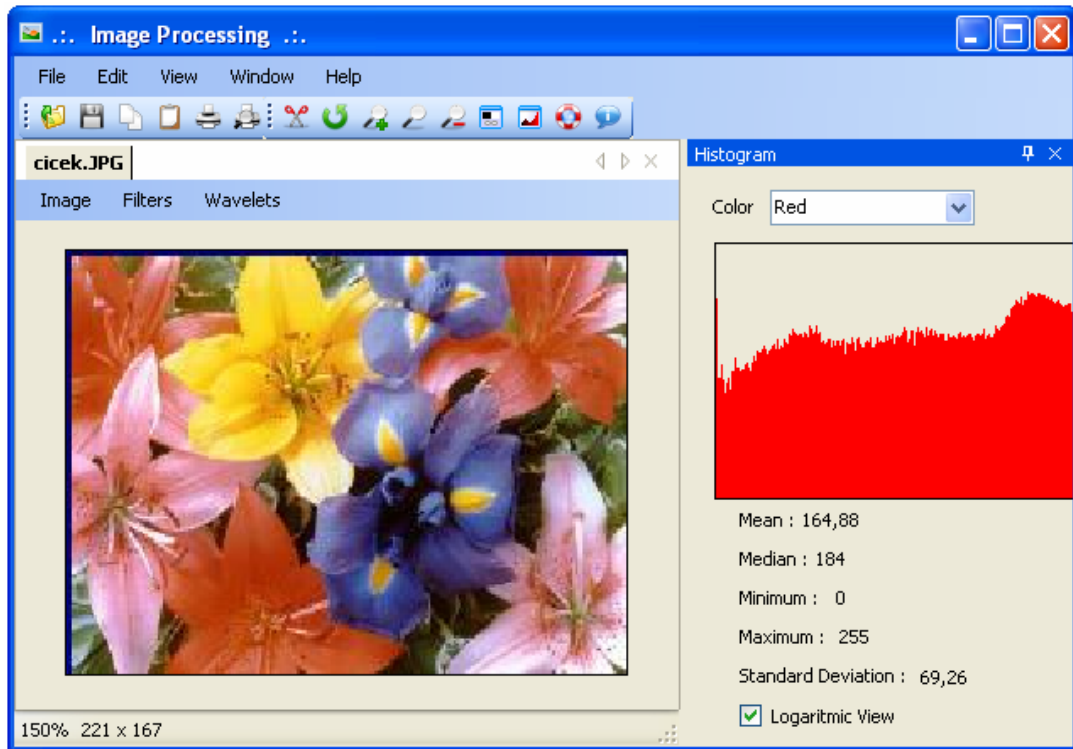
Visual Studio 2005 yeni derleme sistemini destekler. Bu sistem en karmaşık olanından en basit düzeye kadar derleme gereksinimlerinin kurulmasına izin verir. Bu yeni derleme sistemi, derleme işlemini tanımlamak için XML tabanlı dosyaları tanımlar, böylelikle derleme işleminin kullanıcıya özelleştirilmesine olanak sağlamış olur.

Visual Studio 2005 çok işlemcili derleme özelliğine sahiptir. Günümüzde simetrik ve çoklu iş parçacıklarını destekleyen çok işlemcili bilgisayarlar, gelecek yıllardaki çok işlemcili bilgisayarlar ile de uyumlu çalışabilecektir. Çok işlemcili bilgisayarların avantajlarından yararlanılarak, Visual Studio ile projelerin derlenme süresi kısaltılabilmekte ve aynı anda çeşitli projeler derlenebilmektedir. Bir sonraki bölümde, Visual Studio 2005 kullanılarak geliştirilen yazılımın özellikleri açıklanarak örnekler verilmektedir.

6. GELİŞTİRİLEN YAZILIM HAKKINDA

6.1 GİRİŞ

Bu tez çalışması ile geliştirilen yazılım, görüntüleri analiz etmek için farklı filtreler ve araçları içeren bir görüntü işleme aracıdır. Şekil 6.1’de geliştirilen yazılıma ait bir görüntü yer almaktadır.



Şekil 6.1: Görüntü işleme yazılımı

Uygulamanın önemli özelliklerinden biri diğer görüntü tiplerine ek olarak JPEG2000 standardını destekliyor olması (uzantısı J2K, JP2, JPC olabilir) ve bu özellik sadece doğal, bilimsel ve uzaktan algılamalı görüntüler için değil aynı zamanda çok boyutlu tıbbi görüntüler içinde geçerlidir.

Diğer önemli özellik ise, görüntülerin sıkıştırılmasında dalgacık (Daubechies) ve Lifting Scheme kullanılmasıdır. Önemli özelliklerinden bir diğeri ise görüntü sıkıştırma işleminin sadece bir boyutlu görüntüler için değil aynı zamanda iki boyutlu, üç boyutlu ve hatta dört boyutlu görüntüler için de geçerli olmasıdır.

Uygulamanın son bir özelliği ise .NET teknolojisinin son versiyonu olan Visual Studio 2005 ortamında gerçekleştirilmiş olmasıdır.

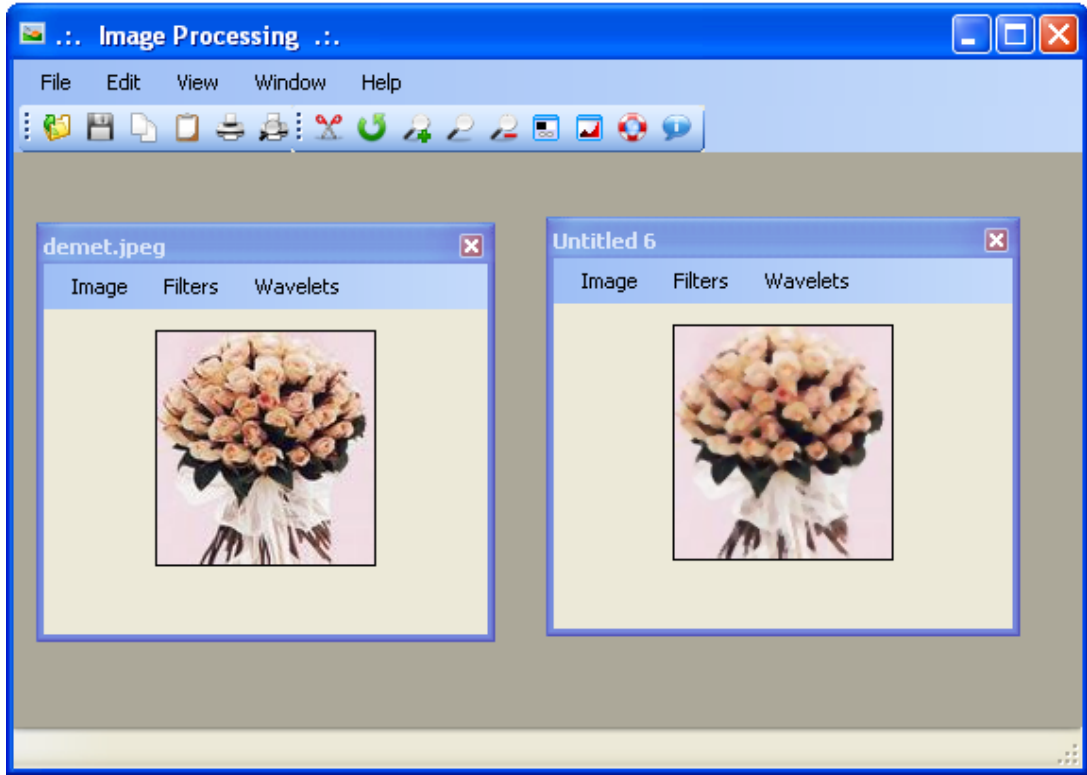
Uygulamada aşağıdaki filtre teknikleri gerçekleştirilmiştir:

- Median
- Threshold
- İskeletleştirme
- Parlaklık Duzeltme
- Adaptive Smoothing
- Bağlantılı Parça Etiketleme
- Fourier Dönüşümü
- Morfoloji (Eritme, Genleşme, Açma, Kapatma, İnceltme, Kalınlaştırma)
- Renk Filtreleme (Kanal Filtreleme, Gri renklendirme ve görüntünün kırmızı, yeşil, mavi, camgöbeği, sarı, eflatun gibi renklere boyanması)
- Kenar Belirginleştirme (Sobel ve Canny kenar belirginleştirme yöntemleri)
- Convolution ve Correlation (Klasik Convolution & Correlation, Keskinleştirme, Mean, Kenarlar, Bulanıklık, Gaussian Blur)
- İki görüntünün kombinasyonu (Ekleme, Çıkarma, Birleşme, Kesişim)

6.2 Median Filtre

Median filtre görüntüdeki gürültüyü azaltmak için kullanılır. Median değeri, bir pikselin komşuluğundaki piksel değerlerinin sıralanması ve ortanca değerinin bulunarak yer değiştirme işleminin yapılması ile sağlanır. Eğer komşulukta çift sayıda piksel var ise sıralama işlemi yapıldıktan sonra ortadaki iki pikselin ortalaması median değeri verir. (WEB_6 2004)

Şekil 6.2'de median filtre uygulanmadan önce ve uygulandıktan sonraki sonuç gösterilmektedir. Şekilde de görüldüğü gibi filtreleme işlemi sonucunda elde edilen görüntü ilk görüntüye göre oldukça bulanıktır.



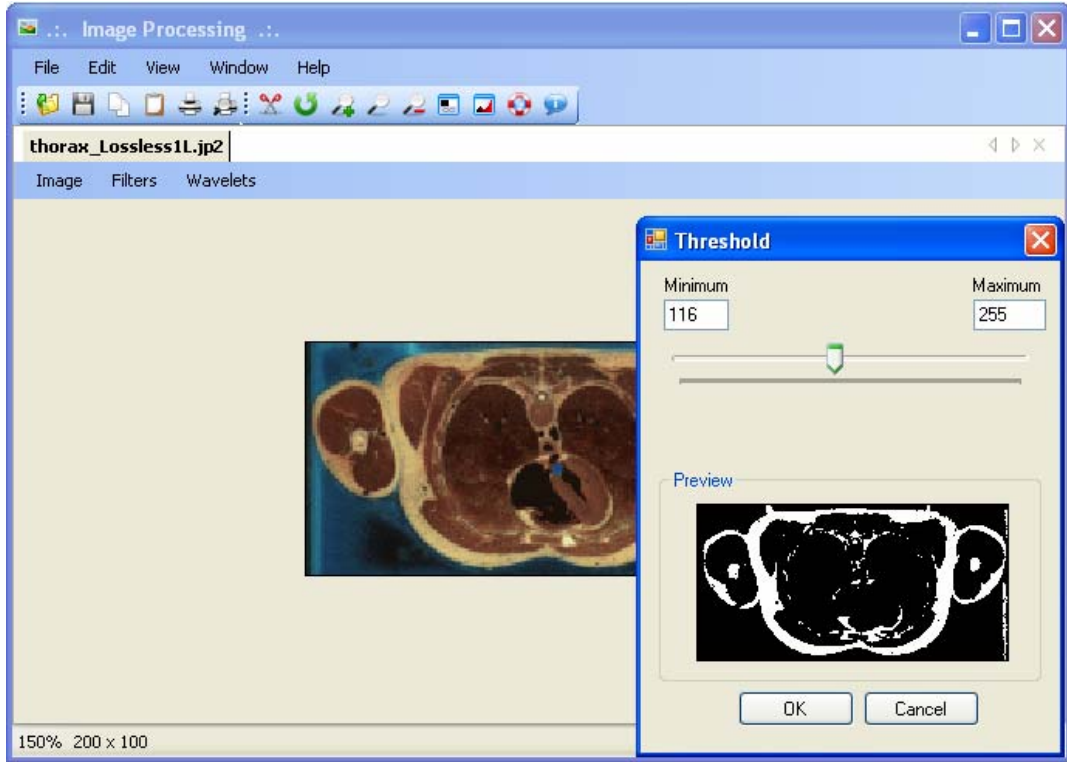
Şekil 6.2: Median Filtre

6.3 Thresholding

Thresholding en çok bilinen bölge tabanlı bölümlenme tekniğidir. Gri renk tonları ve renkli görüntü thresholding işlemine girdi olarak kullanılır ve çıktı olarak bölümlenmeyi gösteren ikili görüntü (siyah ve beyaz) elde edilir.

En basit thresholding işlemi şu şekilde yapılır: ilk olarak sabit bir gri seviyesi seçilir. Bu seçilen gri seviyesinden daha koyu olan piksel değerleri siyah, daha açık olanlar ise beyaz olarak belirtilir. Siyah pikseller arka planı ve beyaz pikseller de görüntüyü gösterir veya tam tersi şeklinde düşünülebilir.

Şekil 6.3'te threshold işlemi için bir örnek gösterilmektedir. Kullanıcı işlem için gereken verileri kendisi verebilir ve işlemin etkilerini ön izleme ekranından görebilir.

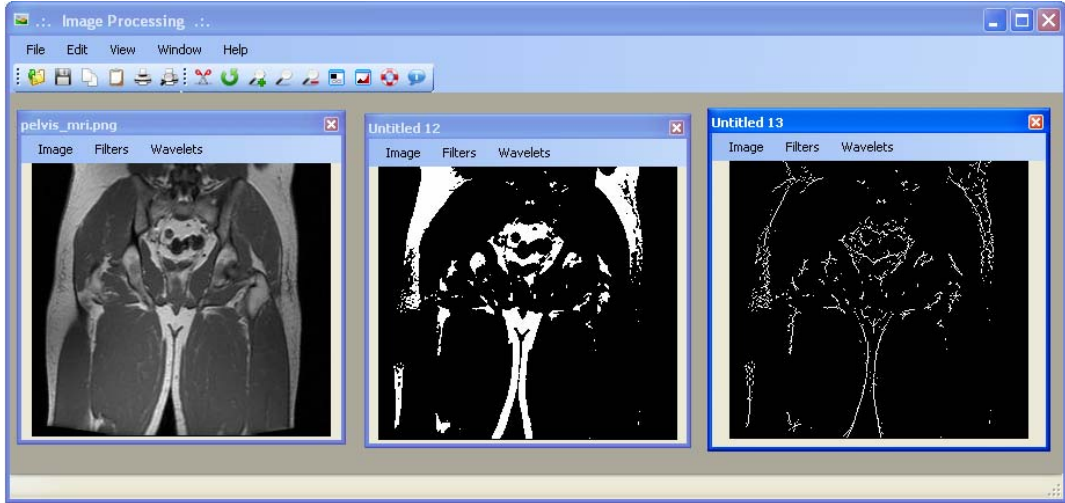


Şekil 6.3: Threshold işlemi

6.4 İskeletleştirme

İkili görüntüdeki ön taraf (foreground) bölgelerin görüntünün genel şeklini bozmayacak şekilde azaltılması için yapılan işlemdir (inceltme işlemi bir çeşit iskeletleştirme işlemidir) (WEB_6 2004)

Şekil 6.4'te üç görüntü gösterilmektedir. İlk resim orjinal şekli, ikincisi threshold işleminden sonraki şekli, ve üçüncüsü ise ikinci görüntünün iskeletleştirilmiş halini göstermektedir. Bu görüntüdeki iskeletleştirilmiş şekil morfolojik işlemlerden biri olan inceltme işlemi ile de elde edilebilir.



