

EGE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

(YÜKSEK LİSANS TEZİ)

**ALÇAK ÇÖZÜNÜRLÜKLÜ FOTOĞRAFLARIN
GÖRÜNTÜLENMESİ VE BUNLARIN OPTİMİZASYONU İLE
İLGİLİ BİR ÇALIŞMA**

NİL KARACA

Uluslararası Bilgisayar Anabilim Dalı

Bilim Dalı Kodu : 619.03.03

Sunuş Tarihi : 25.10.2007

Tez Danışmanı : Prof. Dr. Aydın Öztürk

BORNOVA - İZMİR

III

Nil KARACA tarafından YÜKSEK LİSANS TEZİ olarak sunulan “**Alçak Çözünürlüklü Fotoğrafların Görüntülenmesi ve bunların Optimizasyonu ile ilgili Bir Çalışma**” başlıklı bu çalışma E.Ü. Fen Bilimleri Enstitüsü Eğitim ve Öğretim Yönergesi’nin ilgili hükümleri uyarınca tarafımızdan değerlendirilerek savunmaya değer bulunmuş ve 25.10.2007 tarihinde yapılan tez savunma sınavında aday oybirliği/oyçokluğu ile başarılı bulunmuştur.

Jüri Üyeleri:

Jüri Başkanı : Prof. Dr. Aydın Öztürk

Raportör Üye: Yrd. Doç. Dr. Cengiz Güngör

Üye : Prof. Dr. Serdar Korukoğlu

İmza:

.....

.....

.....

ÖZET**ALÇAK ÇÖZÜNÜRLÜKLÜ FOTOĞRAFLARIN
GÖRÜNTÜLENMESİ VE BUNLARIN OPTİMİZASYONU İLE
İLGİLİ BİR ÇALIŞMA**

KARACA, Nil

Yüksek Lisans Tezi, Uluslararası Bilgisayar Enstitüsü

Tez Yöneticisi: Prof. Dr. Aydın ÖZTÜRK

Ekim 2007, 95 sayfa

Günümüzde teknoloji sürekli ilerlemektedir. Sayısal teknoloji ilerledikçe görüntü işleme konusu da ayrı bir önem kazanmaktadır. Bilgisayar grafikleri son yıllarda çok geniş bir alana yayılmıştır ve gelişmesi de büyük bir hızla devam etmektedir. Bir görüntü üzerinde işlem yapmak birçok alanı ilgilendirmektedir. Sayısal görüntüler üzerinde işlem yapmak, mimarlık, coğrafik sistemler, reklamcılık, yayıncılık alanları örnek olarak verebilir.

Verilen bir fotoğrafın daha alçak çözünürlüklü düzeylerinde görüntülenmesi ile ilgili bir çok grafik uygulaması vardır. Bu tez çalışmasında; seçilen bir resmin törenlerde stadyumda değişik renklerden oluşan levhalarla gösterimi üzerinde durulmuştur. Bu çalışma bilgisayar teknikleri kullanılarak bilgisayar ortamında geliştirilmiştir. Görüntüler “piksel” düzeyinde işlenip, görüntülerin elde edilmesinde bilgisayar grafik tekniklerinden yararlanılmıştır. Elde edilen yeni görüntüler üzerinde görüntü karşılaştırma metrikleri kullanılmış ve sonuçlar tartışılmıştır. Program kullanıcı seviyesinde Visual Studio .NET 2003 de C# dili kullanılarak Windows sistemlerde çalışabilecek şekilde hazırlanmıştır.

Anahtar kelimeler: Bilgisayar grafikleri, görüntü işleme, piksel, çözünürlük, renk azaltımı

VII

ABSTRACT

A STUDY OF RENDERING LOW RESOLUTION IMAGES AND OPTIMIZATION OF THESE LOW RESOLUTION IMAGES

KARACA, Nil

MSc. in International Computer Institute

Supervisor: Prof. Dr. Aydın ÖZTÜRK

October 2007, 95 pages

At present technology is continuously improving. As a result of this image processing field gains significant importance. In recent years computer graphics have been in use in widespread areas and it continues to evolve with rapid speed. Processing on an image suggests many solutions for various fields. Example of image processing use areas expand from architectural and advertisement purposes, to geographical and publishing systems.

There exist many applications that involve a given image to be rendered in low resolution format. In the context of this study pictures, composed of colored panels that are shown during formal ceremonies at stadiums, were emphasized. In this study with the use of computer techniques, application was implemented at computer environment. Images were processed in pixel level and we benefited from computer graphics techniques to obtain images. Image comparison metrics were used on currently obtained new images and results were argued. Application was developed at Visual Studio .NET 2003 programming environment with C# programming language, which executes in Windows systems.

Keywords: Computer graphics, image processing, pixel and resolution,color reduction

IX

TEŞEKKÜR

Öncelikle bu tez konusu üzerinde bana çalışma imkanı sunan tez danışmanım Prof. Dr. Aydın ÖZTÜRK'e çalışma süresince deneyimi, bilgisi ve önerileriyle araştırma ve geliştirmeyi yönlendirmesi ve sağladığı kaynaklarla destek olmasından dolayı teşekkürü bir borç bilirim.

Çalışma sırasında gereksinim duyduğum görüntü işlem uygulamalarında bana yardımcı olan ve çalışmalarım hakkında değerli görüşlerini esirgemeyen Yrd. Doç. Dr. Cengiz GÜNGÖR'e teşekkürlerimi sunarım.

Bu çalışmamı daima yanımda olan ve manevi desteklerini esirgemeyen aileme; annem ve ağbim'e adıyorum.

İÇİNDEKİLER

ÖZET.....	V
ABSTRACT	VII
TEŞEKKÜR.....	IX
İÇİNDEKİLER.....	XI
ŞEKİLLER DİZİNİ	XIII
TABLolar DİZİNİ	XV
KISALTMALAR.....	XVI
1 GİRİŞ.....	1
2 TEMEL KAVRAMLAR.....	6
2.1 BİLGİSAYAR GRAFİKLERİ	6
2.2 PİKSELLER VE GÖRÜNTÜ ARA BELLEK	6
2.3 RESİM TIPLERİ (IMAGE TYPES)	8
2.3.1 Gerçek Renk (<i>true color</i>)	8
2.3.2 Gri Seviye (<i>gray scale</i>)	8
2.3.3 İndeksli Renk (<i>Indexed</i>)	8
2.4 GÖRÜNTÜ HISTOGRAMI (HISTOGRAM).....	9
2.5 HISTOGRAM EŞİTLEME (HISTOGRAM EQUALIZATION)	9
2.6 HISTOGRAM BELİRLEME (HISTOGRAM SPECIFICATION) – YOĞUNLUK AYARLAMA (INTENSITY ADJUSTMENT)	10
2.7 GÖRÜNTÜ KARŞILAŞTIRMA METRİKLERİ	11
2.7.1 RMSE (<i>Root Mean Squared Error</i>)	12
2.7.2 PSNR (<i>Peak Signal-to-Noise Ratio</i>)	13
3 PROBLEMİN TANIMI VE İLGİLİ ÇALIŞMALAR.....	15
3.1 Renk Azaltma (<i>Color Reduction</i>).....	16
3.2 Ana Bileşenler Analizi (<i>Principal Components Analysis – PCA</i>)	18
4 GELİŞTİRİLEN YÖNTEMLER.....	20
4.1 REFERANS GÖRÜNTÜ ÜZERİNDE YAPILAN İŞLEMLER	20
4.2 RENK PALETLERİ.....	23
4.2.1 48, 24, 12 lik renk paletleri	23
4.2.2 Ana Bileşenler Analizi (<i>Principal Components Analysis</i>)... ..	24
4.2.3 Portrelerin PSNR değerleri.....	30

XII

4.3	PCA’IN UYGULANMASI SONUCU ELDE EDİLEN PALETLER İLE REFERANS GÖRÜNTÜ ÜZERİNDE YAPILAN İŞLEMLER.....	32
4.3.1	<i>Görüntü histogramları</i>	35
5	KULLANILAN METRİKLER	51
5.1	BLOKLARIN EŞLEŞTİRİLMESİNDE KULLANILAN METRİKLER	51
5.1.1	<i>En küçük kareler toplamı</i>	51
5.1.2	<i>En küçük mutlak değerler</i>	53
5.1.3	<i>En küçük ortalama değerler</i>	54
5.2	GÖRÜNTÜLERİN KARŞILAŞTIRILMASINDA KULLANILAN METRİKLER.....	55
5.2.1	<i>RMSE</i>	55
5.2.2	<i>PSNR</i>	57
6	SORUNLU BÖLGELERİN TEŞHİSİ	62
7	SONUÇ	64
8	KAYNAKLAR DİZİNİ	67
EKLER		69
	Ek1. UYGULAMANIN KULLANIMI	70
	Ek2. PROGRAM ÇIKTILARI.....	79
	Ek3. TÜRKÇE-İNGİLİZCE TERİMLER SÖZLÜĞÜ	93
ÖZGEÇMİŞ		95

XIII

ŞEKİLLER DİZİNİ

ŞEKİL 2.1: PİKSEL GÖRÜNTÜ (ANGEL EDWARD, 2006).....	7
ŞEKİL 2.2: BİR GÖRÜNTÜ VE HISTOGRAMI (MATLAB6.5 KÜTÜPHANESİ)	9
ŞEKİL 2.3: HISTOGRAM EŞİTLEME (MATLAB6.5 KÜTÜPHANESİ).....	10
ŞEKİL 2.4: HISTOGRAM BELİRLEME	11
ŞEKİL 2.5: PSNR	13
ŞEKİL 3.1: DÜZ PANEL (ANGEL EDWARD, 2006)	17
ŞEKİL 3.2: RENKLERİN OLUŞTURULMASI (HTTP://TR.WIKIPEDIA.ORG/WIKI/RGB,2007)..	17
ŞEKİL 4. 1: ORJİNAL RESİM	20
ŞEKİL 4. 2: REFERANS GÖRÜNTÜ	20
ŞEKİL 4. 3: REFERANS GÖRÜNTÜ	21
ŞEKİL 4.4: 48 LİK PALET VE METRİK UYGULAMA SONUÇ GÖRÜNTÜLERİ	22
ŞEKİL 4.5: 24 LİK PALET VE METRİK UYGULAMA SONUÇ GÖRÜNTÜLERİ	22
ŞEKİL 4.6: 12 LİK PALET VE METRİK UYGULAMA SONUÇ GÖRÜNTÜLERİ	23
ŞEKİL 4.7: RENK PALETLERİ.....	24
ŞEKİL 4.8: SİYAH VE BEYAZ RENKLERİN RGB DEĞERLERİ.....	24
ŞEKİL 4.9: ORJİNAL GÖRÜNTÜ	25
ŞEKİL 4.10: PCA SONUC RENK AZALTILARAK ELDE EDİLEN GÖRÜNTÜ.....	29
ŞEKİL 4.11: ORJİNAL VE PCA SONUCU ELDE EDİLEN GÖRÜNTÜLER	29
ŞEKİL 4.12: PCA SONUCU RENK PALETLERİ	30
ŞEKİL 4.13: ORJİNAL(PORTRE1) GÖRÜNTÜ VE PCA SONUCU GÖRÜNTÜ.....	31
ŞEKİL 4.14: ORJİNAL(PORTRE2) GÖRÜNTÜ VE PCA SONUCU GÖRÜNTÜ.....	31
ŞEKİL 4.15: ORJİNAL(PORTRE3) GÖRÜNTÜ VE PCA SONUCU GÖRÜNTÜ.....	31
ŞEKİL 4.16: (A) ORJİNAL GÖRÜNTÜ	32
ŞEKİL 4.16: (B) PCA SONUCU GÖRÜNTÜ	32
ŞEKİL 4.17: PCA ANALİZİNDE KULLANILAN REFERANS GÖRÜNTÜ	33
ŞEKİL 4.18: PCA SONRASI 48 LİK PALET VE METRİK UYGULAMA SONUÇ GÖRÜNTÜLERİ	33
ŞEKİL 4.19: PCA SONRASI 24'LİK PALET VE METRİK UYGULAMA SONUÇ GÖRÜNTÜLERİ	34
ŞEKİL 4.20: PCA SONRASI 12'LİK PALET VE METRİK UYGULAMA SONUÇ GÖRÜNTÜLERİ	34
ŞEKİL 4.21: REFERANS GÖRÜNTÜ VE 48'LİK PALET VE METRİK UYGULAMA SONUÇ GÖRÜNTÜSÜ	35
ŞEKİL 4.22: REFERANS GÖRÜNTÜ	36
ŞEKİL 4.23: REFERANS GÖRÜNTÜNÜN KIRMIZI RENK H İSTOGRAMI.....	36
ŞEKİL 4.24: REFERANS GÖRÜNTÜNÜN YEŞİL RENK H İSTOGRAMI	37
ŞEKİL 4.25: REFERANS GÖRÜNTÜNÜN MAVİ RENK H İSTOGRAMI	37
ŞEKİL 4.26: EN KÜÇÜK KARELER TOPLAMI METRİĞİ VE PALET 48.....	38
ŞEKİL 4.27: HISTOGRAMI	39
ŞEKİL 4.28: EN KÜÇÜK MUTLAK DEĞERLER METRİĞİ VE PALET 48	40
ŞEKİL 4.29: HISTOGRAMI.....	40
ŞEKİL 4.30: EN KÜÇÜK ORTALAMA DEĞERLER METRİĞİ VE PALET 48.....	41
ŞEKİL 4.31: HISTOGRAM	41

XIV

ŞEKİL 4.32: EN KÜÇÜK KARELER TOPLAMI METRİĞİ VE PALET 24.....	42
ŞEKİL 4.33: EN KÜÇÜK ORTALAMA DEĞERLER METRİĞİ VE PALET 48.....	42
ŞEKİL 4.34: EN KÜÇÜK MUTLAK DEĞERLER METRİĞİ VE PALET 24.....	43
ŞEKİL 4.35: EN KÜÇÜK MUTLAK DEĞERLER METRİĞİ VE PALET 24.....	43
ŞEKİL 4.36: ORTALAMA DEĞERLER METRİĞİ VE PALET 24.....	44
ŞEKİL 4.37: ORTALAMA DEĞERLER METRİĞİ VE PALET 24.....	44
ŞEKİL 4.38 EN KÜÇÜK KARELER TOPLAMI METRİĞİ VE PALET 12.....	45
ŞEKİL 4.39 HISTOGRAM.....	45
ŞEKİL 4.40: EN KÜÇÜK MUTLAK DEĞERLER METRİĞİ VE PALET 12.....	46
ŞEKİL 4.41: HISTOGRAM.....	46
ŞEKİL 4.42: ORTALAMA DEĞERLER METRİĞİ VE PALET 12.....	47
ŞEKİL 4.43: HISTOGRAM.....	47
ŞEKİL 4.44 EN KÜÇÜK KARELER TOPLAMI METRİĞİ VE PALET 32.....	48
ŞEKİL 4.45 HISTOGRAM.....	48
ŞEKİL 4.46: EN KÜÇÜK MUTLAK DEĞERLER METRİĞİ VE PALET 32.....	49
ŞEKİL 4.47: HISTOGRAM.....	49
ŞEKİL 4.48: ORTALAMA DEĞERLER METRİĞİ VE PALET 32.....	50
ŞEKİL 4.49: HISTOGRAM.....	50
ŞEKİL 5. 1: PİKSEL DEĞERLERİN PALETLE İŞLEME ALINMASI.....	52
ŞEKİL 5. 2: METRİK HESAPLAMASI.....	52
ŞEKİL EK1.1: UYGULAMA EKRANI.....	70
ŞEKİL EK1.2: .NET’DE TASARLANAN UYGULAMA EKRANI.....	71
ŞEKİL EK1.3: DOSYA SEÇME EKRANI.....	72
ŞEKİL EK1.4: DOSYA SEÇME EKRANI.....	72
ŞEKİL EK1.5: METRİK UYGULAMALARI SONUNDAKİ GÖRÜNTÜLER.....	73
ŞEKİL EK1.6:RENK DEĞİŞTİRME EKRANI.....	75
ŞEKİL EK1.7:RENK DEĞİŞTİRME EKRANI VE PALET.....	76
ŞEKİL EK1.8:GÖREV KARTI BASIMI.....	76
ŞEKİL EK1.9:GÖREV KARTI BASIMI.....	77
ŞEKİL EK1.10:ÖĞRENCİ BİLGİLERİ TABLOSU.....	77
ŞEKİL EK1.11:GÖRÜNTÜ BİLGİLERİ TABLOSU.....	78
ŞEKİL EK1.12:PALET BİLGİLERİ TABLOSU.....	78
ŞEKİL EK1.13:TABLOLAR.....	78

TABLOLAR DİZİNİ

TABLO 4.1 ORJINAL GÖRÜNTÜNÜN RGB MATRİSİ(TOPLAM N=21600 PİKSEL)	25
TABLO 4.2 PC KATSAYILARI	26
TABLO 4.3 PC DEĞERLERİNİN VARYANSLARI	26
TABLO 4.4 ANA BİLEŞENLERE İLİŞKİN VARIYANSLARIN YÜZDELERİ	26
TABLO 4.5 PC DÖNÜŞÜM MATRİSİ	27
TABLO 4.6 ÜÇÜNCÜ BOYUT SIFIR	27
TABLO 4.7 RENK AZALTILMIŞ MATRİS	28
TABLO 4.8 PORTRELER PCA SONUCU PSNR DEĞERLERİ	31
TABLO 4.9 PORTRELER PCA SONUCU PSNR DEĞERLERİ	32
TABLO 4.10 HISTOGRAM ALGORİTMASI	38
TABLO 5.1 PALET 48'E GÖRE RSME DEĞERLERİ	56
TABLO 5.2 PALET 24'E GÖRE RSME DEĞERLERİ	56
TABLO 5.3 PALET 12'E GÖRE RSME DEĞERLERİ	56
TABLO 5.4 PALET 32' GÖRE RSME DEĞERLERİ	57
TABLO 5.5 PALET 48'E GÖRE PSNR DEĞERLERİ	58
TABLO 5.6 PALET 24'E GÖRE PSNR DEĞERLERİ	58
TABLO 5.7 PALET 12'E GÖRE PSNR DEĞERLERİ	59
TABLO 5.8 PSNR DEĞERLERİ	59
TABLO 5.9 PALET 32'E GÖRE PSNR DEĞERLERİ	60
TABLO 5.10 PCA'IN UYGULAMA SONUCU PSNR DEĞERLERİ	61
TABLO 6.1 RMSE DEĞERLERİ	62
TABLO 6.2 PSNR DEĞERLERİ	63
TABLO EK1.1 HISTOGRAM DEĞERLERİ	74
TABLO EK1.2 RMSE DEĞERLERİ	74
TABLO EK1.3 PSNR DEĞERLERİ	75

KISALTMALAR

.Net	:Visual Studio 2003.Net
BG	:Bilgisayar Grafikleri
C#	:C Sharp
CAD	:Computer-aided design
CG	:Computer Graphics
CMS	:Color management system
CPU	:Control Unit System
CRT	:Cathod-raytube
ICC	:International color consortium
PCA	:Principal Components Analysis
PSNR	:Peak Signal to Noise Ratio
RGB	:Read, Green, Blue
RMSE	:Root Mean Squared
TIF	:Targed Image File Format

1 GİRİŞ

Günümüzde resimler (görüntüler) üzerinde işlemler yapılması, üzerinde çok çalışılan bir konu olarak karşımıza çıkmaktadır. Görüntülerin digital kaydedilebilmesi, digital fotoğraf makinalarının yaygınlaşması ve internet kullanımının artması fotoğraflar üzerinde çeşitli işlemlerin yapılmasına uygun bir ortam oluşturmuştur. Görüntü işlemede (image processing), Bilgisayar Grafik tekniklerinden (BG) (Computer Graphics, CG) büyük ölçüde yararlanılmaktadır. BG leri bir görüntünün oluşma aşamalarının her adımında kullanılır.

Çeşitli törenlerde stadyumlarda yapılan fon gösterilerinin sayısal ortamda otomatik olarak gerçekleştirilmesini sağlayan bir sistemin oluşturulması, bu çalışmanın esas konusunu oluşturmaktadır.

Bugüne kadar ilgili işlemler elle yapılmakta, bu bağlamda görüntüler ızgara şeklinde küçük karelere bölünerek ve bu kareyi temsil edebilecek renk levhaları belirlenmektedir. Bu belirlenen renkler, 150x36'lık ızgara şeklinde karelere ayrılmış şablon üzerine renk isimleri örneğin beyaz renk için B şeklinde yazılır. Bu işlem tüm görüntüler için yapılmakta ve elde edilen veriler ile hangi öğrencinin hangi levhayı kaldıracağını belirten görev kartları basılır. Bu işlemlerin yapılması verilen bir görüntü için büyük emek ve zaman kaybına yol açmaktadır. Bazı durumlarda ise pratik olarak imkansız olmaktadır.

Amacımız bilgisayar dosyalarında tutulan resimlerin daha alçak çözünürlüklü sistemlerde görüntülenmesidir. Tez çalışmamızda resimlerin alçak çözünürlükte görüntülenebilmesi için renk azaltma (Color Reduction) konusu ele alınmıştır.

Ana bileşenler analizi (Principal Components Analysis); yüksek boyutlu verileri düşük boyutlu verilere dönüştürebilen analiz tekniğidir. Tez çalışmamızda, renk azaltma işlemlerinin ana bileşen tekniklerine dayalı olarak yapılması hedeflenmiştir. Buna göre renk paletleri görüntüye özel olarak elde edilmesi ile renk azaltma minimum düzeyde gerçekleştirilmiştir. PCA'da ana bileşenlerin elde edilmesinde istatistik ve geometrik bağımsızlıklar kullanılabilir. İstatistiksel olan ana

bileşenler p tane rastgele $X_1, X_2, \dots, X_p$ nin doğrusal bileşenleridir. Yeni elde edilenler birbirlerinden bağımsızdır. Geometrik bağımsızlıkta ise, bileşenler yeni koordinat sisteminde gösterilir ve birbirlerine dik olup ara değerleri bulunmamaktadır. Bu da aralarında ilişkinin olmadığından kaynaklanmaktadır. Bizim uygulamamızda her ikisinden de faydalanıldı.