Şekil 6.4: İskeletleştirme

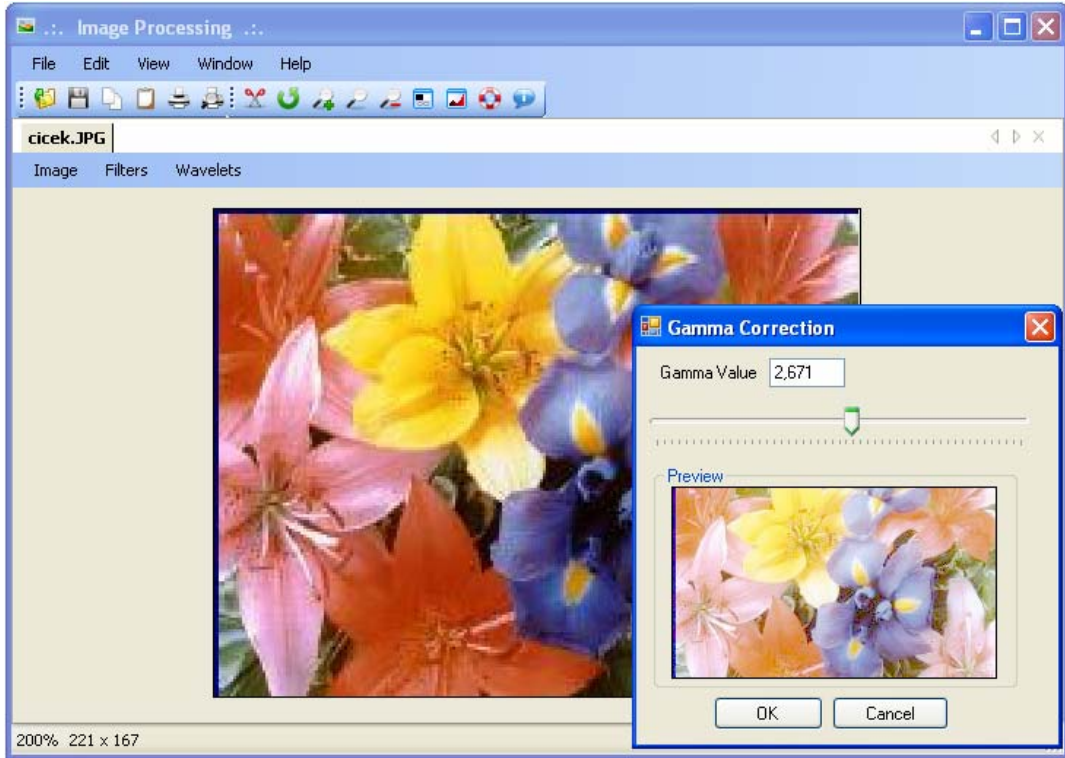
6.5 Parlaklık Düzeltme

Parlaklık Düzeltme ile görüntüdeki parlaklık kontrol edilebilmektedir. Renklerin doğru bir şekilde tekrar elde edilmesi için gamma değerine ihtiyaç duyulur.

Gamma miktarını değiştirmek sadece parlaklığı değil kırmızı, mavi, yeşil oranlarını da değiştirir. Kullanıcı bu oranları değiştirdikçe önizleme ekranından orjinal resim üzerindeki etkilerini kolaylıkla izleyip, istediği parlaklık değerini resim üzerine uygulatabilir. Parlaklık değerini kullanıcı manuel olarak verebileceği gibi ön izleme ekranındaki izleme çubuğunu hareket ettirerekte gerçekleştirebilir. İstenilen parlaklık değeri ile elde edilen sonuç ön izleme ekranından takip edilip orjinal görüntüye uygulanabilir.

Parlaklık düzeltme işlemi genellikle $x = y^\gamma$ (x :gösterilen görüntü, y :görüntü, γ :gamma) ifadesi ile hesaplanılarak yapılır.

Şekil 6.5'te parlaklık düzeltme ile ilgili bir örnek gösterilmektedir.



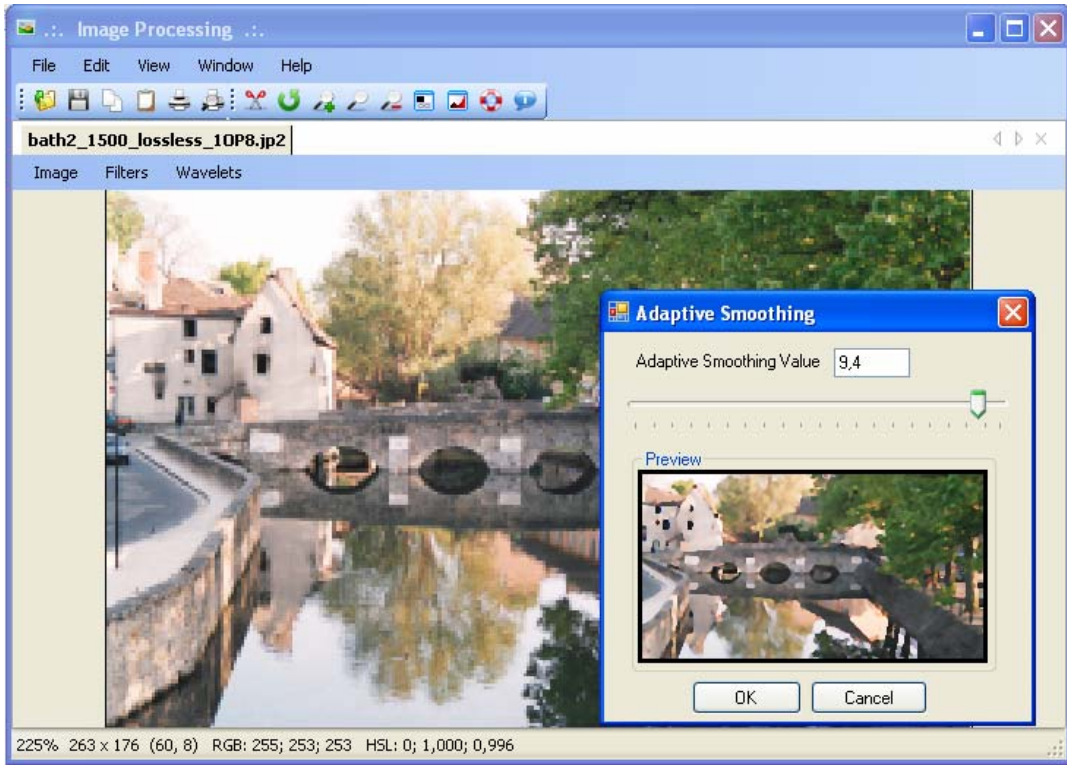
Şekil 6.5: Parlaklık Düzeltme

6.6 Adaptive Smoothing

Adaptive Smoothing yeni görüntü düzeltme yöntemidir. Geleneksel düzeltme yöntemleri aynı pencere şekli ve büyüklüğünü veya görüntüyü düzeltmek için aynı filtre çekirdek yapısını kullanırken, adaptive smoothing yönteminde pencere büyüklüğü ve şekli yerel görüntü içeriğine bağlı olarak değişir.

Geleneksel düzeltme yönteminde olduğu gibi adaptive smoothing yönteminde de görüntüdeki gürültü azaltılır fakat geleneksel düzeltirmeden farklı olarak görüntüdeki kenarları bulanıklaştırmaz.

Şekil 6.6'da bu işlemin görüntü üzerindeki etkisi gösterilmektedir. Diğer uygulamalarda olduğu gibi bu filtreleme işleminde de kullanıcı istediği düzeltme değerini kendisi verebildiği gibi izleme çubuğunu kullanarak gerçekleştirebilir.