PCA'nın kullanımı ile, PCA'nın sunduğu varyansa göre en küçük yoğunlukta olan değer atılarak renk azaltma işlemi gerçekleştirilmiş ve buradaki değerler RGB'e dönüştürülerek görüntüde bulunan renklerden renk paletleri oluşturma işlemi gerçekleştirilmiştir. Burada en küçük yoğunlukta olan değer atılıp, diğer değerler ile verilerin tanımlanabilmesi PCA'nın sunduğu bir avantajdır.

Bu çalışmada öncelik Windows daki renk paletleri kullanılmış ve bunlar eldeki görüntüler üzerinde uygulanmıştır. Daha sonra benzer paletler PCA analizi ile herbir görüntü için ayrıca düzenlenerek geliştirilen uygulamada kullanılmışlardır.

Windows'un basit renklerinden oluşan 48'lik renk paletinin ortalamaları alınarak, 48, 24 ve 12'lik renk paletleri oluşturulmuştur. Bu 24 ve 12'lik renk paletleri oluşturulurken renklerin ortalaması alındığı için beyaz renk kaybolmaktadır. Bu yüzden ek olarak 24 ve 12'lik renk paletlerinde beyaz korunarak yeniden Windows'un 48, 24 ve 12'lik renk paletleri oluşturulmuştur. PCA analizi ile 48, 24 ve 12 renkten oluşan renk paletleri, elde edilen değerlerin birbirinden bağımsız olmaları sayesinde, veriler sıralanarak belirli aralıklara bölünüp, bu aralıkların orta noktaları alınarak elde edilmiştir.

Referans görüntüden elde edilecek yeni görüntü stadyumlarda gösterileceği için stadyumdaki boyutuna göre ayarlanması gerekmektedir. Oluşacak görüntü 150x36 lık boyutlarda olacaktır. Bu yüzden orjinal görüntü piksel piksel okunarak 2x2 lik bloklara ayrılarak işlem yapılmaktadır. 2x2 lik her bir blokta toplam 4 piksel vardır, bir blokta piksellerde bulunan renklerin ortalaması alınarak elde edilen renk yeni görüntünün piksellerine işlenir. Böylece referans görüntümüzü elde ederiz. Yapılan işlemlerde bilgisayar grafiklerinin sunduğu olanaklar kullanılmaktadır.

Bu tezde işletim sistemi olarak Windows XP kullanılmıştır. Ana bileşenler analizi ve görüntünün kendine ait renklerinden oluşan 48, 24, 12 lik renklerden oluşan renk paletinin hazırlanması işlemi Matlab7.1 programlama ortamında, PCA analizi ile elde edilen renk paletleri ve Windows'un basit renklerinden oluşan 48, 24, 12'lik renkten oluşan renk paletleri, görüntü üzerinde basit bir algoritma ile en çok kullanılan 32 renkten oluşan renk paletleri ile görüntü üzerinde metrik kullanılmış ve diğer tüm işlemler için Microsoft Visual Studio .NET 2003 (.Net) C# (C Sharp) kullanılmıştır. C# da görüntüleri piksel olarak okuyup değişiklikler yapılarak tekrardan aynı konumdaki piksellere aktarma işlemleri mevcut sınıfların (class) mevcut metotları yardımı ile basitçe yapılmaktadır.

Matlab7.1 ve .Net ortamında yazılan uygulamada “.tif” (tagged image file format) uzantılı resimler kullanılmıştır. Bu dosya tipi, platformdan bağımsız olarak görüntü değişikliklerini sağlar, ikili görüntü (binary images), gri seviye (gray-scale) görüntü ve RGB gibi birçok renk modelini destekler. Bizim uygulama geliştirme ortamımız için de en uygun dosya tipidir.

Bir görüntünün (resim) üzerinde işlem yapabilmenin ilk adımı görüntünün piksel değerlerinin (x,y) koordinatlarına göre okunmasıdır. Okunan bu pikseller üzerinde gerekli değişiklikler yapıp yeni oluşan pikselin yine aynı koordinata yerleştirilmesi diğer bir adımdır. Bu işlem bizim uygulamamızda ve buna benzer birçok uygulamada ilk adım olarak kullanılır. Pikseller üzerinde işlemler yapılması bölüm 4.1'de anlatıldığı gibi işleme alınmıştır.

İncelenen görüntüler temelde RGB renklerden oluşmaktadır, çoğu grafik uygulamada da yine RGB renkleri kullanılır, fakat bu çalışmada ana amacımız, $2^{24} = 16$ milyon alternatifi bulunan bu sistemin renk adedini düşürmektir. Bunun için ana bileşenler analizi kullanılarak renk boyutu düşürülüp, 48, 24, 12 renk içeren paletler ele alınmakta ve Windows'un renk paletinde bulunan 48 renk ele alınmakta, bu elde edilen paletten, 24 ve 12 renk içeren paletler elde edilerek, eldeki görüntüyü bu paletlerdeki renklerle ifade etmek ve bu görüntüye en yakın çıktı elde edilmeye çalışılmıştır. Palet oluşturma işlemleri bölüm 4.2'de anlatıldığı gibi elde edilmiştir.

Renk paletlerine göre görüntünün işlenmesinde üç farklı metrik kullanılmaktadır; en küçük kareler toplamı, en küçük mutlak değer ve ortalama değerler. Bu metrikler kullanılarak üç farklı palet ve üç farklı metriğe göre toplam dokuz adet görüntü elde edilmektedir. Bu görüntüler, Windows'un basit renk paletlerine, PCA yönteminden elde edilen paletlere ve resim üzerindeki renklerden oluşan 32'lik renk paletine göre ayrı ayrı oluşturulabilir. Bu konu ile ilgili çalışma örnekleri bölüm 5.1'de detaylı olarak verilmektedir.

Elde edilen bu görüntülerin referans görüntü ile karşılaştırılmasında iki farklı metrik kullanılmaktadır; en küçük kareler ortalamasının karekökü RMSE (Root Mean Squared Error) ve RMSE ölçümünün bir fonksiyonu olan PSNR (Peak Signal to Noise Ratio) dır. Elde edilen RMSE ve PSNR değerleri detaylı olarak bölüm 5.2'de incelenmektedir.

Uygulamada elde edilen görüntülerin histogramları, tablo 4.7'deki algoritma ile elde edilerek incelenmektedir. Renk histogram grafikleri bölüm 4.3.1'de gösterilmektedir.

Bölüm 4.2.3'de portre görüntüleri üzerinde PCA analizi uygulanarak elde edilen görüntü orjinal görüntü boyutlarında gösterilerek PSNR değerleri incelenmektedir.

Tezin organizasyonu şu şekildedir: İkinci bölümde, görüntü işleme teknikleri hakkında genel bilgilerin verildiği “temel kavramlar” kısmı yer almaktadır. Görüntü işleme ve renk azaltma konularında özet bilgi verilmektedir. Üçüncü bölümde, uygulama gerçekleştirilirken yararlanılan çalışmaların tanıtıldığı “ilgili çalışmalar” kısmı yer almaktadır. Dördüncü bölüm, tezin gerçekleştirdiği her bir adım ayrıntılı biçimde açıklanmıştır. Bu bölüm görüntülerdeki renk azaltma ve elde edilen bu yeni renklerden görüntü elde etme konusunu detaylı olarak anlatmaktadır. Beşinci bölümde, geliştirilen uygulamada kullanılan metrikler örnekler ile gösterilmiştir. Altıncı bölümde, referans görüntüden elde edilen yeni görüntülerde meydana gelen sorunlu bölgeler için yapılan işlemler anlatılmaktadır. Yedinci bölüm olan “sonuç” bölümünde geliştirilen uygulama yardımı ile ulaşılan bulgular ve ileriye dönük yapılabilecek çalışmalar hakkında bilgiler verilmiştir. Bu bölümü

“kaynaklar dizini”, “program çıktıları” ve “ekler” başlıklı bölümler izlemektedir.

2 TEMEL KAVRAMLAR

Bu bölümde tez kapsamında kullanılan görüntü işleme teknikleri kısaca tanıtılacaktır. Aşağıda anlatılan yöntemlerin çoğu tezin farklı aşamalarında, uygulamanın hedefini gerçekleştirmesinde önemli katkıları olan tekniklerdir.

2.1 Bilgisayar Grafikleri

Bilgisayar Grafikleri bilginin (information) gösterilmesinde gerçek dünyadaki görüntülerin gerçeğe yakın bir şekilde aktarılmasında insanlar ve makinalar arasında kılavuzluk eder. (Angel Edward, 2006)

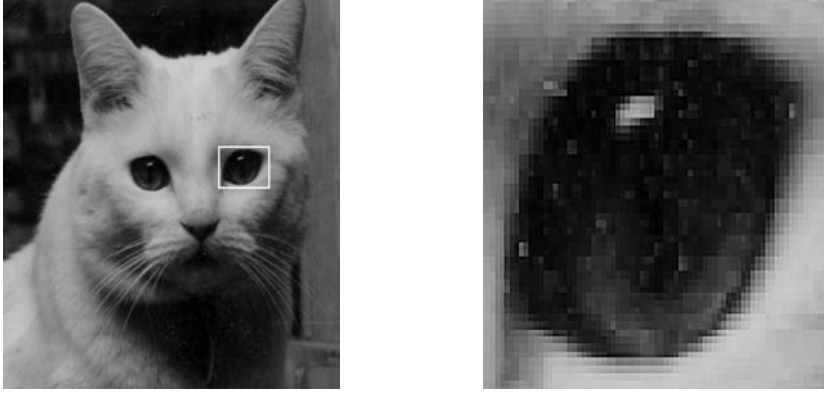
BG yaklaşık 50 yıl önce Cathode-raytube (CRT) üzerinde birkaç çizgi gösterimi ile başlamıştır. Günümüzde teknolojinin ilerlemesi ile bilgisayar ekranı üzerinde görüntü oluşturabilmekte ve bu görüntüler gerçeğe çok yakın olmaktadır. Gerçek zamanlı, gerçek ortamda simülasyonlar oluşturabilir. Örnek olarak uçak simülasyonunu verebiliriz.

Fotoğraf veya bilgisayarda görüntü oluşturmak için uygulama programları, grafik kütüphaneler ve grafik sistemler arasında ilişkiler (relationship) oluşturabiliriz.

2.2 Pikseller ve Görüntü Ara Bellek

Bütün grafik sistemler bit tabanlıdır (bitmap). Bu sistemlerde görüntüler piksellerden oluşmaktadır, görüntü pikselleri nümerik dizilerden oluşmaktadır.

Şekil 2.1 deki görüntüde en küçük alan pikseldir. Görüntünün tüm pikselleri görüntü arabellekte tutulur. Görüntü arabellek grafik sistemler için çok önemlidir.



Şekil 2.1: Piksel görüntü (Angel Edward, 2006)

Çözünürlük (Resolution) için görüntünün renk derinliği (color depth) önemlidir. Derinlik piksellerin bellekte tutulma durumu ile ilgilidir. Örneğin; 1 bit derinlik verilirse görüntü arabellekte iki renk tutulabilir. Çünkü 1 bite 0 veya 1 değerleri verilebilir. Eğer derinlik 8 bit olarak verilirse, görüntü arabellekte $2^8 = 256$ renk tutulabilir. Gerçeğe yakın renkleri kullanmak istersek, tüm renk sisteminde (full colour) 24 bit'lik derinlik kullanmalıyız, böylece görüntü arabellekte $2^{24} = 16,777,216$ renk tutulabilir.

Görüntü arabellekleri (image buffers), üç boyutlu görüntüler ve renkli görüntüler için birçok değeri tutmaktadırlar. Örneğin derinlik önemlidir, bunu tutarlar. Bu tür sistemlerde görüntü arabellekleri birden fazla arabellek (buffer) dan oluşmaktadırlar.

İlk grafik sistemlerde görüntü arabellek standart belleğin bir bölümüydü ve CPU da tutulurdu. Günümüzde hemen hemen bütün grafik sistemler özel amaçlı grafik işlem birimleri (Graphics Processing Units (GPUs)) ile tanımlıdır ve özel grafik fonksiyonlarını bileşenleri içerisinde bulunmaktadır.

2.3 Resim Tipleri (Image Types)

Aşağıda bazı resim tipleri anlatılmaktadır (Matlab6.5 Image Processing Toolbox Documentation). Uygulamaya girdi olarak alınan resimlerin üzerinde işlem yapılabilmesi için, resimlerin mutlaka bu tanımlanan tiplerden birinde olması gerekir.

2.3.1 Gerçek Renk (true color)

Bizim bu uygulamada kullandığımız görüntüler RGB tipindedir. RGB renk sisteminde görüntü derinliği 24 bittir. Her pikselin renk değeri için Kırmızı, Yeşil ve Mavi oranlarını belirleyen üç farklı değer tutarlar. (Red (R) Kırmızı, Green (G) Yeşil, Blue (B) Mavi)

2.3.2 Gri Seviye (gray scale)

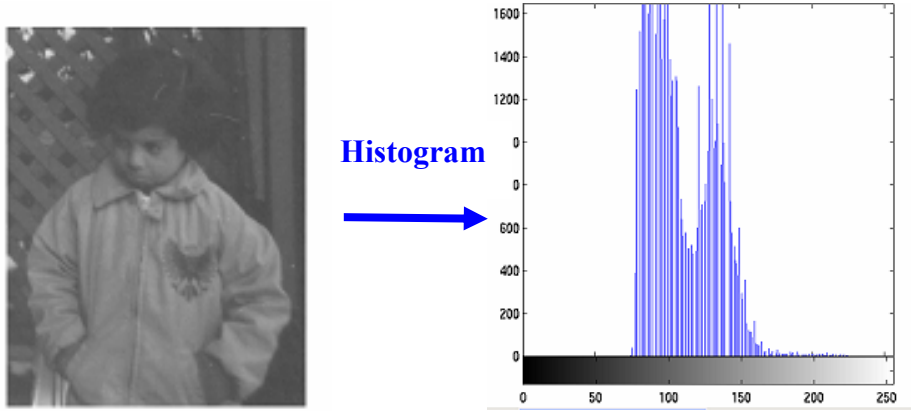
Bu görüntüler bildiğimiz gri seviye görüntülerdir. Bu tipteki görüntülerde her piksel değeri için tek bir değer tutulur. Piksellerin gri seviye değerleri ya $[0, 1]$ aralığındaki “double” sayılarla ya da “uint8” veya “uint16” olarak ifade edilen sırasıyla $[0, 255]$ veya $[0, 65535]$ aralığında değişen değişen değerlerle ifade edilirler.

2.3.3 İndeksli Renk (Indexed)

İndeks tipindeki resimler RGB resimlerden farklı olarak bir renk değerini üç (Red (R) Kırmızı, Green (G) Yeşil, Blue (B) Mavi) farklı değerle temsil etmek yerine tek “double” değer ile ifade edilirler.

2.4 Görüntü Histogramı (Histogram)

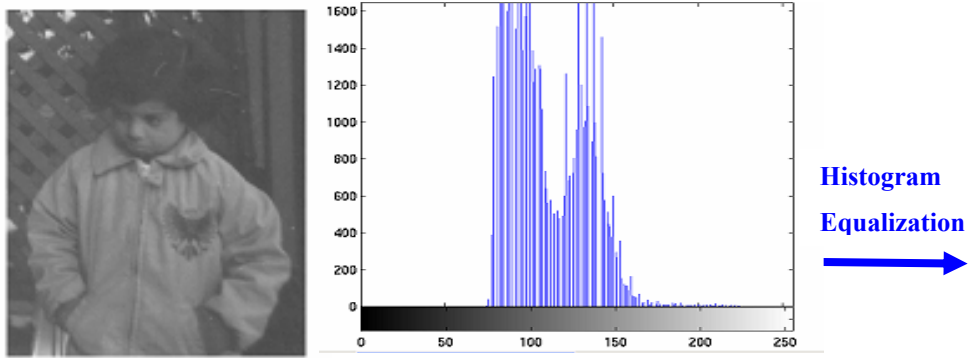
Gri seviye veya indeksli bir resmin piksel yoğunluk (intensity) değerlerinin x ekseninde o yoğunluktaki toplam piksel sayılarının y ekseninde yer aldığı grafiğe o görüntünün histogramı denir (Matlab6.5 Image Processing Toolbox Documentation). Resmin histogramı resmin kontrast düzeyi hakkında bilgi verir. Histogramdaki yığılmalar resmin belirli yoğunluklara sahip piksel sayısının fazla olduğunu ve görüntü kontrastının düşük olduğunu belirtir. Şekil 2.2’de bir görüntü ve resme ait histogram gösterilmektedir.



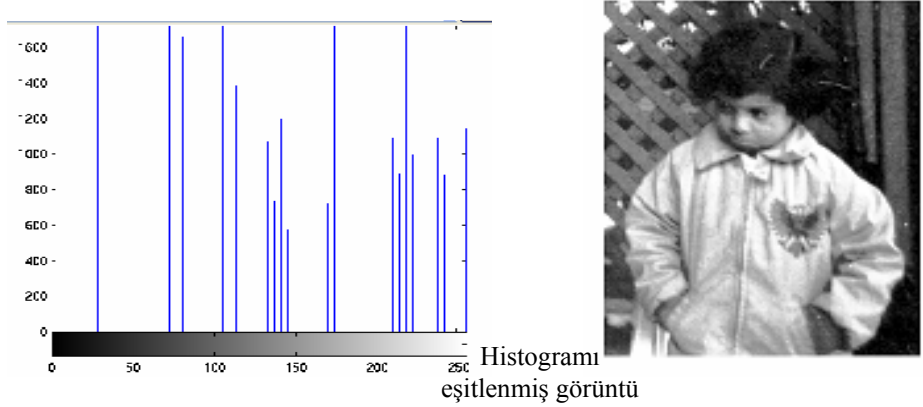
Şekil 2.2: Bir görüntü ve histogramı (Matlab6.5 Kütüphanesi)

2.5 Histogram Eşitleme (Histogram Equalization)

Gri seviye görüntülerin kontrastını artırmak için görüntülerde dar bir aralığa sıkışmış birbirine yakın değerlere sahip gri tonlar arasındaki farkı artırmak için histogram eşitleme yapılmaktadır. Gri ton değerlerini daha geniş bir aralığa yayıp, böylece görüntüdeki mevcut tonlar arasındaki farkı açmak mümkündür (Matlab6.5 Image Processing Toolbox Documentation). Şekil 2.3’da histogram eşitlemenin uygulandığı bir görüntüdeki ve histogramı sunulmuştur.



Orijinal görüntü
histogramı

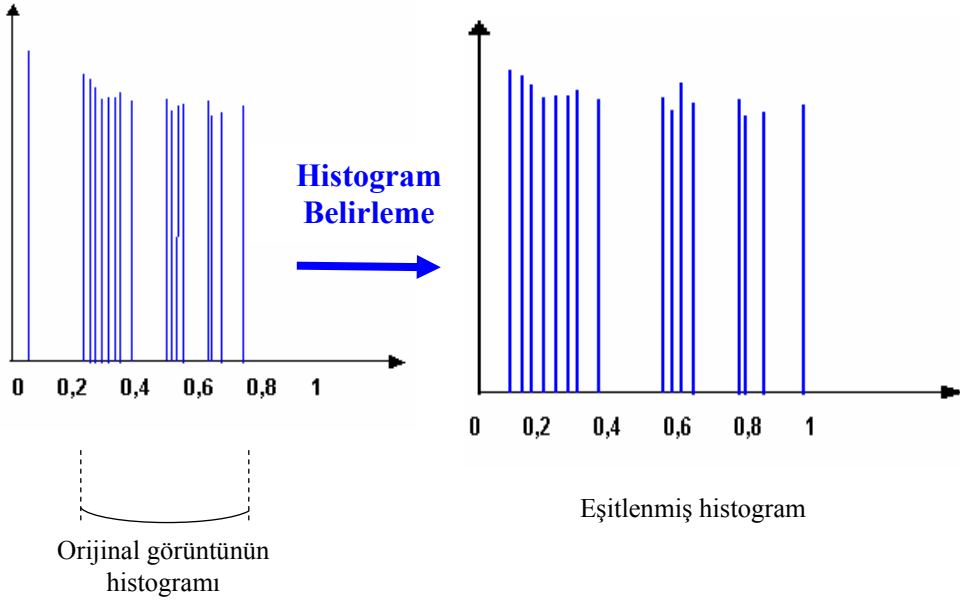


Şekil 2.3: Histogram Eşitleme (Matlab6.5 Kütüphanesi)

2.6 Histogram Belirleme (Histogram Specification) – Yoğunluk Ayarlama (Intensity Adjustment)

Bazen resmin daha kontrast hale getirilmesi için histogram eşitleme bile yeterli olmaz. Örneğin histogram eşitlemeye rağmen yine de yığılmalar olursa görüntüde istenen kontrastlık elde edilemez. Bu gibi durumlarda girdi resimdeki 1-0 aralığına yayılan gri seviye değerlerden belli bir aralıktaki değerler daha farklı bir aralıkla eşleştirilir. Bu işleme

histogram belirleme veya yoğunluk ayarlama denir (Matlab6.5 Image Processing Toolbox Documentation). Şekil 2.4’de orijinal resmin histogramındaki (0,2 - 0,8) aralığındaki değerler (0-1) aralığına eşleştirilmiştir.



Şekil 2.4: Histogram Belirleme

Tezimizdeki çalışmada elde ettiğimiz görüntülerin histogramları alınarak renk tonlamalarına bakılmıştır. Uygulamada kullanılan görüntülerin histogramları bölüm 4’de gösterilecektir.

2.7 Görüntü Karşılaştırma Metrikleri

Aşağıda görüntü karşılaştırma metrikleri anlatılmaktadır. Bu metrikler ile; orijinal görüntü ve uygulamada elde edilen görüntü arasında karşılaştırma yapılarak en uygun görüntü elde edilir.