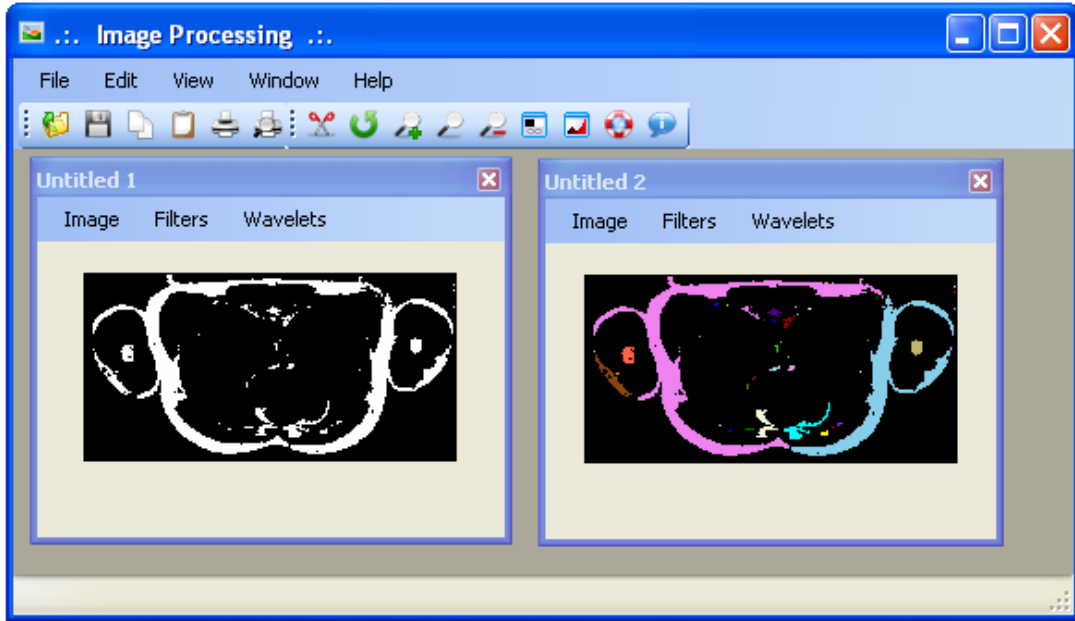
Şekil 6.6: Adaptive Smoothing

6.7 Bağlantılı Parça Etiketleme

Bağlantılı parça etiketleme bir çeşit görüntü analiz işlemi olarak kullanılır. Bu işlemin yapılması için ilk önce görüntünün taranması ve sonra ayrı piksel sınıflarının farklı tonlardaki gri renkleri ile ayırma işlemi yapılır. Görüntünün taranması işlemi bittikten sonra etiketler gruplanıp sıralanır ve her gruba tek bir etiket atanır. Son aşama olarak görüntü ikinci defa taranır ve her etiket eşdeğer gruptaki etiket ile yer değiştirilir.

Bağlantılı parça etiketleme ikili (binary) veya gri renkli görüntüler üzerine uygulanır.

Şekil 6.7’de bu işlem için bir örnek gösterilmektedir.



Şekil 6.7: Bağlantılı parça etiketleme

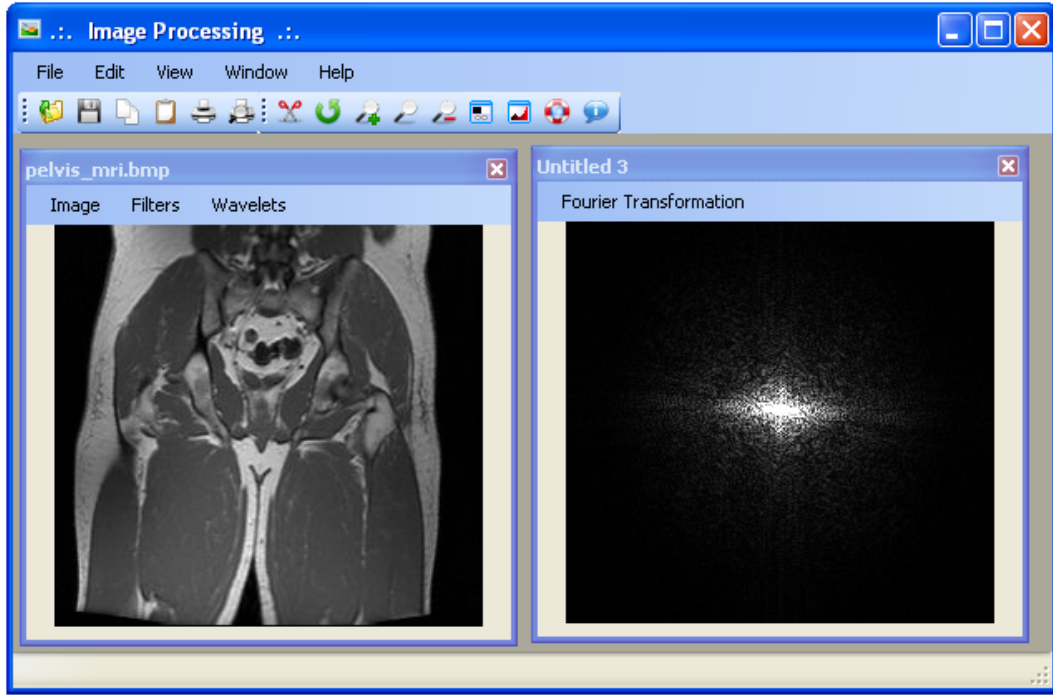
6.8 Fourier Dönüşümü

Fourier dönüşümü bir görüntüyü sinus ve cosinus parçalarına ayırmak için kullanılan önemli bir görüntü işleme aracıdır.

Girdi olarak kullanılan görüntü değerleri *gerçel* (real) uzaydan olmasına rağmen, bu dönüşüm işlemi sonucunda elde edilen görüntü *frekans* veya *Fourier* uzayında görüntülenir. Fourier uzayındaki görüntüde her bir nokta gerçel uzaydaki frekans değerini gösterir.

Fourier dönüşümü görüntü analizi, görüntünün filtrelenmesi veya sıkıştırılması gibi büyük uygulamalarda kullanılır.

Şekil 6.8’de bu uygulamaya ilişkin bir örnek gösterilmektedir.



Şekil 6.8: Fourier Dönüşümü

6.9 Morfoloji

Morfolojiksel görüntü işleme bir görüntünün yapısal element olarak adlandırılan küçük bir şekil veya şablon ile taranmasına dayalı bir işlemdir. Yapısal element görüntü üzerindeki mümkün olan bütün bölgelere yerleştirilip görüntünün içeriğine ve yapısal elemente bağlı olarak belirli bir işlem (kesişim, birleşim gibi) gerçekleştirilir.

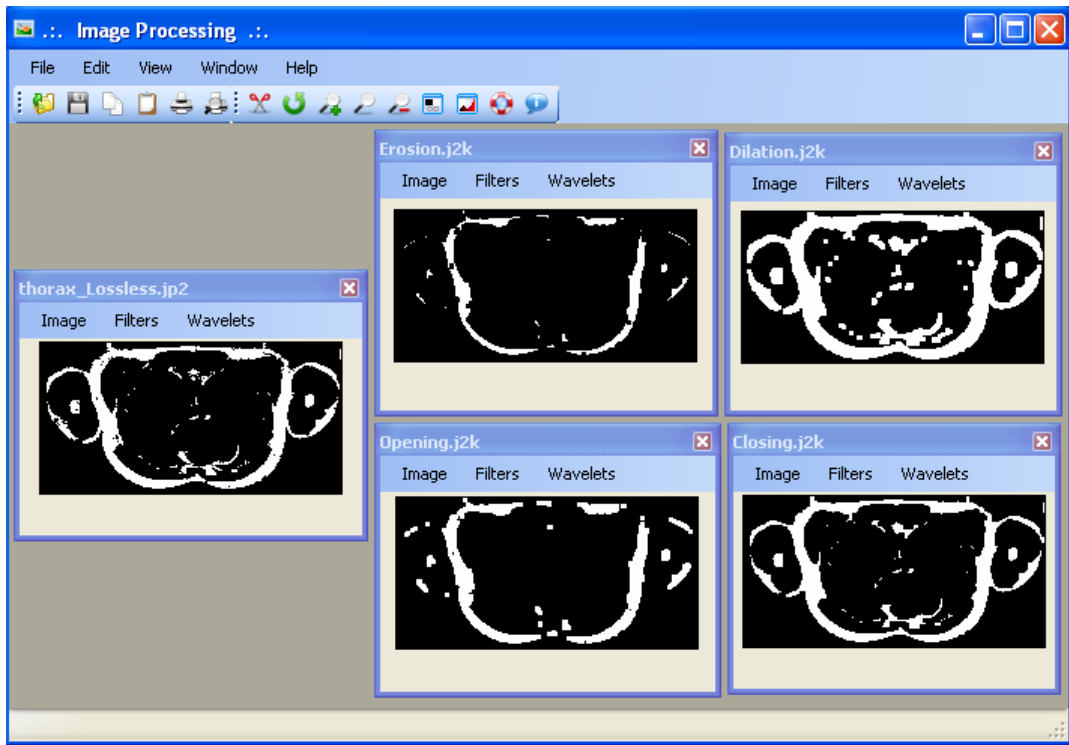
Morfoloji işlemleri girdi olarak bir yapısal element ve ikili (binary) görüntü alır ve bir küme işlemi (kesişim, birleşim, tersini alma veya kapsama) uygulanır. En çok bilinen morfolojik işlemler Eritme, Genleşme, Açma, Kapatma, İnceltme ve Kalınlaştırma işlemleridir. Şekil 6.9 farklı morfolojik işlemlerin görüntü üzerindeki etkilerini göstermektedir.

Görüntüdeki gibi en çok bilinen morfolojik işlemler; Eritme, genleşme, açma, kapatma, inceltme ve kalınlaştırma işlemleridir.

Eritme (veya aşındırma) işlemi nesnelere daha küçük yapar ve tek bir nesneyi çeşitli nesnelere ayırır. Bu işlem tipik olarak ikili (binary) veya gri renkli görüntülere

uygulanır. Bir görüntü üzerindeki en temel etkisi nesnenin kenarlarını aşındırmasıdır. Bu işlem için yapısal element ve aşındırılacak olan görüntü olmak üzere iki girdi kullanılır.

Genleşme işlemi görüntüyü daha geniş ve büyük yapar ayrıca birden fazla nesneyi tek bir nesne gibi birleştirilmiş görüntü etkisi oluşturabilir. Bu işlemde nesneye değen arka plandaki her piksel nesneye ait bir piksel olarak yorumlanır. Bu nedenle en basit etkisi görüntüdeki nesnenin kenarlarının kalınlaştırılmış olmasıdır.



Şekil 6.9: Morfolojik işlemler

Bu işlem için yapısal element ve genişleştirilecek olan görüntü olmak üzere iki girdi kullanılır.

Açma (opening) işlemi önce eritme sonra da genişirme işlemi yapılarak gerçekleştirilir. Eritme işleminde olduğu gibi açma işleminin de görüntü üzerindeki en basit etkisi nesnenin kenarlarındaki bazı piksellerin kaldırılmış olmasıdır. Fakat bu yöntem eritme tekniğinden daha yıkıcıdır. Diğer morfolojik işlemlerde olduğu gibi yapısal element ile görüntü üzerinde bir küme işlemi gerçekleştirilir. Bu işlem ile nesne

üzerinde yapısal element ile aynı olan bölgelerin korunması ve diğer alanların görüntüden kaldırılması sağlanmış olmaktadır.

Kapatma (closing) işlemi önce genişletme sonra eritme işlemi yapılarak gerçekleştirilir. Bu işlem genişletme işleminde olduğu gibi görüntüdeki nesnenin kenarlarını kalınlaştırır fakat şekildeki orjinal kenarlar üzerinde daha az yıkıcı etkisi vardır. Diğer morfolojik işlemlerde olduğu gibi bir yapısal element kullanılır. Böylelikle, sonuçta elde edilen görüntü, orjinal görüntüdeki nesnenin arka tarafında bulunan bölgeler yapısal element ile benzer şekle sahip ise aynen korunup, diğer bölgelerin ise görüntüden kaldırılması ile oluşan etkiye sahiptir.

İnceltme (thinning), eritme veya açma işlemlerinde olduğu gibi nesnenin seçilen yerlerinin ikili (binary) görüntüden kaldırılması ile gerçekleştirilen bir işlemdir. Bu işlem çeşitli uygulamalar için özellikle de iskeletleştirme amaçlı kullanılabilir.

İnceltme, ikili (binary) görüntülere uygulanır ve sonuçta yine ikili (binary) görüntü elde edilir.

Kalınlaştırma (thickening) diğer bir morfolojik işlem olup genişletme ve kapatma işlemlerinde olduğu gibi ikili görüntüdeki nesnenin seçilen bölgelerinin kalınlaştırılması için kullanılır. Bu işlemde ikili görüntülere uygulanıp yine sonuçta ikili görüntü elde edilir.

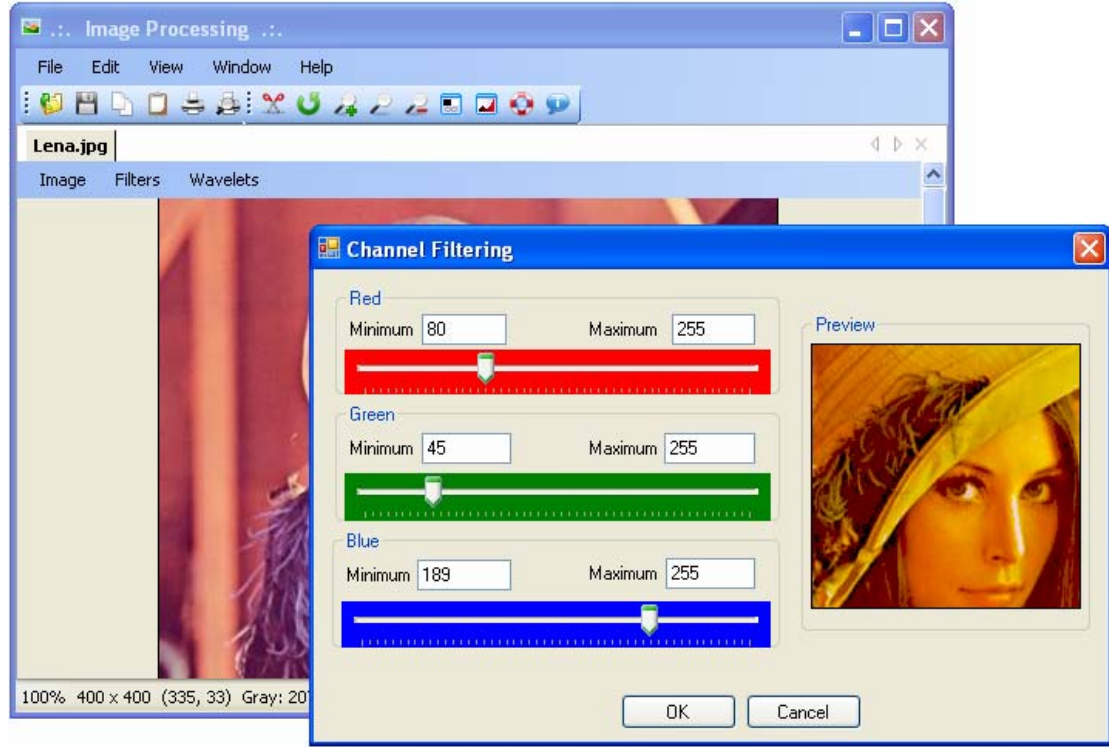
Diğer morfolojik işlemlere benzer şekilde kalınlaştırma işlemi içinde yapısal element kullanılır.

6.10 Renk Filtreleme

Renk filtreleri görüntüdeki istenilen özelliklerin belirginleştirilmesini kolaylaştırır. Kanal filtreleme, gri renk yapma ve görüntünün kırmızı, yeşil, mavi, cam göbeği, sarı, eflatun renklerine boyanması işlemlerinin yapılabildiği üç farklı bölüm bu menü için gerçekleştirilmiştir.

Kanal filtreleme formu üzerindeki önizleme kısmı, kullanıcıların filtrelenecek üzere bir kanal seçmesi ve o kanalın değerinin tanımlanması için oldukça yararlıdır.

Görüntü gri renk tonlarında boyanabilir veya kullanıcı görüntüyü sarı, eflatun gibi farklı renklere boyayabilir. Şekil 6.10'da renk filtreleme örneği gösterilmektedir.