2.7.1 RMSE (Root Mean Squared Error)

RMSE; her veri noktası ile model arasında birbirine uymayan noktaların ortalama oranıdır. RMSE değerlerine bakılarak kalite ölçümü yapılabilmektedir. RMSE değeri yüksek ise problem var, düşük ise veriler modele yakın, sıfır ise veriler model ile aynıdır diyebiliriz (<http://www.mathworks.com>, 1984-2007).

RMSE'in (3.1) ve (3.2)'de formülü görülmektedir, bu formül PSNR değerlerinin hesaplanmasında kullanılmaktadır.

$$MSE = \frac{\sum_{M,N} [I_1(m,n) - I_2(m,n)]^2}{M * N} \quad (3.1)$$

$$RMSE = \sqrt{MSE} \quad (3.2)$$

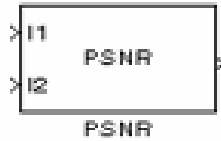
RGB renk değerlerine göre RMSE formülleri (3.3)'de verilmektedir. Formül (3.1)'de kullanılan I_1 ve I_2 ifadeleri Şekil 2.5'de verilmiştir.

$$\begin{aligned} RMSE_r &= \sqrt{\frac{1}{n} \sum_{i=1}^n (r_i - R_i)^2} \\ RMSE_g &= \sqrt{\frac{1}{n} \sum_{i=1}^n (g_i - G_i)^2} \\ RMSE_b &= \sqrt{\frac{1}{n} \sum_{i=1}^n (b_i - B_i)^2} \end{aligned} \quad (3.3)$$

2.7.2 PSNR (Peak Signal-to-Noise Ratio)

Görüntüler arasında en yüksek sinyal gürültü oranı hesaplaması olarak ifade edilebilir (<http://www.mathworks.com>, 1984-2007).

İki görüntü arasında PSNR ölçümü desibel olarak ifade edilir. PSNR oranı; orjinal ile işlenmiş veya sıkıştırılmış görüntü arasındaki kalite ölçümü için kullanılmaktadır. PSNR değerinin yüksek olması elde edilen görüntünün kalitesinin yüksek olduğu anlamına gelmektedir.



Şekil 2.5: PSNR

PSNR, hesaplamasında MSE veya MSE'in karekökü RMSE kullanılmaktadır.

$$PSNR = 10 \log_{10} \left(\frac{R^2}{MSE} \right) \quad (3.4)$$

Formüldeki R, 8 bitlik integer veri tipinde tanımlanır ise, hesaplamalarda 255 olarak alınabilir. Formül (3.5)'de PSNR değerinin RMSE değeri kullanılarak hesaplanması verilmektedir.

$$PSNR = 10 \log_{10} \frac{255^2}{MSE} = 20 \log_{10} \frac{255}{\sqrt{MSE}} = 20 \log_{10} \frac{255}{RMSE} \quad (3.5)$$

$$PSNR = 10 \log_{10} \left(\frac{255^2}{\frac{(MSE(R) + MSE(G) + MSE(B))}{3}} \right) \quad (3.6)$$

Renki görüntülerin PSNR değerleri R,G,B değerleri ile hesaplanır. Renkli görüntülerin PSNR değerleri hesaplama formülleri (3.6) ve (3.7)'de verilmiştir. Her iki formülde kullanılabilir ama genellikle (3.7)'deki formül kullanılmaktadır (<http://ieeexplore.ieee.org>, 2007).

$$PSNR = \frac{PSNR_R + PSNR_G + PSNR_B}{3} \quad (3.7)$$

Formül (3.8)'de RGB değerlerine göre PSNR hesaplama formülleri verilmektedir.

$$PSNR_r = 20 \log \left(\frac{255}{RMSE_r} \right)$$

$$PSNR_g = 20 \log \left(\frac{255}{RMSE_g} \right) \quad (3.8)$$

$$PSNR_b = 20 \log \left(\frac{255}{RMSE_b} \right)$$

3 PROBLEMİN TANIMI VE İLGİLİ ÇALIŞMALAR

Resmi törenlerde stadyumlarda öğrenciler tıpkı bir grafik sistemini taklit eder gibi ellerinde bulundurdıkları renk panolarını kaldırarak bir resmin görüntüsünü elde ederler, bu görüntüler stadyumda orta alanda gerçekleştirilen yer hareketleri ile de uyumludur.

Stadyumda bu görüntünün oluşması için uzun ve yorucu bir çalışma gerektirir. Öncelikle resim öğretmenleri uygun resimleri kağıt üzerine çizerler ve bu resimleri 150 sütun, 36 lık satırdan oluşan karelere bölerler. Ve bu her kareye bir renk gelir, bu renk öğrencinin kaldıracağı panonun rengidir. Bu verilerin bilgisayara işlenmesi için 150x36 kareden oluşan şablonlara bu renkler elle kodlanır. Örnek olarak Kırmızı → K, Beyaz → B harfleri ile temsil edilerek yazılır.

Bunlar bilgisayara girilir ve girilen bu bilgiler doğrultusunda öğrencilerin hangi renkleri kullanacağı ortaya çıkarılıp görev kartları oluşturulur.

Verilerin bilgisayara işlenip sonuç alınıncaya kadar geçen sürede konu ile ilgili olan görevli kişiler uzun bir çalışma yapmaktadırlar. Bu işlemler uzun zaman almakta ve benzer işlerin tekrarları olmaktadır.

Tezimizde sadece resim ele alınacak ve bilgisayara aktarılarak bilgisayar grafik yöntemleri yardımı ile referans resme en yakın görüntü elde edilecek ve gerektiğinde bu görüntü üzerinde değişiklikler yapılarak veya görüntü baştan tekrar oluşturularak kaydedilecektir.

Tez içeriğinde referans olan çalışmalar aşağıda kısaca anlatılmıştır. Tezde temel seçilen bir görüntünün törenlerde stadyumda değişik renklerden oluşan levhalarla gösterilimi üzerinde durulmuştur. Her yıl binlerce öğrencinin ve uzmanın çalışarak oluşturduğu bu görüntüler ve bunlarla ilgili detay çalışma bilgisayar tekniklerini kullanarak bilgisayar ortamında oluşturulmuştur. Aşağıda uygulanan yöntemler incelenecektir.

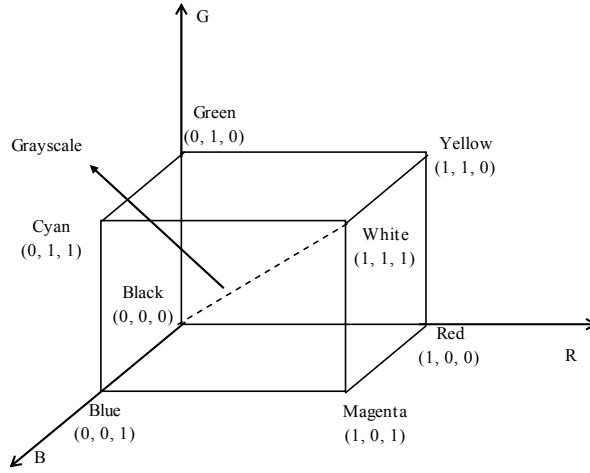
3.1 Renk Azaltma (Color Reduction)

Tüm sayısal işlem yapan makinaların renk yönetimi farklıdır. Çok yüksek kaliteli bir sayısal fotoğraf makinasında çekilen görüntü farklı monitörlerde aynı renkler ile görüntülenemez. Bunun nedeni her makinanın kendisine özgü bir renk uzayı vardır. Sadece kendi renk uzayındakini gösterebilir. Böylece renk kayıpları ortaya çıkar.

Makinalardaki renk eşleştirmelerinde problemler meydana gelmektedir. Bunun nedeni farklı renk uzaylarını kullanan birçok aygıt (device) ve yazılım vardır (software). Bir çözüm olarak renk çeviriciler verilebilir. Renk yönetim sistemi (color management system (CMS)) farklı aygıtlar arasında renk uzaylarını karşılaştırır ve yakın renk ayarlaması yapar. Renk ayarlarında ve doğru rengi bulmada problem vardır. PhotoShop renk yönetiminde uluslararası renk konsorsiyumu (International Color Consortium (ICC)) kullanıyor (PhotoShop Uygulama yardımı (Adobe PhotoShop Help)). ICC'in renk standardı vardır. Görüntüdeki renkler monitörün ICC standardına göre çevrilir.

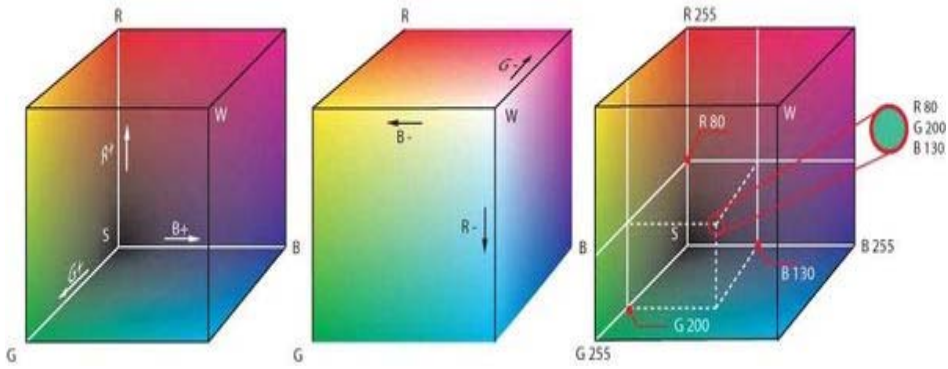
Renk azaltma; görüntüde bulunan renklerden birbirine yakın renkler için kullanılan palettir. Renk azaltma algoritması GIF (Graphical Interchange Format) sıkıştırma işleminde (compression) kullanılır. Renk numaraları azaltılır. Gif maksimum 256 renge sahiptir. (PhotoShop Uygulama yardımı (Adobe PhotoShop Help))

Şekil 3.1'de Renk küpü görülmektedir. Renk küpü renk azaltmalarında kullanılır. Renk küpünden tek tip (uniform) örnek renkler ile palet oluşturulur.



Şekil 3.1: Düz panel (Angel Edward, 2006)

Şekil 3.2’de Renk uzayına göre RGB cinsinden renklerin oluşturulması görülmektedir.



Şekil 3.2: Renklerin oluşturulması (<http://tr.wikipedia.org/wiki/RGB,2007>)

3.2 Ana Bileşenler Analizi (Principal Components Analysis – PCA)

Ana bileşenler analizi; yüksek boyutlu verileri düşük boyutlu verilere dönüştürebilen analiz tekniğidir. Bu analiz; azaltma, indirgeme tekniklerinde kullanılır. Uygulama alanlarına göre, Karhunen-Loeve Transform (KLT, Kari Karhunen ve Michel Loeve), Hotelling transform (Harold Hotelling) veya Proper Orthogonal Decomposition(PDO) isimlerini alır (<http://en.wikipedia.org/wiki/Principalcomponentanalysis>).

PCA, cebirsel olarak ana bileşenler p tane rastgele değişken $X_1, X_2, \dots, X_p$ nin doğrusal bileşenleridir. Geometrik olarak bu doğrusal bileşenler, koordinat eksenleri $X_1, X_2, \dots, X_p$ olan orjinal sistemi döndürerek bulunan yeni koordinat sistemini gösterirler (Özdamar K., 2002).

X matrisinde bulunan değişkenlerin doğrusal bileşenlerini bulmak için kovaryans veya korelasyon matrisinin özdeğerleri (eigenvalues) ve özvektörleri (eigenvectors) kullanılmaktadır. Herbir özdeğere karşılık gelen özvektörler birbirlerinden bağımsızdırlar, başka bir ifade ile birbirlerine diktirler. Ana bileşenler varyansların büyüklüklerine göre sıralanır. . Büyük varyans birinci koordinatta, ikinci büyük varyans ikinci koordinatta ve benzeri devam edilerek bulunurlar.

PCA, büyük varyansların alt uzaylarını korumak için doğrusal dönüşümde üstünlüğe sahiptir. Bu PCA nın avantajı olarak görülmektedir ama kıyaslama yapılırken büyük boyutlu veriler için hesaplamalar yüksek maliyet gerektirebilir.

$X' = [X_1, X_2, \dots, X_p]$ vektörü p adet veriden oluşmaktadır,

Kovaryans hesaplanarak özdeğerler ve özvektörler hesaplanabilirler.

$$C e' = e' D$$

e' öz vektörler matrisi olarak adlandırırız, C kovaryans matrisinin köşegen (diagonalizes) değerleri ile hesaplanır. D , C 'nin özdeğerlerinin köşegen matrisidir.

e' özvektörler matrisi sıralı sütunlardan oluşur ve D özdeğer matrisi, azaltılmış özdeğerler olarak düzenlenir.

$$D = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \lambda_m \end{bmatrix} \quad \lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_p \geq 0 \text{ şeklinde ifade edilebilir.}$$

(Overall John E., Klett C. James, 1972)

$$e' = [e_1, e_2, \dots, e_p]$$

$Y_1, Y_2, \dots, Y_p$ ana bileşenleri birbirlerinden bağımsız ve varyanslarının maksimum değerlerine göre sıralanarak değerlendirmeye alınırlar. Ana bileşenler özdeğerlere karşılık gelirler.

Bu açıklamalardan sonra X matrisinin ana bileşenleri

$$Y_i = e_i' X = e_{1i} X_1 + e_{2i} X_2 + \dots + e_{pi} X_p$$

eşitliği ile bulunurlar.

$$\text{Var}(Y_i) = e_i' \sum e_i = \lambda_i \text{ ve } \text{Cov}(Y_i, Y_k) = e_i' \sum e_k = 0$$

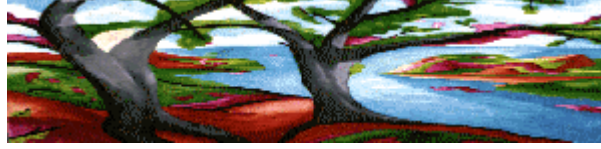
Bulunan ana bileşenlerden birkaçı, toplam değişkenliğin %80'nini oluşturabiliyorsa, bu bileşenler az veri kaybı ile orjinal verileri tanımlayabilir olarak kabul edilirler.

PCA'nın hesaplamalarında kullanılan formüller ve bütün matrisler ile ilgili işlemlerde bilgisayar tabanlı programlar kullanılabilir. Örnek olarak istatistik tabanlı Matlab verilebilir. Matlab'da ana bileşenler ve varyanslar, princomp fonksiyonu kullanılarak kolayca bulunurlar.

4 GELİŞTİRİLEN YÖNTEMLER

4.1 Referans Görüntü Üzerinde Yapılan İşlemler

Bu çalışmada resimler üzerinde rahat işlem yapabilmek için daha önceden belirlediğimiz ortak bir formata çevrilir. Bu resimler “.tif” uzantılı olarak alınır. Şekil 4.1’de orjinal resim görülmektedir ve bu resim RGB renk tipindedir. Görüntünün boyutu soldan sağa 300, yukarıdan aşağıya 72 pikselden oluşmaktadır. Görüntünün tamamında $300 \times 72 = 21.600$ piksel bulunmaktadır. Stadyumda oluşturulacak görüntü ise $150 \times 36 = 5400$ piksel boyutlarındadır.



Şekil 4. 1: Orijinal Resim

Öncelikle orjinal resim piksel piksel okuyarak 2x2 lik bloklara ayrılır. Her 2x2 lik bloklarda 4 piksel yer alır. Şekil 4.2 de 2x2’lik blok gösterilmiştir. Her piksel RGB olarak üç renk tutmaktadır. Bu renklerin ayrı ayrı (kırmızı, yeşil, mavi) ortalamaları alınarak tek bir renge indirgenmektedir. Dört pikselin ortalamasından elde edilen bu yeni renk piksellere aktarılır (set). Böylece 150x36 lık toplam 5400 pikselden oluşan referans görüntü Şekil 4.3 deki gibi oluşturulur.

Piksel 1	Piksel 3
Piksel 2	Piksel 4

Şekil 4. 2: Referans Görüntü

$$\text{Ortalama K} = \frac{(\text{Piksel1}_r + \text{Piksel2}_r + \text{Piksel3}_r + \text{Piksel4}_r)}{4} \quad (4.1)$$

$$\text{Ortalama Y} = \frac{(\text{Piksel1}_g + \text{Piksel2}_g + \text{Piksel3}_g + \text{Piksel4}_g)}{4} \quad (4.2)$$

$$\text{Ortalama M} = \frac{(\text{Piksel1}_b + \text{Piksel2}_b + \text{Piksel3}_b + \text{Piksel4}_b)}{4} \quad (4.3)$$

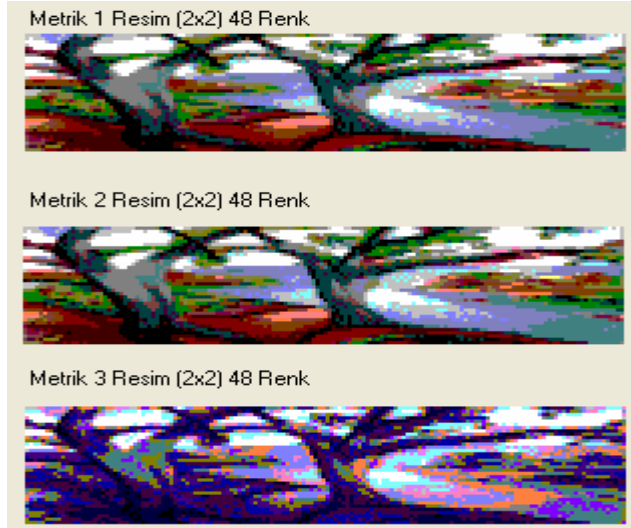
Formül (4.1), (4.2) ve (4.3) den elde edilen üç rengin birleşiminden oluşan renk referans görüntünün piksellerine aktarılır. Bu işlem tüm piksellere uygulandıktan sonra şekil 4.3 deki referans görüntü elde edilir.



Şekil 4. 3: Referans Görüntü

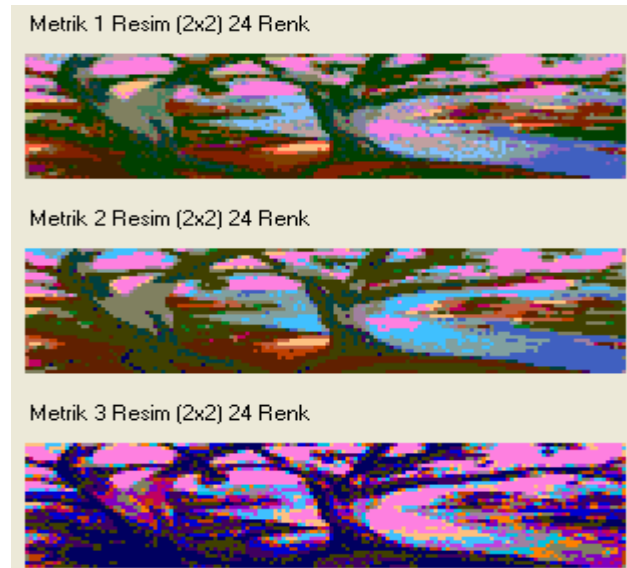
Referans görüntü üzerinde çeşitli metrikler kullanılarak görüntüdeki renk sayısının azaltılması gerekmektedir. Bu işlem içerisinde üç farklı renk paleti kullanılır, bu renk paletleri 48, 24 ve 12 renkten oluşmaktadır. Paletlerdeki renkler ile referans görüntünün renk pikselleri arasındaki ilişkiyi bulmak için üç farklı metrik uygulanır ve elde edilen sonuçlara göre yeni görüntülerin pikselleri oluşturulur.

Şekil 4.4 de referans görüntü ile 48 lik palet arasında uygulanan üç farklı metriğin sonuçlarının görüntüleri verilmiştir.



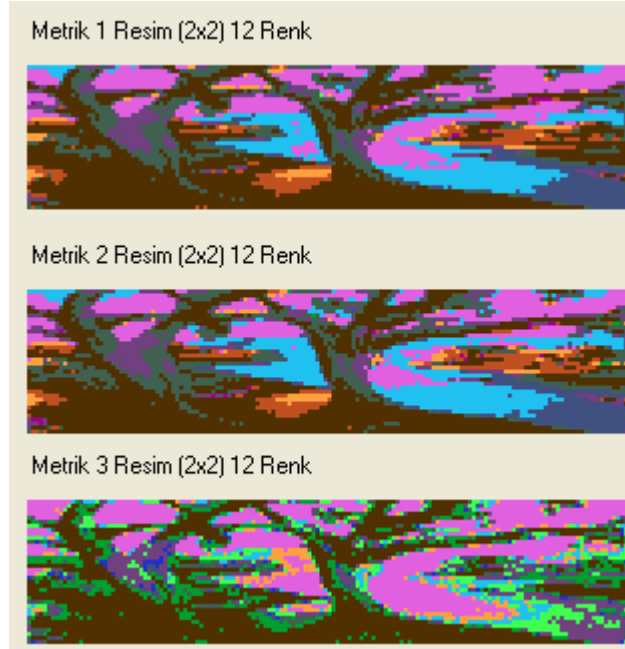
Şekil 4.4: 48 lik Palet ve Metrik Uygulama Sonuç Görüntüleri

Referans görüntüdeki pikseller ile 24'lük renk paletindeki renkler arasında uygulanmış olan üç farklı metrik sonucunda elde edilen yeni görüntüler Şekil 4.5 de verilmiştir.



Şekil 4.5: 24 lik Palet ve Metrik Uygulama Sonuç Görüntüleri

Referans görüntüdeki pikseller ile 12'lik renk paletindeki renkler arasında uygulanmış olan üç farklı metrik sonusunda elde edilen yeni görüntüler Şekil 4.6 de verilmiştir.