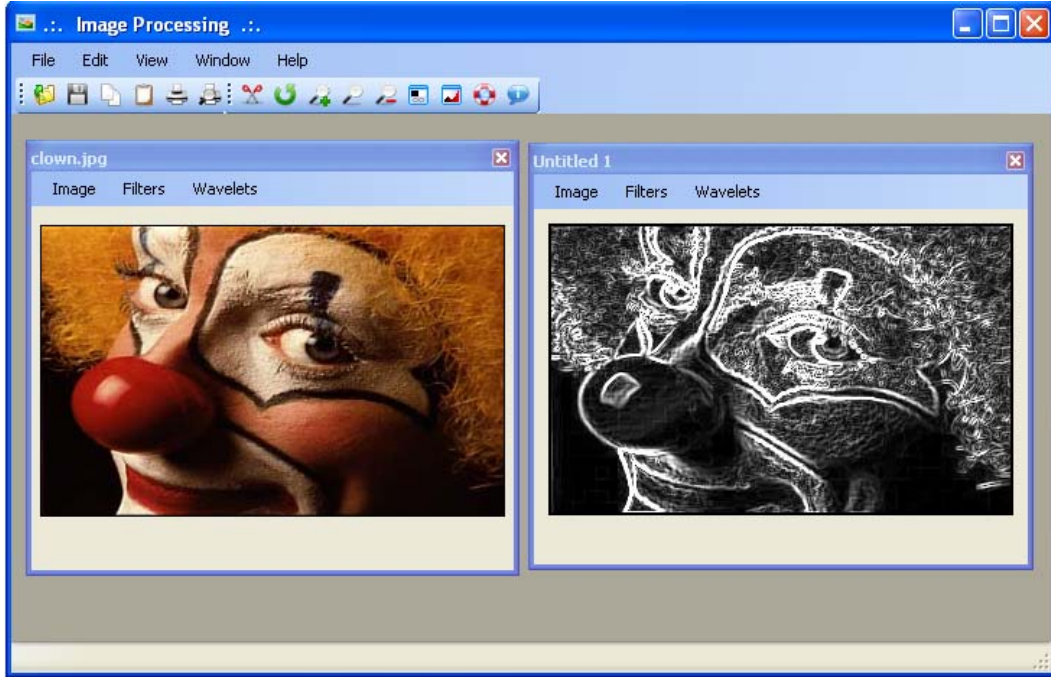
Şekil 6.10: Renk Filtreleme

6.11 Kenar Belirginleştiriciler

Bir görüntüdeki kenarların belirginleştirilmesindeki amaç resimdeki şekillerin kenarlarının kararlaştırılması ve siyah arka plan üzerinde beyaz (veya tam tersi renklendirme) ile renklendirilerek sonuç görüntünün çizilmesini sağlamaktır. Geliştirilen yazılımda en çok bilinen kenar belirginleştiriciler olan Sobel ve Canny Kenar Belirginleştiriciler üzerinde durulmuştur.

Sobel Kenar Belirginleştirici: Girdi olarak kullanılan gri renk tonlu görüntünün her noktasında yaklaşık eğiklik değerini bulmak için kullanılan tekniktir. Sobel Kenar Belirginleştirici hem x boyutundaki hem de y boyutundaki eğikliği belirlemek için bir çift 3×3 matris (maske, yapısal element) kullanır. Matris genellikle görüntü boyutundan

daha küçük olacak şekilde seçilir ve görüntü üzerinde dolaştırılarak işlem gerçekleştirilir. Şekil 6.11'deki görüntü Sobel Kenar Belirginleştirici uygulaması için bir örnektir. İlk resim orijinal görüntü olup bu resim üzerindeki 'image' menüsünden ilgili kenar belirleyici seçilebilir.



Şekil 6.11: Sobel kenar belirginleştirici

Canny Kenar Belirginleştirici: Canny Kenar Belirginleştirici algoritması bilinen en iyileştirilmiş kenar belirleme algoritmasıdır. Canny'nin ilgi alanı kenar belirginleştiricilerin geliştirilmesi üzerinedir. Bu nedenle yayınlarda fikirlerini amaçlarını ve metodlarını oldukça başarılı bir şekilde belirtmiştir.

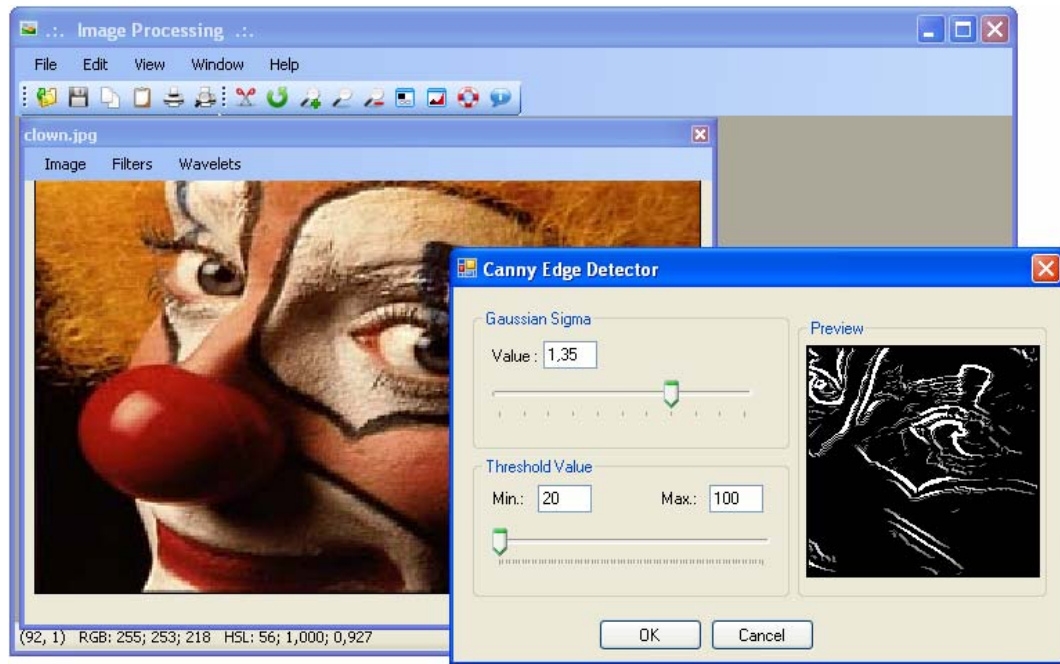
Canny'nin kriterlerine göre Canny Kenar Belirginleştirme ile ilk önce görüntüdeki gürültülerin ortadan kaldırılması için düzleştirme (smoothing) işlemi yapılır. Daha sonra bazı bölgelerin belirginleştirilmesi suretiyle görüntüdeki eğiklikler bulunur.

Şekil 6.12 bu işlem için bir örnek görüntüyü göstermektedir.

6.12 Convolution Filtreler

Gerçekleştirilen uygulamada convolution işlemi ile elde edilen aşağıdaki filtre teknikleri yer almaktadır.

Gaussian Blur: Bu filtre görüntüdeki detayların ve gürültünün kaldırılması ile gerçekleştirilen genel amaçlı bir bulanıklaştırma filtresidir.



Şekil 6.12: Canny kenar belirginleştirici

Bu teknik bir görüntünün farklı katmanlarındaki detaylar üzerinde iyi tanımlanmış bir etkiye sahip olması sebebi ile diğer bulanıklaştırma tekniklerinden ayrılır. Detayın seviyesi küçüldükçe filtre daha az geçişe izin verir. Diğer filtrelerde miktar göreceli olarak değişmektedir. İyi tanımlanmış kararlı frekans yanıtlarına sahip olduğu gibi, Gaussian fonksiyonunun belirgin özellikleri sayesinde diğer benzer filtre çeşitlerine göre daha fazla bulanıklık daha hızlı bir şekilde uygulanabilir.