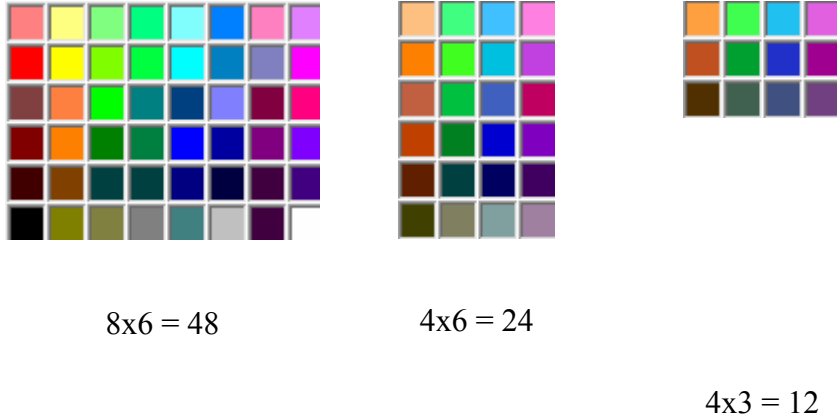


Şekil 4.6: 12lik Palet ve Metrik Uygulama Sonuç Görüntüleri

4.2 Renk Paletleri

4.2.1 48, 24, 12 lik renk paletleri

Referans görüntü üzerinde uygulanacak renk paletleri için Windows işletim sistemine ait temel 48 renkli palet kullanılmıştır. Bu paletin ikişer ikişer sütunlarının ortalaması alınarak 24 renkli palet elde edilir. Oluşan bu yeni paletin ikişer ikişer satırlarının ortalaması alınarak 12 renkli palet elde edilir. Bu paletler Şekil 4.7’de görülmektedir.



Şekil 4.7: Renk Paletleri

Paletteki her renk, kırmızı, yeşil ve mavi (RGB) renklerin birleşiminden oluşmaktadır. Renklerin gerçek yaşamdaki görüntülerini elde etmek için her piksele 1 byte ayrılmıştır. Kırmızı, yeşil ve mavi olarak düşündüğümüzde 3 byte eder ve toplam $3 \times 8 = 24$ bittir. Şekil 4.8’de siyah ve beyaz renge karşılık gelen RGB değerler verilmiştir.

	Siyah Renk		Beyaz Renk
	R $\rightarrow$ 0		R $\rightarrow$ 255
	G $\rightarrow$ 0		G $\rightarrow$ 255
	B $\rightarrow$ 0		B $\rightarrow$ 255

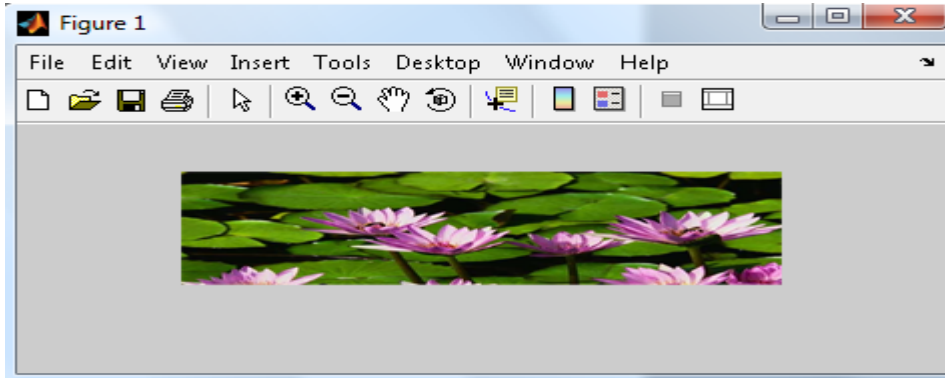
Şekil 4.8: Siyah ve Beyaz renklerin RGB değerleri

4.2.2 Ana Bileşenler Analizi (Principal Components Analysis)

Bu analiz ile yüksek boyutlu verileri düşük boyutlu verilere dönüştürebiliriz. Bu dönüşüm işlemi yapılırken orjinal verileri çok az kayıplı olarak tekrar tanımlayabilmek için gerekli veriler oldukça fazla

korunur ve sonuçta boyut azaltma işlemi daha başarılı olarak elde edilir. Bu konu detaylı olarak Bölüm 2’de anlatılmıştır. Burada bu analiz kullanılarak renk paleti oluşturulma konusu üzerinde durulacaktır.

Ana bileşenler analizi ile ilgili bu uygulama Matlab 7.1’deki fonksiyonlar kullanılarak oluşturulmaktadır. Öncelikle görüntüye ait verilerin matris şeklinde okunması gerekmektedir. Elde edilen matrisde, PCA analizindeki adımlar izlenerek boyut azaltma işlemi gerçekleştirilebilir.



Şekil 4.9: Orjinal görüntü

Tablo 4.1’de şekil 4.9’daki görüntünün matris halinde görüntü örneği verilmektedir. Bu matris üzerinden ana bileşenler analizi uygulanarak boyut azaltma işlemi gerçekleştirilecektir.

R	G	B
150	162	45
118	127	35
75	78	31
.	.	.

Tablo 4.1 Orjinal görüntünün RGB matrisi(toplam $n=21600$ piksel)

Değişkenler	PC 1	PC 2	PC 3
R	-0,6444	0,0338	-0,764
G	-0,4391	0,8015	0,4058
B	-0,6261	-0,597	0,5016

Tablo 4.2 pc katsayıları

Tablo 4.2’de ana bileşenler analizinde kullanılan katsayıları gösteren pc değerleri verilmiştir. Tablodan da anlaşıldığı gibi birinci PC genel olarak üç rengin (Kırmızı (R), Yeşil (G),ve Mavi(B)) ortalamasını, ikinci Pc ise yeşil ile mavi arasındaki kontrastı, üçüncü PC ise kırmızıya karşılık yeşil ve mavi arasındaki kontrastı temsil etmektedir.

10007,29
2035,538
122,0927

Tablo 4.3 pc değerlerinin varyansları

82,26349
16,73286
1,003646

Tablo 4.4 Ana bileşenlere ilişkin varyansların yüzdeleri

Herbir PC’ye ilişkin varyanslar Tablo 4.3’de bunların yüzdeleri ise Tablo 4.4 de verilmiştir. Görüldüğü gibi birinci ana bileşen toplam varyasyonun %82.26’sını, ikinci ana bileşen %16.73’nü, üçüncü ana bileşen ise %1.00’nı temsil etmektedir. Tablodan da anlaşıldığı gibi toplam varyansın yaklaşık %99’u ilk iki bileşken, geri kalan %1’i ise

üçüncü anabileşen tarafından temsil edilmektedir. Bu nedenle üçüncü anabileşen ihmal edilip bahis konusu görüntü için ilk iki ana bileşen dikkate alınmıştır. Böylece RGB ye dayalı 3 boyutlu renklendirme sistemi 2 boyutlu sisteme indirgenmiş olmaktadır.

1	2	3
-195,969	108,0507	-26,2732
-153,718	84,88602	-21,0475
-101,988	46,54626	-10,0902
-60,2622	17,9244	-2,04581
-49,8009	6,774175	1,726959
-51,0347	4,949382	3,995863
.	.	.

Tablo 4.5 PC dönüşüm matrisi

Tablo 4.5’de PCA analizi uygulanarak elde edilen PC değerleri ile matrisin çarpımı sonucunda yeni matris değerleri gösterilmektedir.

1	2	3
-195,969	108,0507	0
-153,718	84,88602	0
-101,988	46,54626	0
-60,2622	17,9244	0
-49,8009	6,774175	0
.	.	.

Tablo 4.6 Üçüncü boyut sıfır

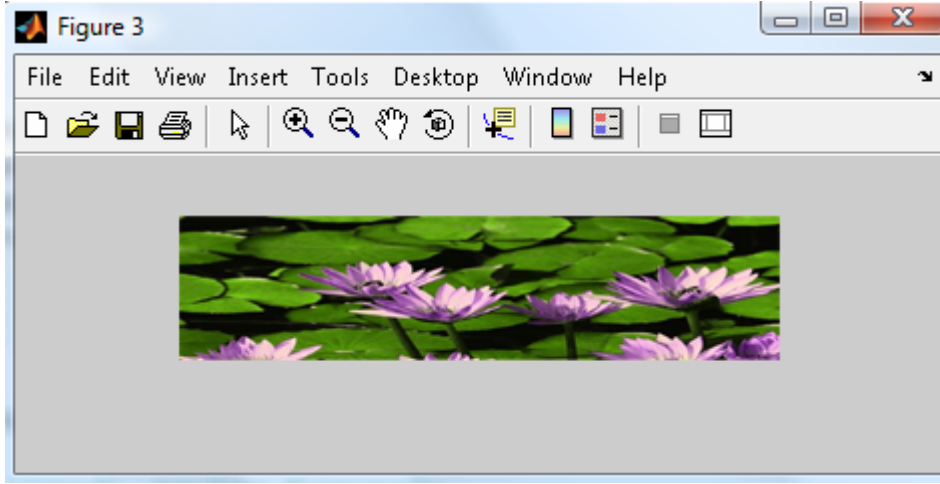
Tablo 4.6’da büyükten küçüğe doğru olan yoğunluktan en az yoğunluğun olduğu üçüncü boyut ihmal edilerek sıfır değerleri aktarılmış olan yeni matris görülmektedir. Bu işlem ile, görüntüde mevcut olan renkleri azaltarak yeni görüntü elde edebilir duruma geliyoruz.

Tablo 4.6’daki PC1 ve PC2 değerleri küçükten büyüğe doğru ayrı ayrı sıralanarak tekrar eşlenirler. Bu yeni değerler, $4 \times 12 = 48$, $3 \times 8 = 24$, $2 \times 6 = 12$ lik alanlara ayrılırlar. Ve bu alanların orta noktalarındaki değerler alınarak, bu değerler pc değerlerinin tersi ile çarpılarak RGB renkleri elde edilir ve böylece 48, 24 ve 12’lik renk paletleri elde edilmiş olunur. Burada elde edilen renk paletleri, access veri tabanına aktarılarak C#’da geliştirilen uygulamada kullanılabilir.

Columns 1 through 17																
130	102	67	39	32	33	34	32	32	32	32	32	32	32	32	32	32
63	31	14	10	10	7	7	7	7	7	7	7	7	7	7	7	7
24	5	7	10	7	6	6	7	7	7	7	7	7	7	7	7	7
20	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
20	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
20	7	7	7	8	8	8	8	8	7	9	9	8	8	8	9	9
21	7	7	7	8	8	7	7	4	4	6	8	7	5	4	7	11
23	10	11	10	13	15	18	20	22	21	18	16	22	40	48	37	22
55	49	58	65	74	74	78	81	82	83	85	88	100	118	134	138	122
81	75	78	79	78	77	75	76	75	74	77	81	86	85	87	90	92
85	76	75														

Tablo 4.7 Renk azaltılmış matris

Tablo 4.7’de PCA uygulandıktan sonra elde edilen renk azaltılmış matris görülmektedir. Bu matris kullanılarak elde edilen yeni görüntü Şekil 4.10’da verilmektedir.



Şekil 4.10: PCA sonuc renk azaltılarak elde edilen görüntü



(a)

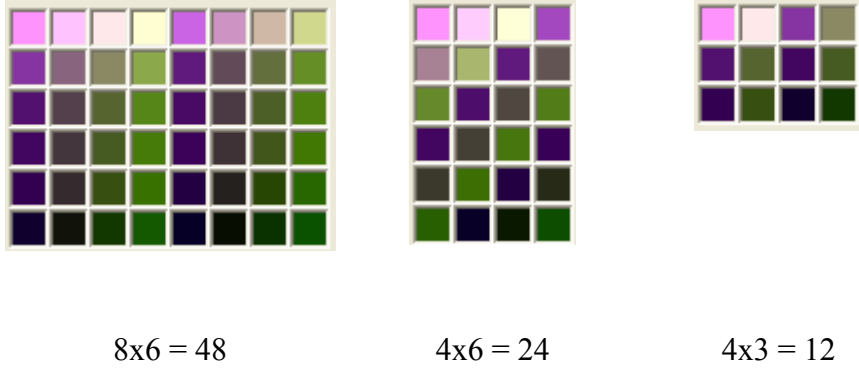


(b)

Şekil 4.11: Orjinal ve PCA sonucu elde edilen görüntüler

Şekil 4.11 (a)'da orjinal görüntü, (b)'de PCA sonucu elde edilen görüntü yer almaktadır. İki görüntünün karşılaştırılmasından da görüleceği gibi renk azaltılması yoluyla elde edilen görüntü, orjinaline çok benzemektedir. Bu sonuç bu tez kapsamında ileri sürülen tekniğin etkinliği konusunda iyi bir fikir vermektedir. Bu görüntüler için PSNR

değeri 32.0874 ($PSNR_R=28.7628$, $PSNR_G=34.6918$, $PSNR_B=32.8077$) olarak bulunmuştur. Bu değerin, bu tür uygulamalar için oldukça yüksek bir değer olduğu söylenebilir.



Şekil 4.12: PCA sonucu renk paletleri

Şekil 4.12’de PCA uygulaması sonucunda elde edilen renk paletleri verilmektedir.

4.2.3 Portrelerin PSNR değerleri

Şekil 4.13 (a), 4.14 (a) ve 1.5 (a)’da orjinal görüntü yer almaktadır. Orjinal görüntü üzerinde PCA analizi ile renk azaltma işlemi uygulandıktan sonra elde edilen yeni görüntü Şekil 4.13(b), 4.14(b) ve 4.15(b)’de orjinal görüntü boyutlarında verilmektedir. Orjinal ve işlenmiş görüntülere bakıldığında, PCA analizi sonucu renk azaltılarak elde edilen görüntünün orjinal görüntüye çok yakın olduğu görülmektedir. PSNR değerlerine bakıldığında sonucun oldukça başarılı olduğu söylenebilir. Renk azaltılarak elde edilen yeni görüntünün PSNR değerleri Tablo 4.8’de verilmektedir.



(a)



(b)

Şekil 4.13: Orjinal(Portre1) görüntü ve PCA sonucu görüntü



(a)



(b)

Şekil 4.14: Orjinal(Portre2) görüntü ve PCA sonucu görüntü



(a)



(b)

Şekil 4.15: Orjinal(Portre3) görüntü ve PCA sonucu görüntü

	$PSNR_r$	$PSNR_g$	$PSNR_b$	$PSNR_{ort}$
Portre1	45,2576	37,6632	40,6671	41,1960
Portre2	38,2511	34,8927	37,7065	36,9501
Portre3	34,9547	27,6979	30,1180	30,9235

Tablo 4.8Portreler PCA sonucu PSNR değerleri

Şekil 4.16 (a)'da yağlı boya tablonun orjinal görüntüsü yer almaktadır. Orjinal görüntü üzerinde PCA analizi ile renk azaltma işlemi uygulandıktan sonra elde edilen yeni görüntü Şekil 4.16(b)'de orjinal görüntü boyutlarında verilmektedir. Orjinal ve işlenmiş görüntülere bakıldığında, PCA analizi sonucu renk azaltılarak elde edilen görüntünün orjinal görüntüye çok yakın olduğu görülmektedir. PSNR değerleri Tablo 4.9'de verilmektedir.



Şekil 4.16: (a) Orjinal görüntü



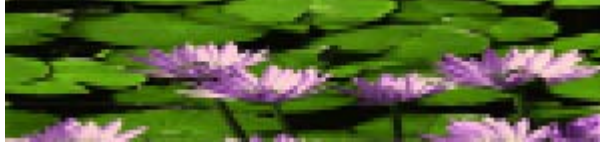
Şekil 4.16: (b) PCA sonucu görüntü

	PSNR _r	PSNR _g	PSNR _b	PSNR _{ort}
Tablo	38,6006	29,3769	30,7547	32,9107

Tablo 4.9Portreler PCA sonucu PSNR değerleri

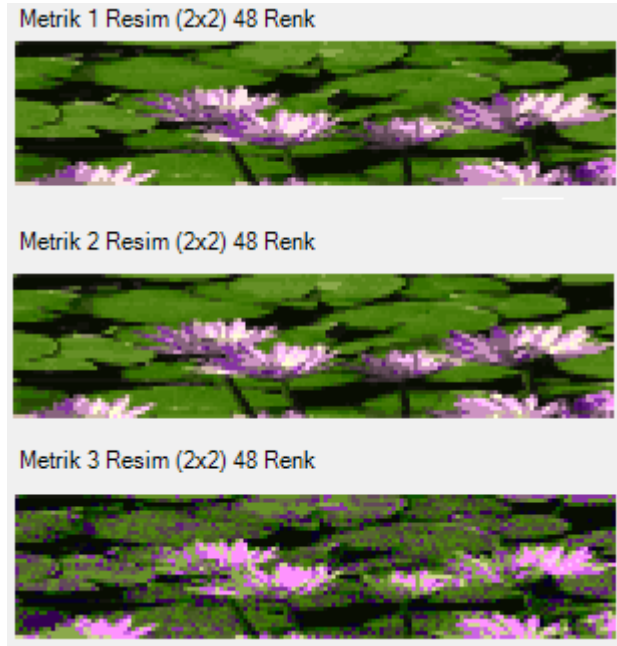
4.3 PCA'ın Uygulanması Sonucu Elde Edilen Paletler ile Referans Görüntü Üzerinde Yapılan İşlemler

PCA analizinin uygulanması sonucunda görüntüye göre renk paletleri oluşturulmuştur. Bölüm 4.1'de anlatılan Windows'un renk paletleri ile referans görüntü üzerinde metriklerin uygulanması işlemleri, PCA sonucu elde edilen paletler ile de referans görüntüye uygulanmıştır.

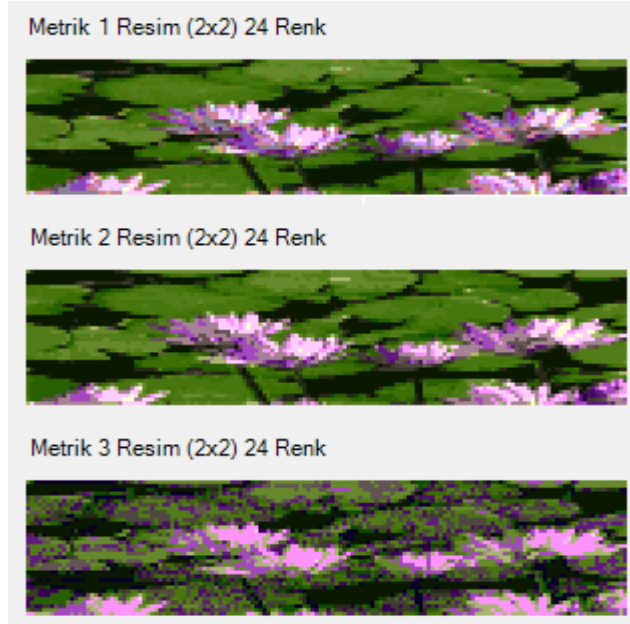


Şekil 4.17: PCA analizinde kullanılan Referans Görüntü

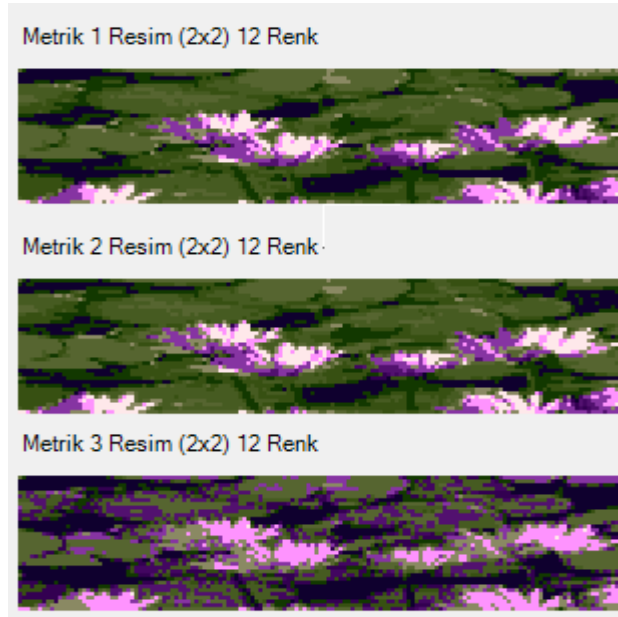
Şekil 4.17’de PCA analizi yardımı ile renk paletleri oluşturulan görüntünün 2x2’lik bloklar halindeki görüntüsü yer almaktadır. Tez çalışmamızda, bu görüntü referans görüntü olarak işleme alınacaktır. Bu referans görüntü üzerinde, PCA sonucu görüntüye göre elde edilen renk paletleri ile metrik uygulama işlemleri gerçekleştirilmiş ve Şekil 4.18, 4.19 ve 4.20’de sonuç görüntüleri verilmiştir.



Şekil 4.18: PCA sonrası 48 lik Palet ve Metrik Uygulama Sonuç Görüntüleri

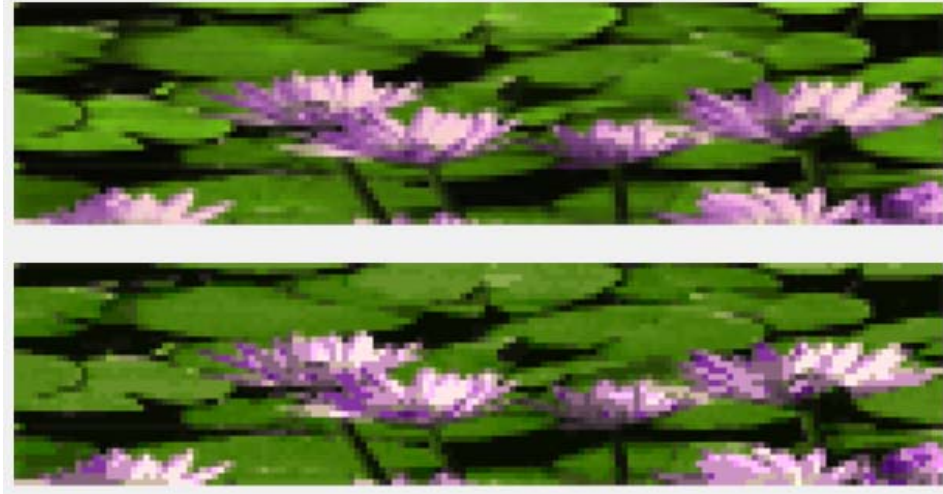


Şekil 4.19: PCA sonrası 24'lük Palet ve Metrik Uygulama Sonuç Görüntüleri



Şekil 4.20: PCA sonrası 12'lik Palet ve Metrik Uygulama Sonuç Görüntüleri

Görüntüler incelendiğinde ve psnr değerlerine bakıldığında en uygun görüntü, 48'lik renk paleti ve farkların karesi metriğine göre elde edilen görüntüdür. Şekil 4.21'de referans görüntü ve Şekil 4.17'deki görüntü büyütülerek verilmiştir.