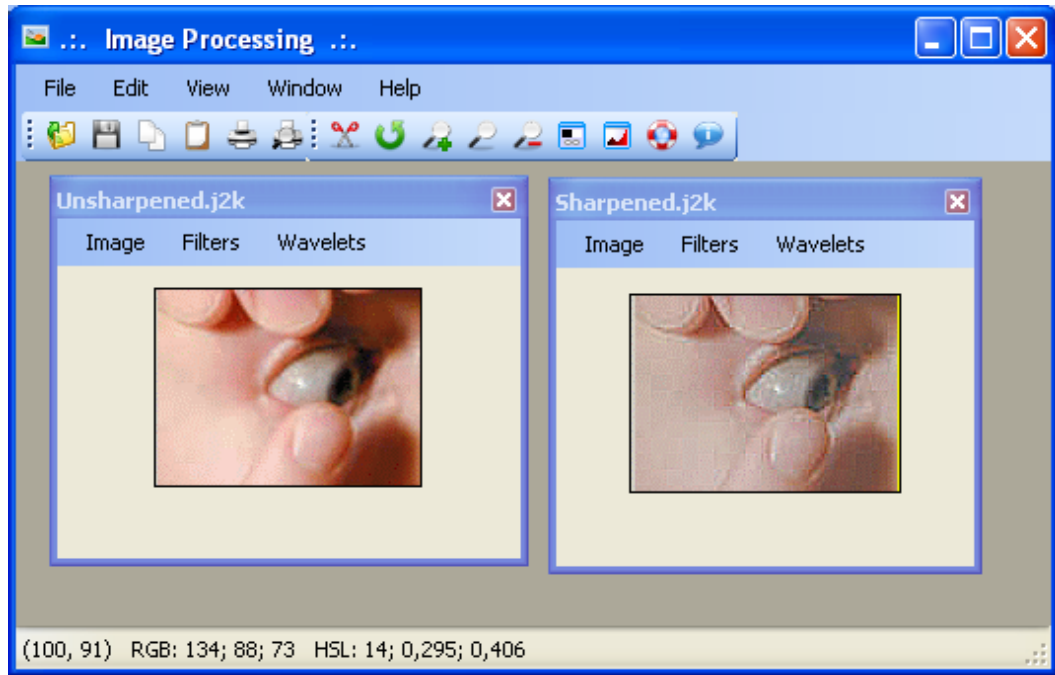
Blur: Bir görüntüdeki bilinen örnek bulanıklık etkisini göstermek amacıyla gerçekleştirilen bir algoritmadır.

Mean: Bu filtre, görüntüdeki bir piksel ile komşuluğunda bulunan pikseller arasındaki yoğunluk değişkenlik miktarının azaltılması ile düzeltilmiş görüntü üzerinde basit, sezgisel ve kolayca kodlanabilen bir filtredir. Genellikle görüntüdeki gürültünün azaltılması için kullanılır. Diğer filtre teknikleri içinde en çok bilinen tekniktir.

Mean filtreleme tekniği görüntüdeki her piksel değerinin, komşuluğunda bulunan piksellerin mean (ortalama) değeri ile yer değiştirilerek gerçekleştirilir. Bu işlem, komşulukları tarafından görüntülenmeyen piksel değerlerinin görüntüden kaldırılması etkisini verir. Diğer filtrelerde olduğu gibi bir maske veya yapısal element kullanılarak mean değeri hesaplanır.

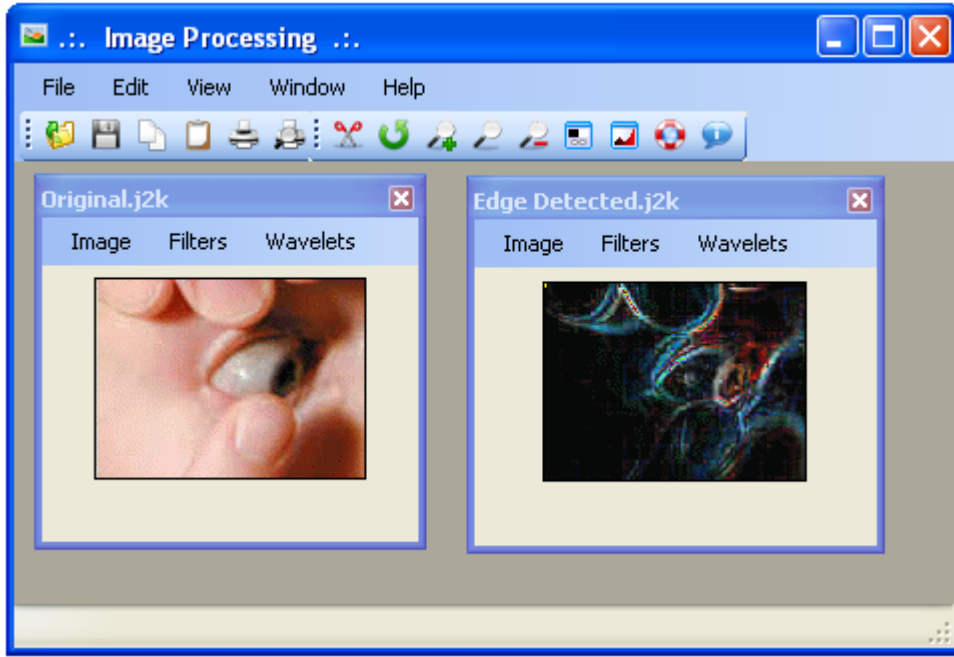
Keskinleştirme: Şekil 6.13 bir görüntüye ait keskinleştirme algoritmasının uygulanması ile elde edilen sonucu göstermektedir.

Genel olarak, keskinleştirme bir görüntüdeki küçük ayrıntıların daha belirgin hale getirilmesi için uygulanır. Keskinleştirilmiş görüntü ile orjinal resim aşağıdaki örnekte görülmektedir.



Şekil 6.13: Keskinleştirme

Kenarlar: Bu işlemin amacı gölgelerden dolayı oluşan yoğunluk ve renk değişimlerini içeren görüntülerdeki farklı nesnelere kenarlarının belirginleştirilmesidir. Şekil 6.13, gözüne kontak lens yerleştiren birisini göstermekte olup kenar belirginleştirme işleminin uygulanması sonucu elde edilen resmi belirtmektedir. Bu resimde parmakların kenarlarını, gözdeki iris bölümünü, burun üzerindeki gölgeyi kolaylıkla görmek mümkündür.



Şekil 6.14: Kenarlar

6.13 İki Görüntünün Kombinasyonu

Basit görüntü işleme teknikleri ile görüntüdeki sadece piksellerin fiziksel yerleri değiştirilirken, görüntü üzerinde yapılan işlemlere bir diğer alternatif yöntem ise aritmetik işlemlerin kullanılmasıdır. Bir tam sayı değerini piksel değerlerine eklemek çıkarmak veya görüntüdeki pikseller üzerinde sabit bir değer ile çarpma, bölme işlemleri yapılabilir. Aşağıda bu işlemlere ilişkin açıklamalar yer almaktadır.

Toplama: Bu işlem ile girdi olarak genellikle iki görüntü alınıp toplama işlemi gerçekleştirildikten sonra çıktı olarak bir görüntü elde edilir. Elde edilen görüntü girdi

olarak kullanılan ilk iki görüntü ile aynıdır. İki den fazla görüntünün toplanması işlemide gerçekleştirilebilir.

Çıkarma: Bu işlem girdi olarak iki görüntü alıp çıktı olarak tek bir görüntü oluşturur. Elde edilen görüntü ilk iki görüntü ile aynı boyuttadır.

Görüntülerde çıkarma işleminin en başarılı ve yararlı kullanım alanı tıbbi görüntüleme alanıdır.

Kesişim: ADD operatorü kullanılarak gerçekleştirilen kesişim işleminde, her iki girdi değeri 1 ise çıktı olarak 1 elde edilir iken, girdilerden birisinin değerinin sıfır olması durumunda çıktı da sıfır olacaktır. Kullanılacak olan görüntüler eşit büyüklükte olmalıdır.