Şekil 4.21: Referans görüntü ve 48'lik Palet ve Metrik Uygulama Sonuç Görüntüsü

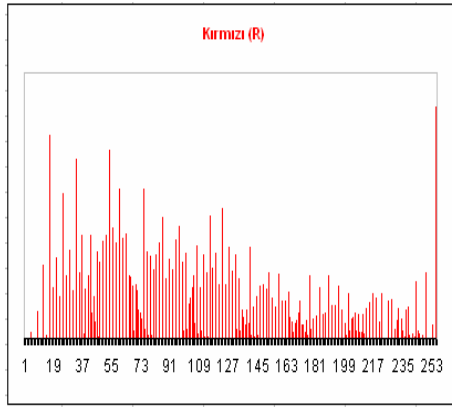
4.3.1 Görüntü histogramları

Görüntüler piksellerden oluşmakta ve bu piksellerde kırmızı, yeşil, mavi renklerin birleşimi bulunmaktadır, histogram bu renklerin ton dağılımını gösterir.

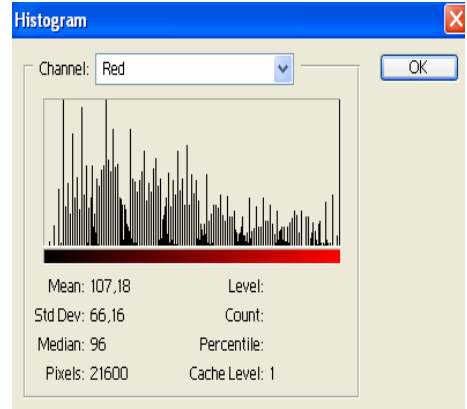
Tez çalışmasında; kullanılan görüntülerin herbir pikselinde yer alan renklerin içerisinde bulunan kırmızı, yeşil, mavi renklere göre histogramları elde edilmiştir.



Şekil 4.22: Referans Görüntü



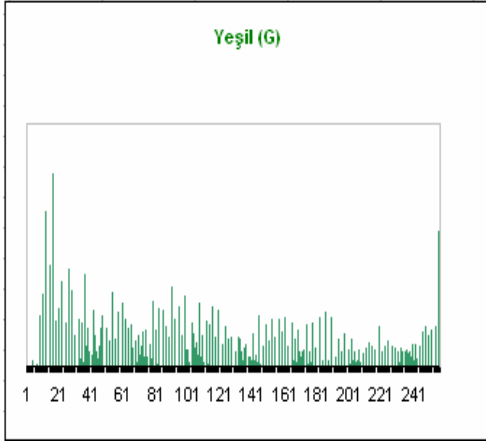
(a)



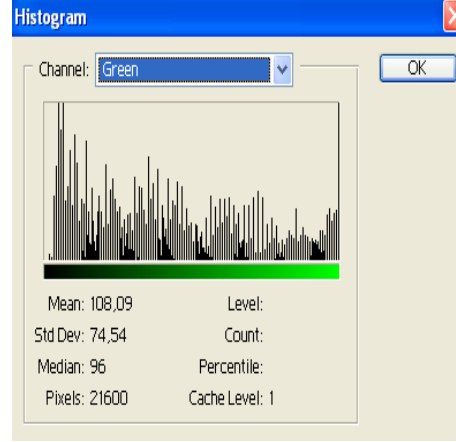
(b)

Şekil 4.23: Referans Görüntünün Kırmızı Renk H istogramı

Bu çalışma C# da geliştirildiği için görüntülerin histogramları da bir algoritma ile hesaplanmıştır. Elde edilen sonuçların doğruluğu test edilmiştir. Bunun için referans görüntünün bir başka grafik işleme programından kırmızı, yeşil ve mavi renk histogramları alınmıştır. Şekil Şekil 4.23 (a)'da algoritma ile elde edilen değerlerden elde edilen histogram, Şekil 4.23 (b)'de ise diğer bir görüntü işlem programından elde edilen kırmızı renk histogramıdır. Bu histogramlar da Şekil 4.22'deki referans görüntü kullanılmıştır.



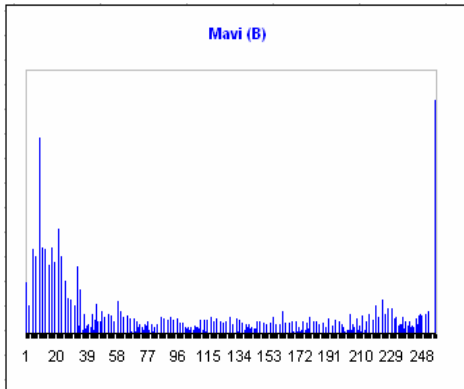
(a)



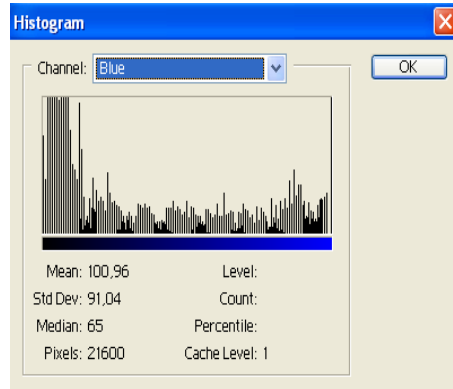
(b)

Şekil 4.24: Referans Görüntünün Yeşil Renk H istogramı

Şekil 4.24 (a)'da algoritma ile elde edilen değerlerden elde edilen histogram, Şekil 4.24 (b)'de ise diğer bir görüntü işlem programından elde edilen yeşil renk histogramıdır. Bu histogramlar da Şekil 4.22'deki referans görüntü kullanılmıştır.



(a)



(b)

Şekil 4.25: Referans Görüntünün Mavi Renk H istogramı

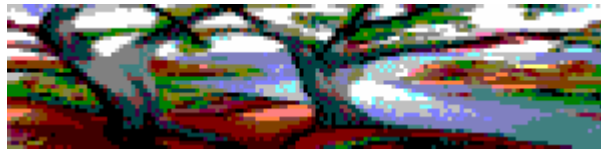
Şekil 4.25 (a)'da algoritma ile elde edilen değerlerden elde edilen histogram, Şekil 4.25 (b)'de ise diğer bir görüntü işlem programından elde edilen yeşil renk histogramıdır. Bu histogramlar da Şekil 4.22'deki referans görüntü kullanılmıştır.

```
long []histogram_Read = new long[256];
long []histogram_Green = new long[256];
long []histogram_Blue = new long[256];

for (int i = 0;i<5400;i++)
{
    histogram_Read[renkler_gecici_1[i]]++;
    histogram_Green[renkler_gecici_2[i]]++;
    histogram_Blue[renkler_gecici_3[i]]++;
}
```

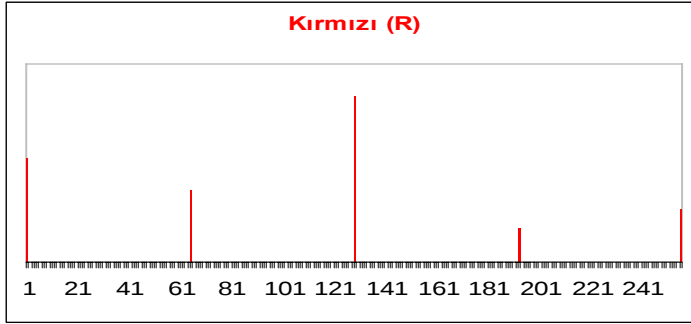
Tablo 4.10 Histogram Algoritması

Tablo 4.10'da uygulamada kullanılan histogram algoritması verilmiştir. Bu çalışmada, görüntüde yer alan bütün pikseller 5400 lük dizilere piksel piksel okunarak aktarılmıştır. Bu dizilerdeki renklerin toplamalarını kırmızı, yeşil ve mavi olarak bulabilmek için bir döngü oluşturulur. Piksellerin içersindeki değerler 0-255 aralığında olacağı için toplamaların tutulduğu değişken diziler 256 uzunluğunda tanımlanır.

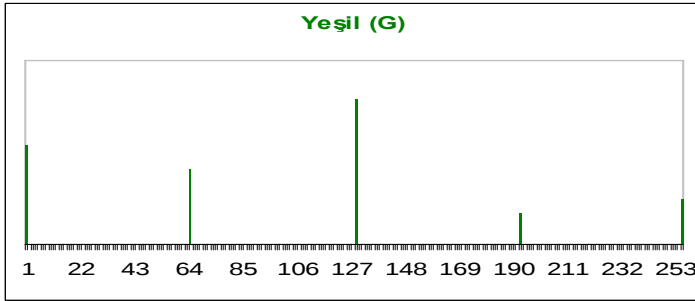


Şekil 4.26: En Küçük Kareler Toplamı Metriği ve Palet 48

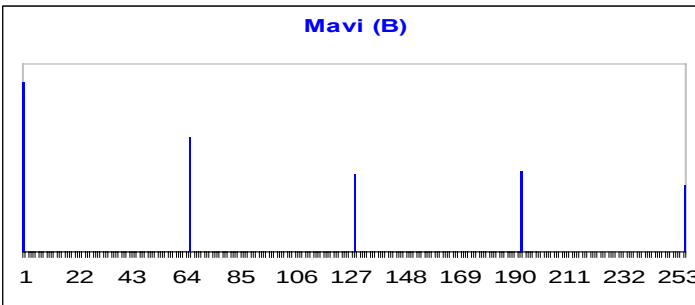
Şekil 4.26’da 48 renk içeren paleti ile en küçük kareler toplamı metriğinin uygulanmasından sonra elde edilen görüntü yer almaktadır. Bu görüntünün kırmızı renk histogramı Şekil 4.27 (a)’da, yeşil renk histogramı (b)’de, mavi renk histogramı (c)’de verilmiştir.



(a)



(b)



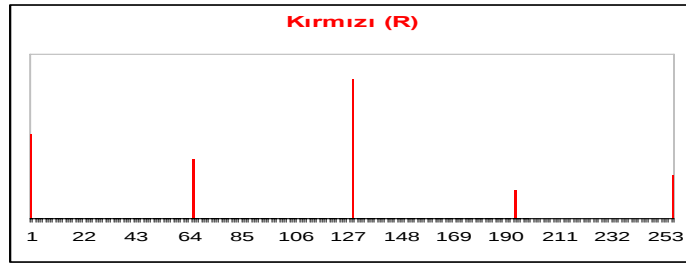
(c)

Şekil 4.27:Histogramı

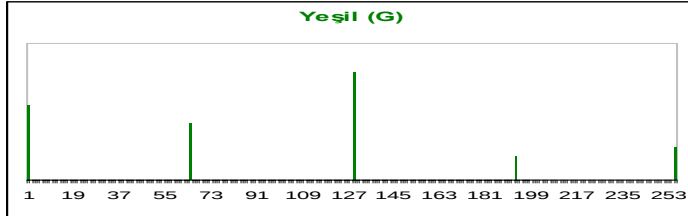


Şekil 4.28: En Küçük Mutlak Değerler Metriği ve Palet 48

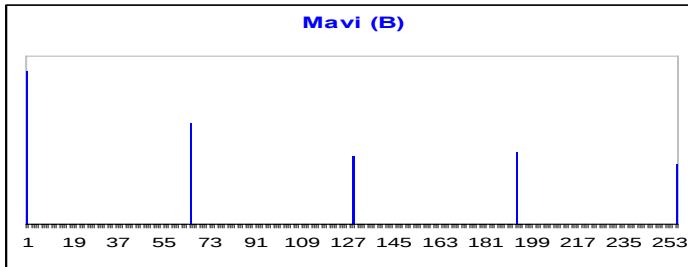
Şekil 4.28’de 48 renk içeren paleti ile en küçük mutlak değerler metriğinin uygulanmasından sonra elde edilen görüntü yer almaktadır. Bu görüntünün kırmızı renk histogramı Şekil 4.29 (a)’da, yeşil renk histogramı (b)’de, mavi renk histogramı (c)’de verilmiştir.



(a)

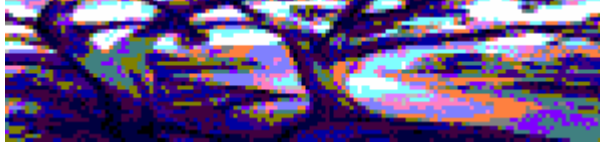


(b)



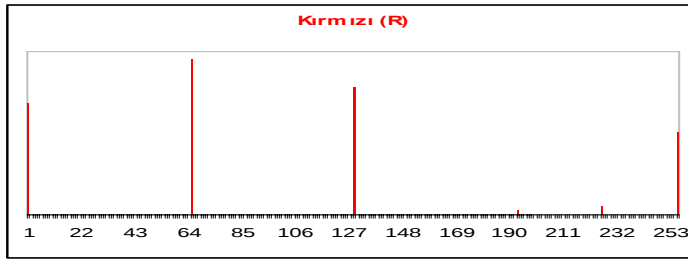
(c)

Şekil 4.29: Histogramı

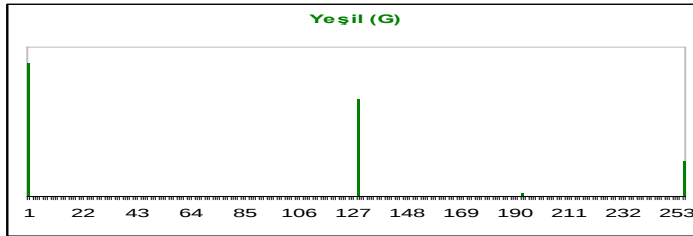


Şekil 4.30: En Küçük Ortalama Değerler Metriği ve Palet 48

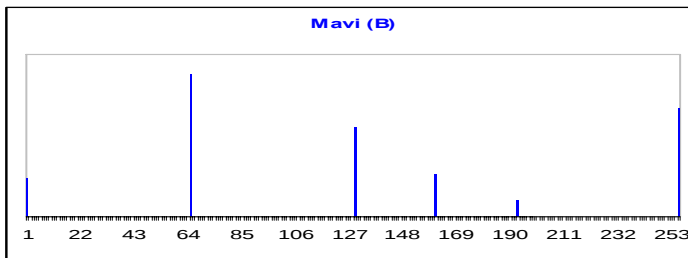
Şekil 4.30’da 48 renk içeren paleti ile en küçük ortalama değerler metriğinin uygulanmasından sonra elde edilen görüntü yer almaktadır. Bu görüntünün kırmızı renk histogramı Şekil 4.31 (a)’da, yeşil renk histogramı (b)’de, mavi renk histogramı (c)’de verilmiştir.



(a)



(b)



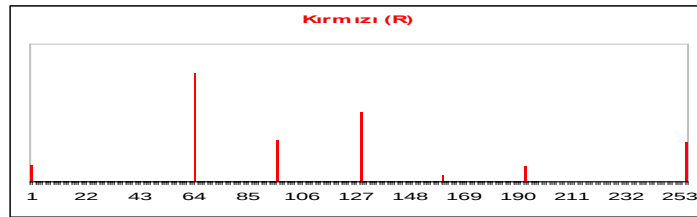
(c)

Şekil 4.31: Histogram

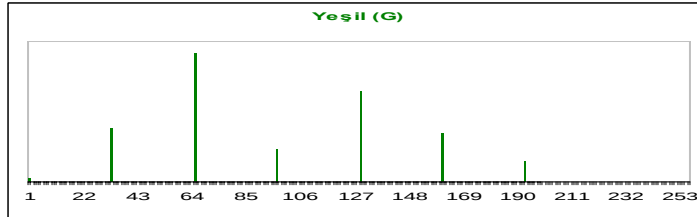


Şekil 4.32: En Küçük Kareler Toplamı Metriği ve Palet 24

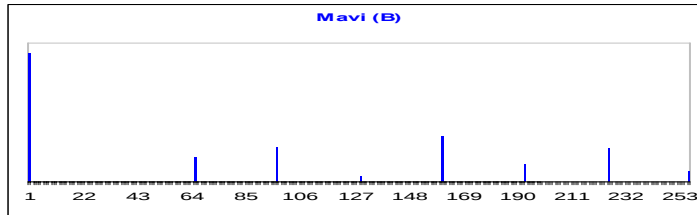
Şekil 4.32’de 24 renk içeren paleti ile en küçük kareler toplamı metriğinin uygulanmasından sonra elde edilen görüntü yer almaktadır. Bu görüntünün kırmızı renk histogramı Şekil 4.33 (a)’da, yeşil renk histogramı (b)’de, mavi renk histogramı (c)’de verilmiştir.



(a)



(b)



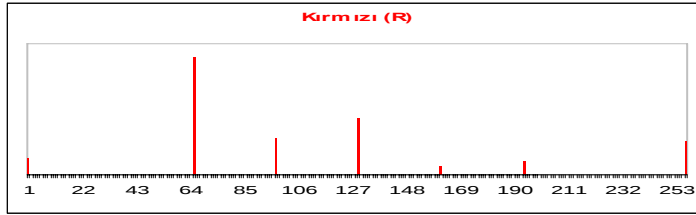
(c)

Şekil 4.33: En Küçük Ortalama Değerler Metriği ve Palet 48

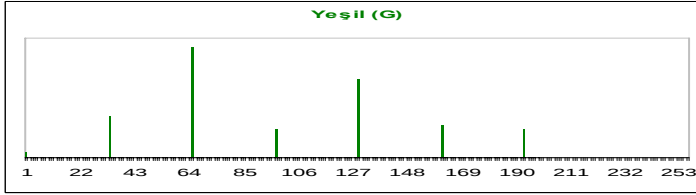


Şekil 4.34: En Küçük Mutlak Değerler Metriği ve Palet 24

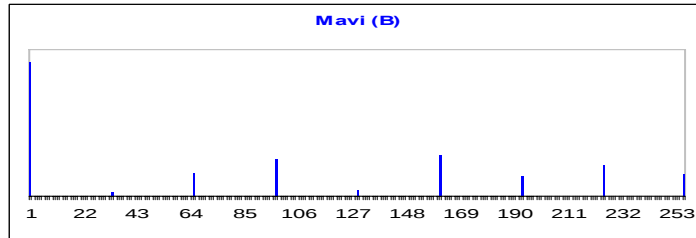
Şekil 4.34’de 24 renk içeren paleti ile en küçük mutlak değerler metriğinin uygulanmasından sonra elde edilen görüntü yer almaktadır. Bu görüntünün kırmızı renk histogramı Şekil 4.35 (a)’da, yeşil renk histogramı (b)’de, mavi renk histogramı (c)’de verilmiştir.



(a)



(b)



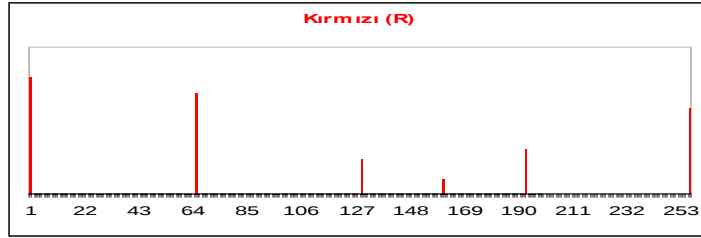
(c)

Şekil 4.35: En Küçük Mutlak Değerler Metriği ve Palet 24

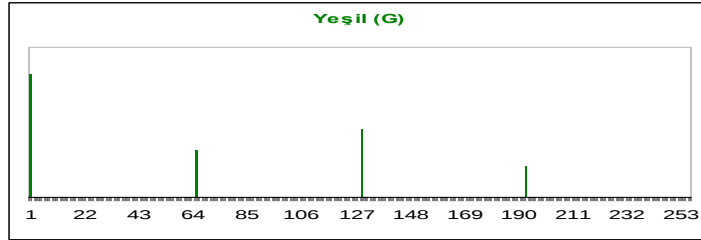


Şekil 4.36: Ortalama Değerler Metriği ve Palet 24

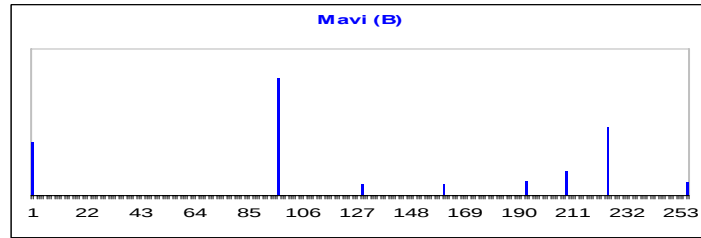
Şekil 4.36’da 24 renk içeren paleti ile en küçük ortalama değerler metriğinin uygulanmasından sonra elde edilen görüntü yer almaktadır. Bu görüntünün kırmızı renk histogramı Şekil 4.37 (a)’da, yeşil renk histogramı (b)’de, mavi renk histogramı (c)’de verilmiştir.