Birleşim: OR operatorü kullanılarak gerçekleştirilen bu işlemde ise girdi olarak alınan iki değerden her ikisinde sıfır ise sonuç sıfır olacaktır. En az bir tane girdi değerinin 1 olması durumunda sonuç 1 olacaktır. Kullanılacak olan her iki görüntünün de eşit büyüklükte olması gerekmektedir.

Bu filtre tekniklerine ek olarak aşağıdaki yapılarda geliştirilmiştir;

Surface Voxel: Kullanıcılar üç boyutlu görüntüdeki en küçük birim olan 'voxel' değerini değiştirip sonuçları görüntü üzerinde görebilir.

Üç Boyutlu Görüntüleme: Bu pencere yardımı ile kullanıcılar üç boyutlu bir görüntünün her boyutunu görebilir.

Ayrıca, Döndürme, Ölçekleme, Kesme, Klonlama, Farklı Kaydetme, Yazdırma, Kopyalama ve Yapıştırma işlemleri bu yazılım kullanılarak gerçekleştirilebilmektedir.

Farklı kaydetme işleminin önemli bir özelliği ise üç boyutlu bir görüntünün J2K biçiminde kaydedilebilmesidir. Bu işlem üç boyutlu görüntünün her çerçevesinin, dosya ismi ve çerçeve numarası kullanılarak otomatik bir şekilde kaydedilmesi ile sağlanır.

Bu uygulama, kullanıcı kolaylığı sađlayan bir uygulama olmasının yanısıra, Görüntü (View) menüsü yardımı ile kullanıcılar tarafından arayüz üzerinde istenildiđi gibi bazı deđişikliklerin yapılmasına da olanak sađlar.

7. SONUÇLAR

İnanıyoruz ki, geliştirilen görüntü işleme aracı hem JPEG2000 standardı hem de çok boyutlu tıbbi görüntüler için oldukça yararlıdır.

Çeşitli görüntü işleme işlemlerinin orjinal görüntüler üzerinde gerçekleştirilmesinden sonra elde edilen sonuçlar ile dalgacık dönüşümü uygulanan görüntüler üzerinde gerçekleştirildikten sonra elde edilen sonuçlar birbirine oldukça benzerlik göstermektedir.

Geliştirilen yazılım ile iki boyutlu (2D), zaman bilgisi ile iki boyutlu (2D in time), üç boyutlu (3D) ve hatta zaman bilgisi ile üç boyutlu (3D in time) görüntüler üzerinde filtreleme ve sıkıştırma işlemleri yapılabilir.

Uygulamalar için Microsoft Visual Studio 2005 teknolojisi çok geniş .NET Framework tabanlı çözümler ve çok yönlü araçlar sunar, ayrıca uygulama geliştirme karmaşıklığını azaltır.

Eminiz ki, bu yazılım görüntü işleme teknikleri ve filtreler üzerinde çalışacak kişiler için oldukça ilgi çekici olacaktır.

İleride geliştirilecek çalışma ise kenar belirleme yordamları ile ilgili problemlerin tanımlanması ve çözümü olarak planlanıp, özellikle tıbbi görüntüler üzerindeki problemlere önem verilmesi düşünülmektedir.

Geliştirilmesi düşünülen diğer bir çalışma ise, ters dalgacık dönüşümü ve hareket belirleme özelliklerinin bu yazılıma eklenmesi olarak belirtilebilir. Ayrıca, hesaplamalardaki karışıklığı azaltmak için gerekli uygun yaklaşımların belirlenmesi ve kodlanması çalışmalarının yapılmasıda planlanmaktadır.

KAYNAKLAR

- Athanassios, S., Charilaos, C. and Touradj, E. (2001) “*The JPEG2000 Still Image Compression Standard*”, **IEEE Signal Processing Magazine**, 45s
- Athanassios, S., Charilaos, C. and Touradj E. (2000) “*The JPEG2000 Still Image Coding System*”, **IEEE Signal Processing Magazine**, 55s
- Boliek, M., Christopoulos, C. and Majani, E. (2000) JPEG2000 “JPEG2000 Image Coding System”, <http://www.upatras.gr/ieee/skodras/pubs/ans-b17.pdf>
- David, S., Taubman, M. and Marcellin, W. (2002) “*Image Compression Fundamentals, Standards And Practice*”, **Kluwer Academic Publishers**, Boston, s.68-125
- Frey, F. and Suesstrunk, S. (2000) “Digital photography - How long will it last?”, **IEEE Int. Symp. Circuits and Systems (ISCAS 2000)**, V. Geneva, Switzerland, 27s
- Gaetano, I. (2004) “A tutorial – JPEG2000”, <http://www.dmi.unict.it/~impoco/> (10/10/2005)
- Goceri, E. and Boyen, H. (2006) “Classical Image Processing”, **Xios Hogeschool Bölüm Seminerleri**, Hasselt, 62s.
- Goceri, E. and Boyen, H. (2005) “Wavelets”, **Xios Hogeschool Bölüm Seminerleri**, Hasselt, 48s.
- Karen, L. (2005) "The JPEG2000 Standard", <http://www.witi.cs.uni-magdeburg.de> (30/09/2005)
- Marcellin, W., Gormish, M., Bilgin, A., and Boliek, M. (2000) “An overview of JPEG2000”, **IEEE Data Compression Conf.**, Snowbird
- Michael, L., Hilton, D., and Jawerth, A. (1994) “Compression Still and Moving with Wavelets”, **Springer -Verlag New York**, s.125-144
- Parulski, K., and Rabbani, M. (2000) “The continuing evolution of digital cameras and digital photography systems”, **IEEE Int. Symp. Circuits Syst.** Geneva, Switzerland
- Walker, J. (1999) “*Wavelets and their Scientific Applications*”, **CRC Press, New York**, s.25-89
- WEB_1. (2006) JPEG topluluğunun JPEG2000 resmi websitesi <http://www.jpeg.org/jpeg2000/j2kpart11.html> (03/03/2006)
- WEB_2. (2006) Amara'nin wavelet web sitesi <http://www.amara.com/current/> (02/02/2006)
- WEB_3. (2006) CMP (United Business Media)'nin web sitesi <http://www.seyboldreports.com/SRIP/wavelet/> (02/02/2006)

WEB_4. (2006) Mikrossoft'un MSDN resmi web sitesi <http://msdn2.microsoft.com> (15/02/2006)

WEB_5. (2001) Robi Polikar'ın web sitesi <http://users.rowan.edu/~polikar/wavelets/wttutorial.html> (20/02/2006)

WEB_6. (2004) Bob Fisher, Simon Perkins, Ashley Walker ve Erik Wolfart'in web sitesi, <http://www.cee.hw.ac.uk/hipr/html/glossary.html> (24/04/2006)

ÖZGEÇMİŞ

Evgin GÖÇERİ, 1979 yılında İzmir’de doğdu. İlkokul ve ortaokulu İzmir’de tamamlayıp yine aynı şehirde bulunan Bornova Mustafa Kemal Lisesini bitirdikten sonra Ege Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü’nden 2002 yılında mezun oldu. Üniversite eğitimi yıllarında çalıştığı Bornova Yazılım Geliştirme Merkezi’nden edindiği tecrübeler ve özellikle internet teknolojileri alanındaki çalışmalarından dolayı Cyber Matrix (U.S.A) firmasından aldığı teklife rağmen çalışma hayatını İzmir’de devam ettirip, 2001 yılında başladığı Pınar Süt Mamulleri Gıda Sanayi A.Ş.’de çeşitli projelerde görev aldı. 2003 yılında Aydınel Tekstil A.Ş.’de sistem mühendisi olarak çalışmalarına devam edip, Ocak 2004’te Pamukkale Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü’nde Araştırma Görevlisi olarak çalışmaya başladı. Eylül 2004’te Pamukkale Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Ana Bilim Dalı’nda yüksek lisans eğitimine başlayan Evgin GÖÇERİ, Avrupa Birliği tarafından desteklenen Socrates-Erasmus değişim programı ile Eylül 2005’te yüksek lisans tez projesini gerçekleştirmek amacıyla gittiği Belçika’da araştırmalarını tamamlayıp yurda dönmüştür.