(a)



(b)



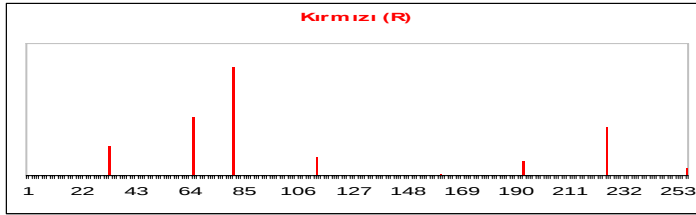
(c)

Şekil 4.37: Ortalama Değerler Metriği ve Palet 24

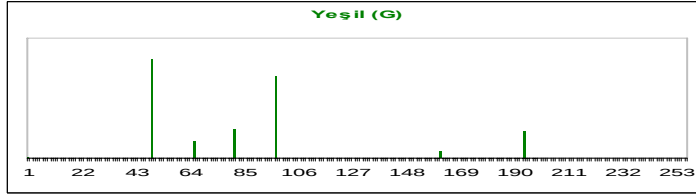


Şekil 4.38 En Küçük Kareler Toplamı Metriği ve Palet 12

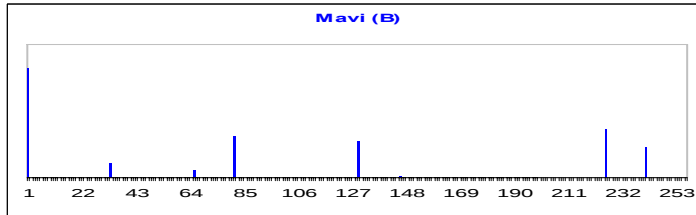
Şekil 4.38’de 12 renk içeren paleti ile en küçük ortalama değerler metriğinin uygulanmasından sonra elde edilen görüntü yer almaktadır. Bu görüntünün kırmızı renk histogramı Şekil 4.39 (a)’da, yeşil renk histogramı (b)’de, mavi renk histogramı (c)’de verilmiştir.



(a)

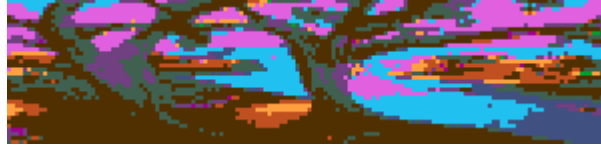


(b)



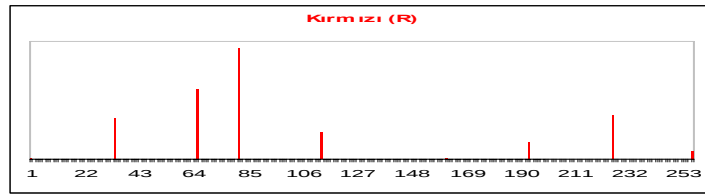
(c)

Şekil 4.39 Histogram

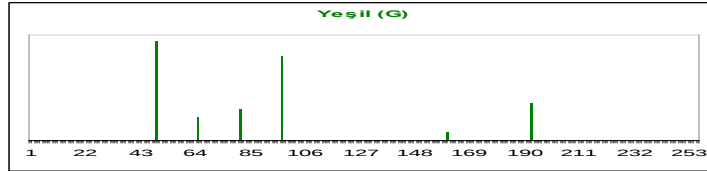


Şekil 4.40: En Küçük Mutlak Değerler Metriği ve Palet 12

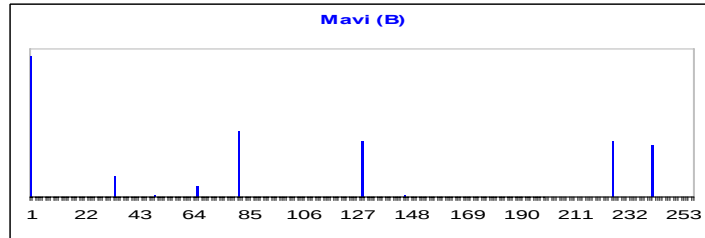
Şekil 4.40’da 12 renk içeren paleti ile en küçük ortalama değerler metriğinin uygulanmasından sonra elde edilen görüntü yer almaktadır. Bu görüntünün kırmızı renk histogramı Şekil 4.41 (a)’da, yeşil renk histogramı (b)’de, mavi renk histogramı (c)’de verilmiştir.



(a)

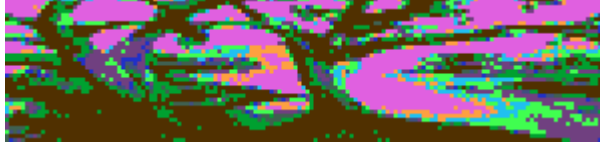


(b)



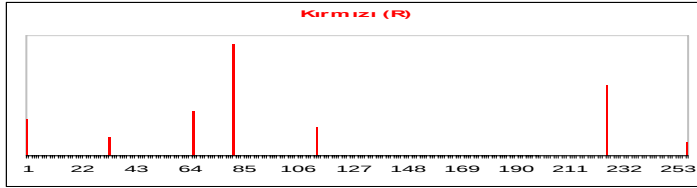
(c)

Şekil 4.41: Histogram

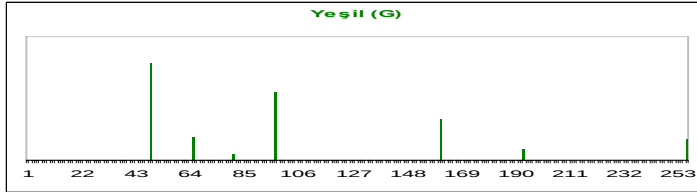


Şekil 4.42: Ortalama Değerler Metriği ve Palet 12

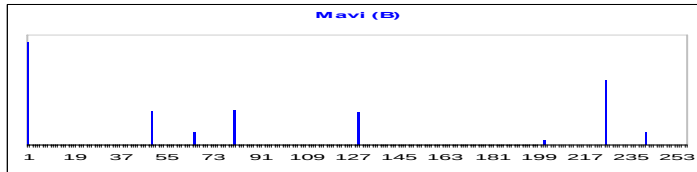
Şekil 4.42’de 12 renk içeren paleti ile en küçük ortalama değerler metriğinin uygulanmasından sonra elde edilen görüntü yer almaktadır. Bu görüntünün kırmızı renk histogramı Şekil 4.43 (a)’da, yeşil renk histogramı (b)’de, mavi renk histogramı (c)’de verilmiştir.



(a)



(b)



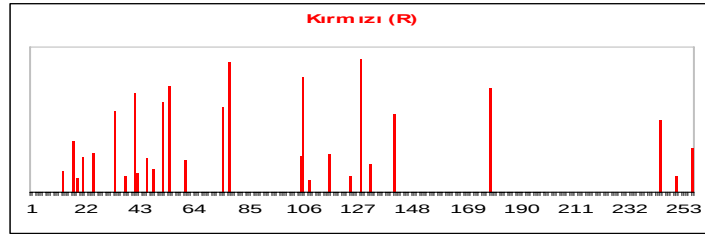
(c)

Şekil 4.43: Histogram

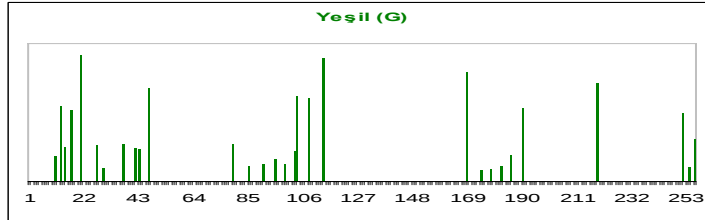


Şekil 4.44 En Küçük Kareler Toplamı Metriği ve Palet 32

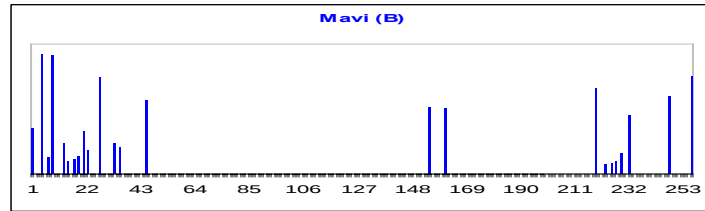
Şekil 4.44’de 32 renk içeren paleti ile en küçük ortalama değerler metriğinin uygulanmasından sonra elde edilen görüntü yer almaktadır. Bu görüntünün kırmızı renk histogramı Şekil 4.45 (a)’da, yeşil renk histogramı (b)’de, mavi renk histogramı (c)’de verilmiştir.



(a)



(b)



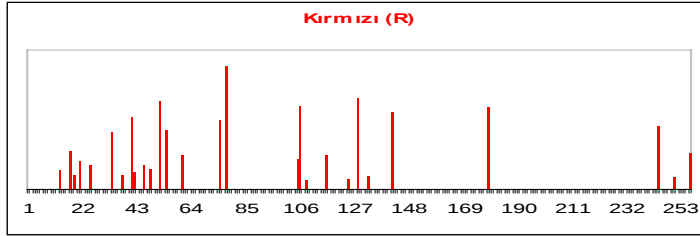
(c)

Şekil 4.45 Histogram

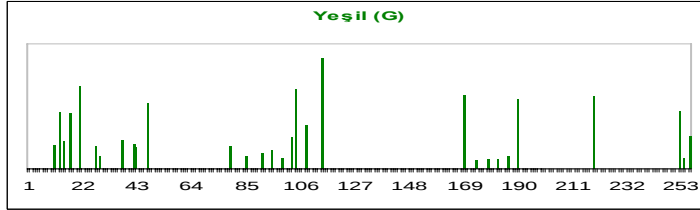


Şekil 4.46: En Küçük Mutlak Değerler Metriği ve Palet 32

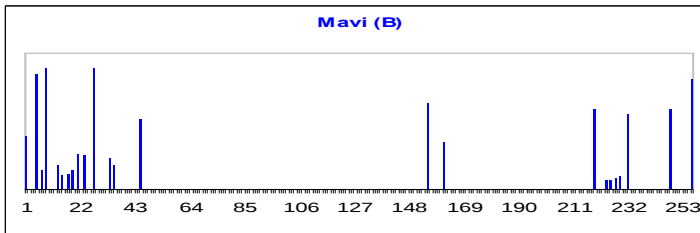
Şekil 4.46’da 32 renk içeren paleti ile en küçük ortalama değerler metriğinin uygulanmasından sonra elde edilen görüntü yer almaktadır. Bu görüntünün kırmızı renk histogramı Şekil 4.47 (a)’da, yeşil renk histogramı (b)’de, mavi renk histogramı (c)’de verilmiştir.



(a)



(b)



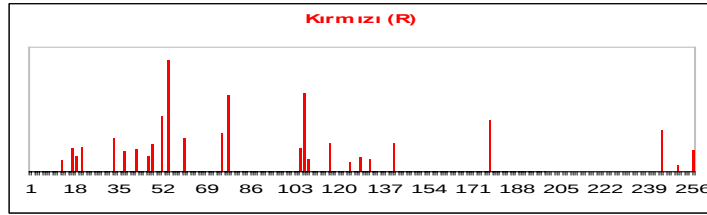
(c)

Şekil 4.47: Histogram

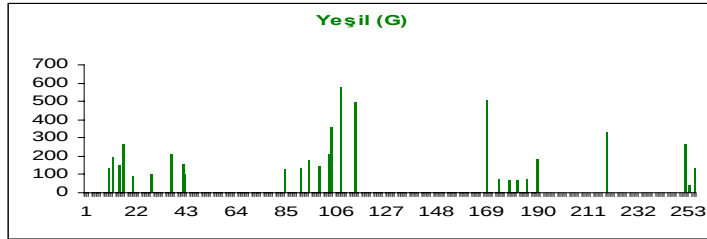


Şekil 4.48: Ortalama Değerler Metriği ve Palet 32

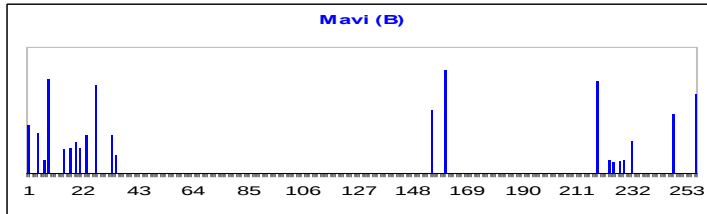
Şekil 4.48’de 32 renk içeren paleti ile en küçük ortalama değerler metriğinin uygulanmasından sonra elde edilen görüntü yer almaktadır. Bu görüntünün kırmızı renk histogramı Şekil 4.49 (a)’da, yeşil renk histogramı (b)’de, mavi renk histogramı (c)’de verilmiştir.



(a)



(b)



(c)

Şekil 4.49: Histogram

5 KULLANILAN METRİKLER

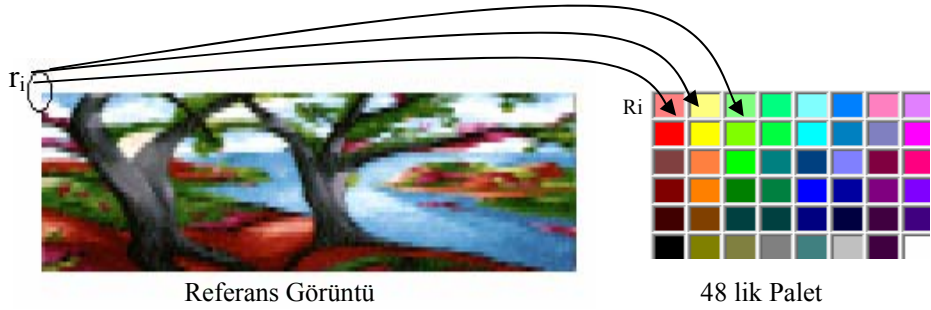
Referans görüntü üzerinde renk ayarlamaları metrikler kullanılmıştır. Bu metrikler referans görüntüdeki her pikselde bulunan renklerle üç farklı renk paletindeki (48, 24, 12) renkler arasında uygulanmış ve referans görüntüyü paletlere uydurmak için elde edilen renklerle piksellere işlenerek yeni görüntüler oluşturulmuştur.

5.1 Blokların Eşleştirilmesinde Kullanılan Metrikler

Bu uygulama üç farklı metrik ve üç farklı renk paleti kullanılmıştır. Bu yüzden referans görüntüden Windows'un renk paletleri için dokuz farklı yeni görüntü elde edilmiştir. Bu işlemler için, Windows'un 48, 24 ve 12'lik üç farklı renk paleti ve PCA'nın uygulanması sonucu görüntünün kendi renklerinden oluşturulan 48, 24 ve 12'lik üç farklı renk paleti kullanılmıştır.

5.1.1 En küçük kareler toplamı

Referans görüntünün ilk pikselindeki renk alınıp, 48'lik renk paletinin tüm renkleri ile kırmızı, yeşil ve mavi (RGB) değerlerinin farklarının kareleri toplamı bulunmuştur. Ve çıkan sonuçlardan minimum olan renk seçilip yeni oluşacak görüntünün ilk pikseline yerleştirilmiştir. Bu işlem 150x36'lık referans görüntüsünün 5400 pikselinin herbiri için uygulanmıştır.



Şekil 5. 1: Piksel değerlerin Paletle işleme alınması

$$S_i^{(1)} = (r_i - R_i)^2 + (g_i - G_i)^2 + (b_i - B_i)^2 \quad (5.1)$$

$$\Rightarrow \min_i \sum S_i^{(1)}$$

Örnek olarak 48'lik renk paletini verirsek; referans görüntüdeki ilk pikseli alıp 48 renk ile kareler toplamını buluyoruz. Her piksel değeri için kırk sekiz adet S_i değeri elde ediyoruz. Bu 48 adet değerden en küçüğünü alıyoruz ve bu rengin palettteki en yakın rengi seçilip tekrar ilgili piksele aktarılıyor. Bizim uygulamamızda toplam 5400 adet piksel olduğu için 5400 defa 48 lik S_i değerleri elde edilip minimumu alınıyor.

r	255	r	...
g	0	g	...
b	64	b	...

(a)

r	128	r	128	
g	64	g	192	
b	255	b	64	
r	0	r	255	
g	124	g	255	
b	255	b	64	

(b)

Şekil 5. 2: Metrik Hesaplaması

Şekil 5.2 (a) da referans görüntüdeki pikseller içerisindeki değerler örneği verilmiştir. Şekil 5.2 (b) de palet 48 deki renkler örneği gösterilmiştir.

Şekil 5.2 deki değerler dikkate alınıp formül 5.1 uygulanarak aşağıdaki hesaplanmalar yapılmıştır.

$$\begin{aligned}
 (255-128)^2 + (0-64)^2 + (64-255)^2 &= 56796 \\
 (255-128)^2 + (0-192)^2 + (64-64)^2 &= 18433 \\
 (255-0)^2 + (0-124)^2 + (64-255)^2 &= 116882 \\
 (255-255)^2 + (0-255)^2 + (64-64)^2 &= 65025
 \end{aligned}$$

Bu değerlerin en küçüğünü alırsak 18433 dür, palettteki renklerden en yakın renk birinci satır ikinci sütundaki (128, 192, 64) renktir. Bu renk ilgili piksellere işlenmiştir.

5.1.2 En küçük mutlak değerler

$$\begin{aligned}
 S_i^{(2)} &= |r_i - R_i| + |g_i - G_i| + |b_i - B_i| \\
 \Rightarrow \min_i \sum S_i^{(2)}
 \end{aligned} \tag{5.2}$$

Şekil 5.2 deki değerler dikkate alınıp formül 5.2 uygulanarak aşağıdaki hesaplanmalar yapılmıştır.

$$\begin{array}{rclclcl}
|255-128| & + & |0-64| & + & |64-255| & = & 382 \\
|255-128| & + & |0-192| & + & |64-64| & = & 319 \\
|255-0| & + & |0-124| & + & |64-255| & = & 570 \\
|255-255| & + & |0-255| & + & |64-64| & = & 255
\end{array}$$

Bu değerlerin en küçüğü 255 dir, palettteki renklerden en yakın renk ikici satır ikinci sütundaki (255, 255, 64) renktir. Bu renk ilgili piksellere işlenmiştir.

5.1.3 En küçük ortalama değerler

$$\begin{aligned}
S_i^{(3)} &= \left| \left(\frac{(r_i + g_i + b_i)}{3} \right) - \left(\frac{(R_i + G_i + B_i)}{3} \right) \right| \quad (5.3) \\
\Rightarrow \min_i \sum S_i^{(3)}
\end{aligned}$$

Şekil 5.2 deki değerler dikkate alınıp formül 5.3 uygulanarak aşağıdaki hesaplanmalar yapılmıştır.

$$\begin{aligned}
\left| \frac{(255 + 0 + 64)}{3} - \frac{(128 + 64 + 255)}{3} \right| &= 42.67 \\
\left| \frac{(255 + 0 + 64)}{3} - \frac{(128 + 192 + 64)}{3} \right| &= 78.33 \\
\left| \frac{(255 + 0 + 64)}{3} - \frac{(0 + 124 + 255)}{3} \right| &= 20 \\
\left| \frac{(255 + 0 + 64)}{3} - \frac{(255 + 255 + 64)}{3} \right| &= 85
\end{aligned}$$

Bu değerlerin en küçüğü 20 dir, paletteki renklerden en yakın renk ikici satır birinci sütundaki (0, 124, 255) renktir.

5.2 Görüntülerin Karşılaştırılmasında Kullanılan Metrikler

Referans görüntü ile referans görüntüden elde edilen yeni görüntülerin karşılaştırılmasında RMSE ve PSNR metrikleri kullanılmıştır.

5.2.1 RMSE

$$RMSE_r = \sqrt{\frac{1}{n} \sum_{i=1}^n (r_i - R_i)^2} \quad (5.4)$$

$r_i \rightarrow$ referans görüntünün kırmızı renkleri

$R_i \rightarrow$ Metrikler uygulanarak elde edilen görününün kırmızı renkleri

$n \rightarrow$ Görüntülerde toplam piksel sayısı

$$RMSE_g = \sqrt{\frac{1}{n} \sum_{i=1}^n (g_i - G_i)^2} \quad (5.5)$$

$g_i \rightarrow$ referans görüntünün kırmızı renkleri

$G_i \rightarrow$ Metrikler uygulanarak elde edilen görününün kırmızı renkleri

$n \rightarrow$ Görüntülerde toplam piksel sayısı

$$RMSE_b = \sqrt{\frac{1}{n} \sum_{i=1}^n (b_i - B_i)^2} \quad (5.6)$$

$b_i \rightarrow$ referans görüntünün kırmızı renkleri

$B_i \rightarrow$ Metrikler uygulanarak elde edilen görüntünün kırmızı renkleri

$n \rightarrow$ Görüntülerde toplam piksel sayısı

Referans görüntü ile elde edilen görüntü arasında yapılan karşılaştırmalar için formül (5.4), (5.5), (5.6) kullanılarak RMSE değerleri hesaplanmış ve Tablo 5.1, 5.2, 5.3, 5.4 de gösterilmiştir.

RMSE_r	RMSE_g	RMSE_b	RMSE_{ort}
31	25	22	26
31	25	22	26
60	51	74	62

Tablo 5.1 Palet 48'e göre RSME değerleri

RMSE_r	RMSE_g	RMSE_b	RMSE_{ort}
33	46	30	36
40	44	28	37
69	67	80	72

Tablo 5.2 Palet 24'e göre RSME değerleri

RMSE_r	RMSE_g	RMSE_b	RMSE_{ort}
43	63	28	45
49	60	29	46
64	79	48	64

Tablo 5.3 Palet 12'e göre RSME değerleri

$RMSE_r$	$RMSE_g$	$RMSE_b$	$RMSE_{ort}$
27	17	29	24
29	17	31	25
44	28	37	36

Tablo 5.4 Palet 32'göre RSME değerleri

5.2.2 PSNR

Referans görüntü ile elde edilen görüntü arasında yapılan karşılaştırmalar için formül (5.4), (5.5), (5.6) kullanılarak RMSE değerleri hesaplanmış ve bu değerler formül (5.7), (5.8), (5.9) daki formüllerde kullanılarak PSNR değerleri bulunmuştur.

$$PSNR_r = 20 \log \left(\frac{255}{RMSE_r} \right) \quad (5.7)$$

255 → görüntü değerinin alabileceği en yüksek değer.

$RMSE_r$ → Kırmızı rengin RMSE ölçümü

$$PSNR_g = 20 \log \left(\frac{255}{RMSE_g} \right) \quad (5.8)$$

$RMSE_g$ → Yeşil rengin RMSE ölçümü

$$PSNR_b = 20 \log \left(\frac{255}{RMSE_b} \right) \quad (5.9)$$

$RMSE_b$ → Mavi rengin RMSE ölçümü

Referans görüntüler ile metrik uygulamaları sonucunda elde edilen görüntüler arasındaki karşılaştırma PSNR'ları aşağıdaki tablolarda verilmiştir.

	PSNR_r	PSNR_g	PSNR_b	PSNR_{ort}
Metrik1	18,062	20,000	20,828	19,630
Metrik2	18,062	20,000	20,828	19,630
Metrik3	12,041	13,979	9,542	11,854

Tablo 5.5 Palet 48'e göre PSNR değerleri

Tablo 5.5'de 48 renkli palete göre kırmızı, yeşil, mavi ve bunların ortalamasından oluşan PSNR değerleri verilmiştir. Metrik1 ve metrik2 kullanılarak elde edilmiş olan görüntülerin PSNR değerleri birbirine eşittir, metrik3 e göre elde edilmiş olan görüntünün PSNR değerleri düşüktür. Görüntünün kalitesine bakacak olursak, PSNR değeri düşüncce kalitede düşüyor. En uygun görüntü metrik1 ve 2 e göre elde edilmiş olan görüntüdür.

	PSNR_r	PSNR_g	PSNR_b	PSNR_{ort}
Metrik1	16,902	13,979	18,062	16,314
Metrik2	15,563	13,979	19,085	16,209
Metrik3	9,542	9,542	9,542	9,542

Tablo 5.6 Palet 24'e göre PSNR değerleri

Tablo 5.6'de 24 renkli palete göre PSNR değerleridir. Bu tabloda da metrik1 ve metrik2 kullanılarak elde edilmiş olan görüntülerin PSNR değerleri birbirine yakın, metrik3 e göre elde edilmiş olan görüntünün PSNR değerleri düşüktür. Kaliteli görüntü için PSNR değeri yüksek olan metrik1'e göre hesaplanmış birinci görüntü alınır.

	PSNR_r	PSNR_g	PSNR_b	PSNR_{ort}
Metrik1	13,979	12,041	19,085	15,035
Metrik2	13,979	12,041	18,062	14,694
Metrik3	9,542	9,542	13,979	11,021

Tablo 5.7 Palet 12'e göre PSNR değerleri

Tablo 5.7'da 12 renkli palete göre PSNR değerleridir. Diğer iki tabloda olduğu gibi, bu tabloda da metrik1 ve metrik2 kullanılarak elde edilmiş olan görüntülerin PSNR değerleri birbirine yakın, metrik3'e göre elde edilmiş olan görüntünün PSNR değerleri düşüktür. Kaliteli görüntü için PSNR değeri yüksek olan metrik1'e göre hesaplanmış birinci görüntü alınır.

Tablo 5.5, 5.6, 5.7'deki değerleri ayrı ayrı incelenmiş ancak bu tez çalışmasında bu üç tablonun birleşiminden bir karar verilmektedir. Toplam dokuz görüntünün PSNR değerlerine bakılıp en uygun olan görüntü seçilir. Bu bilgiler tablo 5.8'de gösterilmiştir.

Metrik	PSNR_r	PSNR_g	PSNR_b	PSNR_{ort}
48/1	18,062	20,000	20,828	19,630
48/2	18,062	20,000	20,828	19,630
48/3	12,041	13,979	9,542	11,854
24/1	16,902	13,979	18,062	16,314
24/2	15,563	13,979	19,085	16,209
24/3	9,542	9,542	9,542	9,542
12/1	13,979	12,041	19,085	15,035
12/2	13,979	12,041	18,062	14,694
12/3	9,542	9,542	13,979	11,021

Tablo 5.8 PSNR değerleri

Tablo 5.8'deki referans görüntüden oluşturulan toplam dokuz adet görüntünün PSNR değerleri üzerinden en uygun görüntü seçimi 19,630 PSNR ortalama değerine sahip olan görüntüdür.

	PSNR_r	PSNR_g	PSNR_b	PSNR_{ort}
Metrik1	19,085	23,522	18,062	20,223
Metrik2	18,062	23,522	18,062	19,882
Metrik3	13,979	19,085	15,563	16,209

Tablo 5.9 Palet 32'e göre PSNR değerleri

Tablo 5.9'da 32 renk paletine göre PSNR değerleri verilmiştir. Metrik1 e göre PSNR değeri en yüksektir. Tablo 5.9'da yer alan üç görüntünün PSNR değerleri üzerinden en uygun görüntü seçimi 20,223 PSNR ortalama değerine sahip olan görüntüdür.

Bölüm 4.3'de anlatılan referans görüntüye, PCA analizi uygulanması sonucu görüntü renklerinden oluşan 48, 24 ve 12'lik renk paletleri ile metrikler uygulanarak elde edilen görüntülerin PSNR değerleri Tablo 5.10'da verilmiştir. Tablo incelendiğinde, 48 renkten oluşan palet ile en küçük kareler toplamı metriğinin uygulanmasından sonra elde edilen görüntünün PSNR değeri 27,835 olarak bulunmuştur. Bu değer, görüntü işleme uygulamaları için iyi olarak değerlendirilebilir.

Metrik	PSNR_r	PSNR_g	PSNR_b	PSNR_{ort}
48/1	29,827	26,444	27,235	27,835
48/2	29,827	26,444	26,444	27,572
48/3	29,827	13,979	15,563	19,790
24/1	27,235	23,522	23,522	24,759
24/2	27,235	23,522	23,522	24,759
24/3	27,235	13,979	13,979	18,398
12/1	24,609	18,062	19,085	20,585
12/2	24,609	16,902	19,085	20,199
12/3	25,105	12,041	12,041	16,396

Tablo 5.10 PCA'ın uygulama sonucu PSNR değerleri

6 SORUNLU BÖLGELERİN TEŞHİSİ

Bu tez çalışmasında geliştirilen uygulama farklı görüntüler için çalıştırılmıştır. Bayrak görüntüsündeki ay-yıldız kenarlarında bozulmalar görülmüştür.

Bu sorunun giderilmesi için görüntünün kendi renkleri üzerinden palet oluşturulur. Görüntü piksel olarak okunur ve en çok kullanılan renk pikselleri büyükten küçüğe sıralanır. İlk 32 renk alınır ve palet oluşturulur.

Metrik	RMSE _r	RMSE _g	RMSE _b	RMSE _{ort}
48/1	13	9	5	9
48/2	13	9	4	9
48/3	110	113	21	81
24/1	48	62	9	40
24/2	48	62	8	39
24/3	53	62	21	45
12/1	48	79	31	53
12/2	48	79	31	53
12/3	169	87	101	119
32/1	2	1	2	1
32/2	2	1	2	1
32/3	8	7	3	6

Tablo 6.1 RMSE değerleri

Tablo 6.1 de bayrak görüntüsünün RMSE değerleri verilmiştir. Bu değerlere göre karşılaştırma yapıldığında en yüksek PSNR değeri, 32 renk paleti ile elde edilmiştir.

Tablo 6.2 de bayrak görüntüsünün PSNR değerleri verilmiştir. Bu değerlere göre karşılaştırma yapıldığında en yüksek PSNR değeri, 32 renk paleti ile elde edilmiştir.

Metrik	PSNR_r	PSNR_g	PSNR_b	PSNR_{ort}
48/1	25,575	28,943	34,151	29,557
48/2	25,575	28,943	35,987	30,168
48/3	6,021	6,021	21,584	11,208
24/1	13,979	12,041	28,943	18,321
24/2	13,979	12,041	29,827	18,616
24/3	12,041	12,041	21,584	15,222
12/1	13,979	9,542	18,062	13,861
12/2	13,979	9,542	18,062	13,861
12/3	0,000	6,021	6,021	4,014
32/1	42,076	48,131	42,076	44,094
32/2	42,076	48,131	42,076	44,094
32/3	29,827	31,126	38,588	33,181

Tablo 6.2 PSNR değerleri

7 SONUÇ

Bu tez çalışmasında her görüntü piksel piksel okunup bu pikseller üzerinde işlemler yapılarak yeni görüntüler elde edilmiştir. Bu yeni görüntüler referans görüntüde Windows'un 48, 24, 12'lik renk paletleri ile en küçük kareler toplamı, en küçük mutlak değerler, en küçük ortalama değerler metrikleri bulunarak piksellere işlenerek elde edilmiştir.

Bu çalışmada öncelikle Windows'un renk paletleri kullanılmıştır. Bu renk paletleri Windows'un 48 basit renkten oluşan palet alınarak, renklerin ortalaması ile 24 ve 12'lik renk paletleri oluşturulmuştur. Elde edilen 24 ve 12'lik renk paletlerinde beyaz renk kaybolmuştur. Bu yüzden beyaz renk sabit tutularak 24 ve 12'lik renk paletleri tekrar elde edilerek işleme alınmıştır. Çalışmamızın esasını oluşturan renk azaltma işlemi için ana bileşenler analizi(PCA) kullanılarak, görüntünün kendi renklerinden oluşan 48, 24 ve 12'lik renk paletleri oluşturulmuştur.

Bu uygulama .Net C#'da geliştirilmiştir. C# grafik konusunda sınıflar ve bu sınıflara ait yöntemler sunmaktadır. Örneğin bir görüntünün bir pikselini okumak için getpixel metodu kullanılır. Bu metotta pikselin bulunduğu konum (x,y) koordinatları şeklinde verilir. Renk değişkeni RGB şeklinde piksel döndürülür. Üzerinde işlem yapılır ve setpixel ile (x,y) koordinatları ile konum belirlenip işlenir. Böylece bir görüntü üzerinde işlem yapmak kolaylaşır.

Uygulamada, ana bileşenler analizi için istatistik uygulamalar için kullanılan Matlab 7.1'de çalışılmıştır. Matlab, matrisler üzerinde hızlı işlem yapabilme özelliğine sahiptir. Bu da PCA çalışmamızda bize kolaylık ve hız kazandırmıştır.

Uygulamada kullanılan görüntü dosyaları “.tif” uzantılıdır. Referans görüntü üzerinde paletler ile metriklerin uygulanması sonucu elde edilen görüntülerde bilgisayara “.tif” uzantılı olarak kaydedilmektedir.

Uygulama referans ve referans üzerinden elde edilen görüntülerin arasında RMSE ve PSNR değerleri bulunmuş ve değerler uzantısı “.txt” yada “.xls” dosyalarına kaydedilmiştir.

Bölüm 5’de anlatılan metrikler ile referans görüntü Windows’un 48, 24, 12’lik paletleri ve PCA analizi sonucu elde edilen 48, 24, 12’lik paletleri ile uygulanır ve farklı görüntüler elde edilir. Bu görüntüler üzerinde blok karşılaştırma yapılır ve en uygun görüntü seçimi başarılı bir şekilde gerçekleştirilir. Görüntü karşılaştırma RMSE in bir fonksiyonu olan PSNR değerleri ile yapılır.

Renk azaltma işlemi için kullanılan PCA analizi ile çeşitli görüntüler üzerinde çalışma yapılmış ve çok iyi sonuçlar elde edilmiştir. PSNR değerlerine bakıldığında renk azaltma konusunda oldukça başarılı sonuçlar elde edilmiştir.

Bölüm 4’de anlatılan paletler dışında görüntünün kendi renklerinden C#’da geliştirilen algoritma ile 32 renkli palet oluşturulur. Görüntü piksel piksel okunur ve renklerin görüntüde kaç kez kullanıldığı hesaplanır. Elde edilen değerlerden en çok kullanılan 32 renk alınarak palet oluşturulur.

Referans görüntü ile 32 renkli palet arasında bölüm 5’de anlatılan metrikler uygulanır üç adet görüntü elde edilir. Bu görüntülerin PSNR değerlerine bakılır ve en yüksek PSNR değerleri bulunur. PSNR değerleri yüksek olan görüntülerin kaliteside yüksek olarak elde edilir.

Seçilen görüntüler Microsoft Access veri tabanında kaydedilmektedir. Veri tabanında görüntü bilgileri, paletler (uygulamada kullanılan tüm renk paletleri) ve öğrenci bilgileri tutulmaktadır. Veri tabanında tutulan bu veriler üzerinden öğrencilerin törenlerde kullanacakları görev kartları basılmaktadır.

Tez çalışması kapsamında, verilen bir görüntüden törenlerde stadyumda değişik renklerden oluşan levhalarla gösterimi için uygun görüntünün belirlenmesi çözümü geliştirilmiştir. Bu uygulamada yüksek çözünürlüklü görüntüler işleme alınıp, renk azaltmaları teknikleri

kullanılarak daha alçak çözünürlüklü görüntüler elde edilmiştir. Uygulama çalışır durumdadır ve ileride oluşabilecek ihtiyaçlara göre kolayca genişletilebilecek bir yapıdadır.

8 KAYNAKLAR DİZİNİ

Adobe PhotoShop 7 Documentation, Adobe Systems Incorporated, 2002

Algan S., 2007, Her Yönüyle C#, Pusula Yayıncılık

Angel E., 2006, Interactive Computer Graphics, Addison Wesley

Hearn D., Baker M.P., 1997, Computer Graphics C Version, Pearson Prentice-Hall, Pearson Education, Inc. Upper Saddle River, New Jersey 07458.

Matlab6.5 Image Processing Toolbox Documentation, The MathWorks Inc., 2002

Overall J.E., Klett C.J., 1972, Applied Multivariate Analysis, McGraw-Hill, Inc.

Özdamar K., 2002, Paket Programlar ile İstatistiksel Veri Analizi (Çok Değişkenli Analizler), ISBN 975-6787-00-7, 975-6787-02-3

<http://ieeexplore.ieee.org>

<http://images.google.com.tr/>

<http://en.wikipedia.org/wiki/Principalcomponentanalysis>

<http://www.mathworks.com/access/helpdesk/help/toolbox/mbc/index.htm>

http://www.msakademik.net//makaleler_detay.aspx?id=46

http://www.msakademik.net//makaleler_detay.aspx?id=48

http://www.msakademik.net//makaleler_detay.aspx?id=51

<http://www.tablocu.com>

<http://tr.wikipedia.org/wiki/RGB>, 2007

EKLER

Ek 1 Uygulamanın Kullanımı

Ek 2 Programın Çıktıları

Ek 3 Türkçe-İngilizce Terimler Sözlüğü

Ek1. Uygulamanın Kullanımı

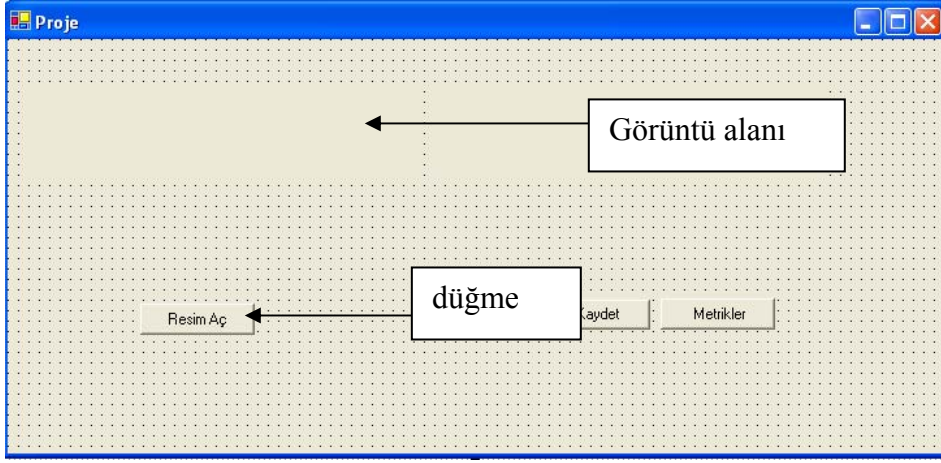
Tez kapsamında geliştirilen uygulama Visual Studio .NET 2003 C# da formlar kullanılarak kullanıcı arayüzü oluşturulmuştur. Uygulama kurulum dosyası olarak hazırlanıp kullanıcının bilgisayarına kurulabilmektedir. Uygulama kurulumunda sadece çalıştırılabilir dosyalar ve uygulama için gerekli olan dosyalar bulunur ve diskte fazla yer kaplamaz. Uygulamanın kullanıcı ara yüzü şekil ek1.1 de verilmiştir. Kullanıcı bu ara yüz sayesinde resim seçebilir ve seçtiği görüntü yukarıda bölümlerde anlatılan adımlardan geçirilir ve sonuçta stadyumda gösterilecek olan en uygun görüntü elde edilir.



Şekil ek1.1: Uygulama Ekranı

Bu ara yüz .NET in sunmuş olduğu form özellikleri ile hazırlanmıştır. Bu ortam, program geliştiricilerin, kullanıcı ara yüzü oluşturmalarında kolaylık sağlar, form üzerine resimler için picturebox

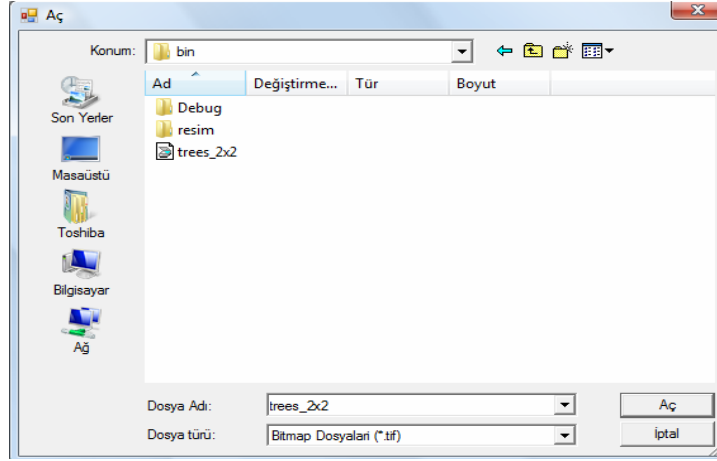
lar yerleştirilir ve program içinden bu picturebox'lara ilgili dosyalar aktarılır. Hangi dosyanın seçileceğine kullanıcı karar verir. Bu etkileşimi sağlamak için form üzerinde düğmeler (button) vardır. Şekil ek1.2'de form şekli verilmiştir.



Şekil ek1.2: .NET'de tasarlanan uygulama ekranı

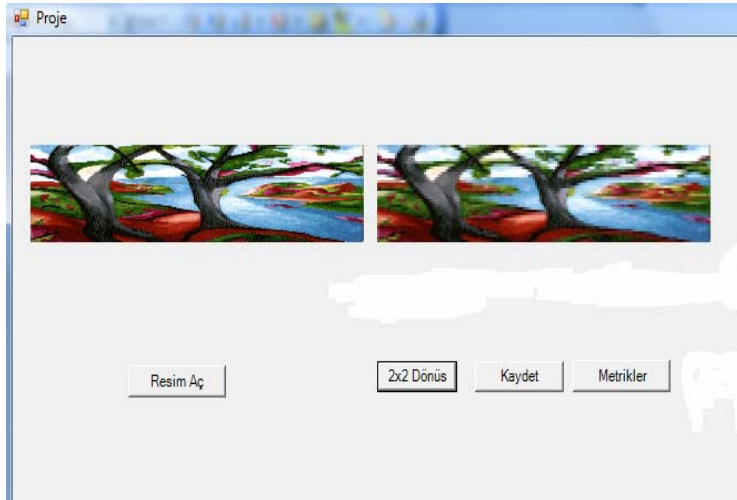
.Net'de kod dosyaları “.csproj” uzantılı olarak kaydedilirken kullanıcı ara yüzü dosyaları “.cs” uzantılı olarak kaydedilmektedir. Bu dosyalar sadece programın hazırlandığı makinada vardır, kullanıcı makinasında yer almaz. Çünkü proje dosyasını kurulum projesi ile birleştirip kurulum dosyası oluşturulur. Kullanıcı sadece kurulum dosyasını çalıştırıp projeyi makinasına yükler.

Kullanıcı “Resim Aç” düğmesini tıklayarak dosya seçme penceresini karşısına getirir, bu pencerede “.tif” uzantılı dosyaları seçebilir. Ara yüzün “Resim Aç” düğmesi tıklandığında karşımıza gelen pencere görüntüsü Şekil ek1.3’de verilmiştir.

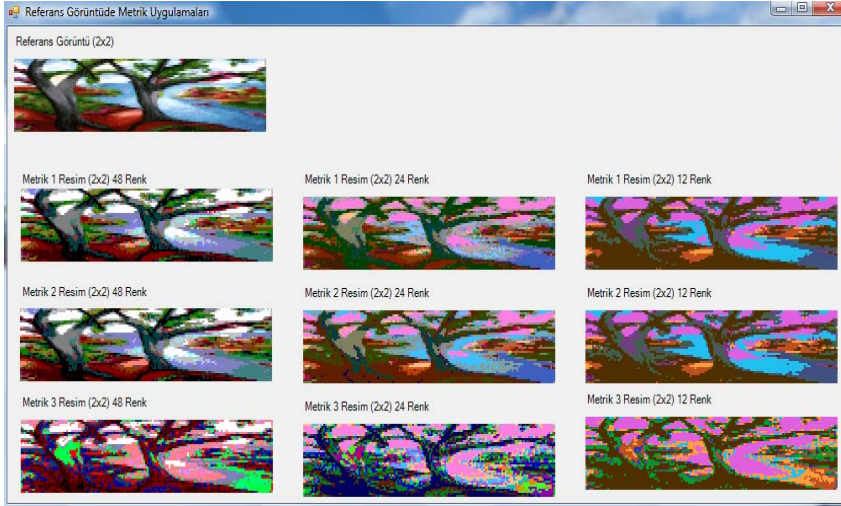


Şekil ek1.3: Dosya Seçme Ekranı

Şekil ek1.3'deki pencereden ilgili dosya seçildiğinde Şekil ek1.4'deki görüntü oluşur. Ve penceredeki 2x2'lik dönüştürme düğmesi tıklandığında Şekil ek1.4'deki pencere ekrana gelir. Bu penceredeki görüntülerden birincisi kullanıcının seçtiği resimdir ikinci ise 2x2'lik bloklara ayrılmış referans görüntüdür.



Şekil ek1.4: Dosya Seçme Ekranı



Şekil ek1.5: Metrik Uygulamaları Sonundaki Görüntüler

Bu aşamada kullanıcı ara yüzünde yer alan üzerinde “Metrikler” yazılı düğme tıklanırsa, yukarıda bölüm 5.1’de ayrıntıları ile açıklanmış olan “blokların eşleştirilmesinde kullanılan metrikler” gerçekleştirilip elde edilen görüntüler Şekil ek1.5 de verilmiştir. En üstteki resim referans görüntüsüdür, sıra ile 48, 24, 12’lik paletlere göre metrik görüntüleri görülmektedir.

Bu pencerede mevcut olan RMSE ve PSNR hesaplama düğmeleri, histogtam hesaplama düğmeleri tıklanarak bu penceredeki bütün görüntülerin histogramları ve RMSE ile PSNR’ları hesaplanıp diskte bir dosyaya yazdırılır. Bu pencerede toplam on adet görüntü olduğundan görüntülerin histogram dosyaları da on adettir.

Yukarıda bölüm 4’de anlatılan tablo 4.1’de yer alan algoritma referans görüntü üzerinde uygulanarak elde edilen histogram değerlerinin dosyadaki görüntüsünün bir bölümü tablo ek1.1’de verilmektedir.

Kırmızı (R)	Yeşil (G)	Mavi (B)
0	0	86
0	0	0
0	0	47
0	0	0
4	5	144
0	0	0
0	2	131
0	0	0
18	51	333
0	0	0
0	75	146
0	0	0
50	160	143
0	0	0

Tablo ek1.1 Histogram değerleri

Yukarıda bölüm 5’de verilen formül (5.7), (5.8), (5.9) un kullanılarak elde edilen RMSE değerlerinin kaydedilen dosyadaki görüntülerinden bir bölümü tablo ek1.2’de, PSNR değerleride tablo ek1.3 de verilmektedir.

RMSE_r	RMSE_g	RMSE_b	RMSE_{ort}
35	27	24	28
31	25	22	26
86	70	68	75
33	46	30	36

Tablo ek1.2 RMSE değerleri

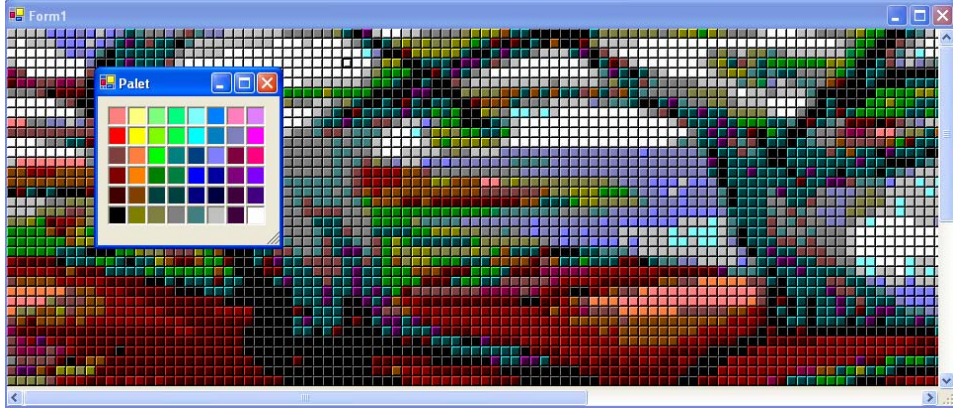
Metrik	PSNR _r	PSNR _g	PSNR _b	PSNR _{ort}
48/1	16,902	19,085	20,000	18,662
48/2	18,062	20,000	20,828	19,630
48/3	6,021	9,542	9,542	8,368
24/1	16,902	13,979	18,062	16,314

Tablo ek1.3 PSNR değerleri

Bu işlemlerden sonra en uygun görüntü seçilir ve bu görüntüde bazı piksellerdeki renkler kullanıcıya uygun gelmeyebilir veya değiştirmek isteyebilir, bu durumda görüntü tıklanır ve her piksele bir kare denk gelecek şekilde yeni bir pencerede gösterilir. Bu penceredeki her kare tıklanabilir özelliktedir ve tıklandığında ilgili palet ekrana gelerek istenilen renk değişikliği yapılabilir ve yapılan bu değişiklikler “Kaydet” düğmesi tıklanarak kaydedilir. Şekil ek1.6’da düğmelerden oluşan görüntü ve Şekil ek1.7’de palet görüntüsü verilmektedir.



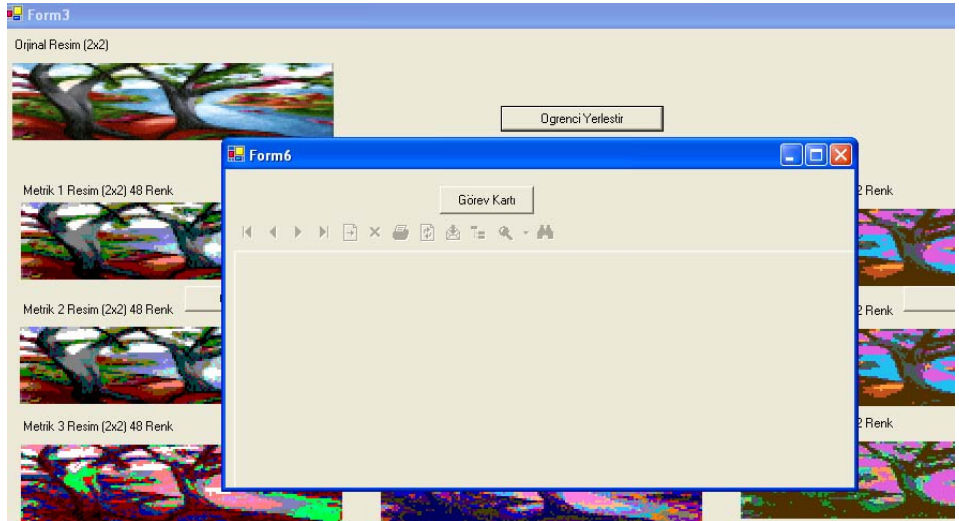
Şekil ek1.6:Renk Değiştirme ekranı



Şekil ek1.7:Renk Değişirme ekranı ve Palet

Görüntü seçim işlemi bittikten sonra öğrencilerin hangi renkteki levhayı kaldıracağını gösteren tablo bir dosyanın içine yazdırılır.

Oluşturulan tüm görüntüler veri tabanına kaydedilebilir. Kaydedilen verilerden öğrencilere resmi törenlerde kullanacakları görev kartları basılabilir.



Şekil ek1.8:Gorev kartı basımı

Öğrenci Görev Kartı 02.10.2007

Ad - Soyad	Deneme1	deneme1
Görüntü 1:	Renk_6_6	Görüntü 4: Renk_6_8
Görüntü 2:	Renk_6_6	Görüntü 5: Renk_6_8
Görüntü 3:	Renk_6_4	Görüntü 6: Renk_6_8
		Görüntü 7: Renk_6_8
		Görüntü 8: Renk_6_8
		Görüntü 9: Renk_6_8

Şekil ek1.9:Gorev kartı basımı

Şekil ek1.8’de öğrenci yerleştir düğmesi tıklanarak elde edilen görüntü verilmiştir. Bu ekrandaki görev kartı düğmesi tıklanarak, öğrenci bazında görev kartları görüntüsü Şekil ek1.9’da verilmiştir.

Öğrenci görev kartı basımı için öğrenci_master, görüntü_master ve paletler tabloları kullanılmış ve bunların görüntüsü Şekil ek1.10, ek1.11, ek1.12’de verilmiştir. Veri tabanı olarak Access veri tabanı seçilmiştir.

Oğrenci_Master

Oğrenci_No	Oğrenci_Adi	Oğrenci_SoyAdi	Okul_Adi
1	Deneme1	deneme1	dene1
2	Deneme2	deneme2	dene2

Şekil ek1.10:Öğrenci bilgileri tablosu

Goruntu_Master

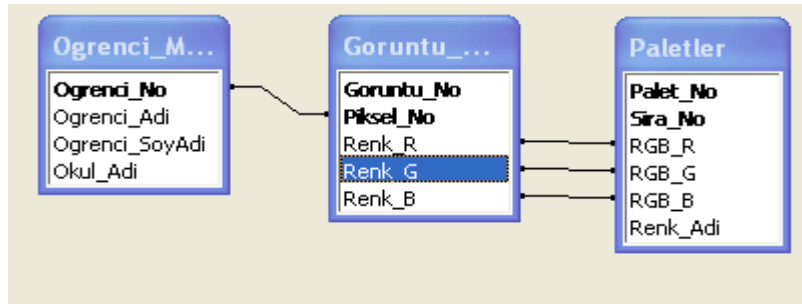
<i>Goruntu_No</i>	<i>Piksel_No</i>	<i>Renk_R</i>	<i>Renk_G</i>	<i>Renk_B</i>
1	0	192	192	192
1	1	192	192	192
1	2	255	255	255
1	3	192	192	192
1	4	128	0	64
1	5	64	0	0

Şekil ek1.11:Görüntü bilgileri tablosu

Paletler

<i>Palet No</i>	<i>Sıra No</i>	<i>RGB_R</i>	<i>RGB_G</i>	<i>RGB_B</i>	<i>Renk_Adi</i>
48	1	255	128	128	Renk_1_1
48	2	255	255	128	Renk_1_2
48	3	128	255	128	Renk_1_3
48	4	0	255	128	Renk_1_4
48	5	128	255	255	Renk_1_5

Şekil ek1.12:Palet bilgileri tablosu



Şekil ek1.13:Tablolar

Şekil ek1.13’de öğrenci görev kartı basımında kullanılan tabloların birbirleri arasındaki ilişkiler (relationship) verilmiştir.

Ek2. Program Çıktıları

Procedure 2x2

/* 2x2 lik bloklara ayrılarak renk azaltması işlemi.

Char mesaj

byte dizi1[300,72]

byte dizi1[300,72]

byte dizi1[300,72]

byte renkler1[5400]

byte renkler2[5400]

byte renkler3[5400]

integer toplam_kirmizi, toplam_yesil, toplam_mavi

integer ortalama_kirmizi, ortalama_yesil, ortalama_mavi

if resmi_sec = null then

 mesaj ← 'Lutfen once resim seciniz.'

 Print(mesaj)

else

 Color Renk

 Bitmap bmp ← Bitmap(pictureBox.Image)

 for i=0 to bmp.Heght-1, j←i+2 do

 for j=0 to bmp.Width-1, j←j+2 do

 toplam_kirmizi←0

 toplam_yesil←0; toplam_mavi←0

 ortalama_kirmizi←0; ortalama_yesil←0

 ortalama_mavi←0

```

for ii=i , ii1=0 to ii1<2 do
    for ii=i , ii1=0 to ii1<2 do
        Renk ← bmp.GetPixel(ii,jj)
        Dizi1[ii,jj] ← Renk.R
        Dizi1[ii,jj] ← Renk.G
        Dizi1[ii,jj] ← Renk.B
    /* Renklerin toplamı hesaplanır.

Toplam_kirmizi=toplam_kirmizi+dizi1[ii,jj]
Toplam_yesil=toplam_yesil+dizi1[ii,jj]
Toplam_mavi=toplam_mavi+dizi1[ii,jj]
    Repeat
Repeat
/* 2x2 lik blogun ortalaması alınır.
Ortalama_kirmizi ← toplam_kirmizi/4
Ortalama_yesil ← toplam_yesil/4
Ortalama_mavi ← toplam_mavi/4
Renkler1[nn1] ← ortlama_kirmizi
Renkler2[nn1] ← ortlama_yesil
Renkler3[nn1] ← ortlama_mavi
nn1 ← nn1+1
for kk=i , kk2=0 to kk2<2 do
    for kk1=j , kk3=0 to kk3<2 do
        renk← Color.FromArgb(ortalama_kirmizi,
                                ortalama_yesil,
                                ortalama_mavi)
/* Elde edilen yeni renk ilgili piksellere yerleştirilir.

```

```

        bmp.SetPixel(kk, kk1, renk)
        repeat
    repeat
        picturebox.image ← bmp
    repeat
repeat
end 2x2

```

/* Metriklerin Hesaplanması

Procedure Palet_Secimi(renk_adet, satir, sutun)

```

    Renkler_Dizini ← picturebox[renk_adet]

```

```

    For rows=0 to rows<satir do

```

```

        For cols = 0 to cols < sutun do

```

```

            i ← (rows * sutun) + cols

```

```

            Renkler_Dizini[i] ← PictureBox

```

```

            Read(paletler)

```

```

            Renkler_Dizini[i].BackColor ← (Read, Green, Blue)

```

```

        repeat

```

```

    repeat

```

End Palet_Secimi

Procedure Metrikler

```

    Palet_Secimi(48, 6, 8)

```

Palet_Secimi(24,6,4)

Palet_Secimi(12,3,4)

integer S[48]; S_24[24]; S_12[12]

integer toplam_RMSE_48_R[3]; toplam_RMSE_48_G[3];

toplam_RMSE_48_B[3];

integer toplam_RMSE_24_R[3]; toplam_RMSE_24_G[3];

toplam_RMSE_24_B[3];

integer toplam_RMSE_12_R[3]; toplam_RMSE_12_G[3];

toplam_RMSE_12_B[3];

integer toplam_piksel $\leftarrow$ 5400

for i=0 to 3 do

toplam_RMSE_48_R[i] $\leftarrow$ 0

toplam_RMSE_48_G[i] $\leftarrow$ 0

toplam_RMSE_48_B[i] $\leftarrow$ 0

toplam_RMSE_24_R[i] $\leftarrow$ 0

toplam_RMSE_24_G[i] $\leftarrow$ 0

toplam_RMSE_24_B[i] $\leftarrow$ 0

toplam_RMSE_12_R[i] $\leftarrow$ 0

toplam_RMSE_12_G[i] $\leftarrow$ 0

toplam_RMSE_12_B[i] $\leftarrow$ 0

repeat

for i = 0 to 5400 do

/* Metrik 1

For j = 0 to 48 do

$$S[j] \leftarrow (\text{renkler1}[i] - \text{Renkler_Dizini}[j].R) * \\ (\text{renkler1}[i] - \text{Renkler_Dizini}[j].R) + \\ (\text{renkler2}[i] - \text{Renkler_Dizini}[j].G) * \\ (\text{renkler1}[i] - \text{Renkler_Dizini}[j].G) + \\ (\text{renkler3}[i] - \text{Renkler_Dizini}[j].B) * \\ (\text{renkler1}[i] - \text{Renkler_Dizini}[j].B)$$

repeat

for j = 0 to 48 do

if S[j] <= küçük then

kucuk $\leftarrow$ S[j]

kuck_index $\leftarrow$ j

endif

repeat

renklerS11[i] = Renkler_Dizini[kucuk_index].R

renklerS12[i] = Renkler_Dizini[kucuk_index].G

renklerS13[i] = Renkler_Dizini[kucuk_index].B

k48 $\leftarrow$ 0

toplaml_RMSE_48_R[k48] $\leftarrow$ toplaml_RMSE_48_R[k48] +
 (renkler1[i] - Renkler_Dizini[kucuk_index].R)
 * (renkler1[i] - Renkler_Dizini[kucuk_index].R)

toplaml_RMSE_48_G[k48] $\leftarrow$ toplaml_RMSE_48_G[k48] +
 (renkler2[i] - Renkler_Dizini[kucuk_index].G)
 * (renkler2[i] - Renkler_Dizini[kucuk_index].G)

toplaml_RMSE_48_B[k48] $\leftarrow$ toplaml_RMSE_48_B[k48] +
 (renkler3[i] - Renkler_Dizini[kucuk_index].B)

```

* (renkler3[i] –Renkler_Dizini[kucuk_index j].B)

/* Metrik 2
for i=0 to 48 do
    S[i] ← 0
Repeat

For j = 0 to 48 do
    S[j] ← Abs(renkler1[i] –Renkler_Dizini[j].R) +
            (renkler2[i] –Renkler_Dizini[j].G)+
            (renkler3[i] –Renkler_Dizini[j].B)
repeat
for j = 0 to 48 do
    if S[j] <= kucuk then
        kucuk ← S[j]
        kuck_index ← j
    endif

repeat

renklerS21[i] = Renkler_Dizini[kucuk_index].R
renklerS22[i] = Renkler_Dizini[kucuk_index].G
renklerS23[i] = Renkler_Dizini[kucuk_index].B

k48 ← k48+1
toplam_RMSE_48_R[k48] ← toplam_RMSE_48_R[k48] +
                        (renkler1[i] –Renkler_Dizini[kucuk_index].R)

```

```

* (renkler1[i] – Renkler_Dizini[kucuk_index j].R)
toplaml_RMSE_48_G[k48] ← toplaml_RMSE_48_G[k48] +
    (renkler2[i] – Renkler_Dizini[kucuk_index].G)
    * (renkler2[i] – Renkler_Dizini[kucuk_index j].G)
toplaml_RMSE_48_B[k48] ← toplaml_RMSE_48_B[k48] +
    (renkler3[i] – Renkler_Dizini[kucuk_index].B)
    * (renkler3[i] – Renkler_Dizini[kucuk_index j].B)

/* Metrik 3
for i=0 to 48 do
S[i] ← 0
repeat
for j = 0 to 48 do
    S[j] ← Abs((renkler1[i] + renkler2[i] + renkler3[i])/3 –
        (Renkler_Dizini[j].R)+Renkler_Dizini[j].G
        +Renkler_Dizini[j].B) / 3)
repeat
for j = 0 to 48 do
    if S[j] <= kucuk then
        kucuk ← S[j]
        kuck_index ← j
    endif
repeat

renklerS31[i] = Renkler_Dizini[kucuk_index].R
renklerS32[i] = Renkler_Dizini[kucuk_index].G
renklerS33[i] = Renkler_Dizini[kucuk_index].B

```

```

k48 ← k48+1
toplaml_RMSE_48_R[k48] ← toplaml_RMSE_48_R[k48] +
    (renkler1[i] –Renkler_Dizini[kucuk_index].R)
    * (renkler1[i] –Renkler_Dizini[kucuk_index j].R)
toplaml_RMSE_48_G[k48] ← toplaml_RMSE_48_G[k48] +
    (renkler2[i] –Renkler_Dizini[kucuk_index].G)
    * (renkler2[i] –Renkler_Dizini[kucuk_index j].G)
toplaml_RMSE_48_B[k48] ← toplaml_RMSE_48_B[k48] +
    (renkler3[i] –Renkler_Dizini[kucuk_index].B)
    * (renkler3[i] –Renkler_Dizini[kucuk_index j].B)

```

/* 24 ve 12 lik paletlerde aynı algoritma ile işlenir.

Repeat

/* RMSE hesaplaması

for i = 0 to 3 do

RMSE_R48[i] ← Sqrt(toplaml_RMSE_48_R[i] / toplaml_piksel

RMSE_G48[i] ← Sqrt(toplaml_RMSE_48_G[i] / toplaml_piksel

RMSE_B48[i] ← Sqrt(toplaml_RMSE_48_B[i] / toplaml_piksel

repeat

End Metrikler

/* Görüntüye göre palet oluşturma

Procedure Renk_32

```

integer toplam_piksel  $\leftarrow$  5400
integer toplam_kac_renk[toplam_piksel]

for i=0 to i<nn-1 do
  for j = i+1 to j<nn do
    if renkler1[i] = renkler1[j] then
      if renkler2[i] = renkler2[j] then
        if renkler3[i] = renkler3[j] then
          toplam_kac_renk[i]  $\leftarrow$ 
            toplam_kac_renk[i] +1
        endif
      endif
    endif
  repeat
repeat

```

/* Büyükten küçüğe doğru sıralama

```

integer sakla_toplam_renk
integer sakla_indexler
integer sakla_index[toplam_piksel]
for i=0 to toplam_piksel do
  sakla_index[i]  $\leftarrow$  i
repeat

```

```

for i=0 to i<toplaml_piksel-1 do
  for j = i+1 to j < toplaml_piksel do
    if (toplaml_kac_renk[i] < toplaml_kac_renk[j]) then
      sakla_toplam_renk ← toplaml_kac_renk[i]
      sakla_indexler ← sakla_index[i]
      toplaml_kac_renk[i] ← toplaml_kac_renk[j]
      sakla_index[i] ← sakla_index[j]
      sakla_index[j] ← sakla_indexler
      toplaml_kac_renk[j] ← sakla_toplam_renk
    endif
  repeat
repeat

```

/* Aynı olanların listeden silinmesi

```

For i = 0 to i < toplaml_piksel-1 do
  For j = i+1 to j < toplaml_piksel do
    if (renkler1[sakla_index[i]] = renkler1[sakla_index[j]]) then
      if (renkler2[sakla_index[i]] = renkler2[sakla_index[j]])
      then
        if (renkler3[sakla_index[i]] =
          renkler3[sakla_index[j]]) then
          Clear(toplam_kac_renk,j,1)
          Clear(sakla_index,j,1)
        endif
      endif
    endif
  endif
endif

```

```

        repeat
repeat
/* 32 renk bulma
integer n1 = 0
for i = 0 to i < toplam_piksel-1 do
    If (toplam_kac_renk[i] not = 0 ) & n1 < 32 then
        Son_index[n1] ← sakla_index[i]
        Son_toplam_kac_renk[n1] ← toplam_kac_renk[i]
        n1 ← n1 + 1
    endif
repeat

Palet_32
/* elde edilen renkler renkler dizinine palet 48 algoritması ile aktarılır.
end Renk_32

/* PSNR değerlerinin hesaplanması
Procedure PSNR_RMSE
/* RMSE ler metrikler bölümünde hesaplanmıştır.
integer PSNR_R_48[3]
for i=0 to 3 do
    PSNR_R_48[i]←20 * Log10(255/toplam_RMSE_R_48[i]
    PSNR_G_48[i]←20 * Log10(255/toplam_RMSE_G_48[i]
    PSNR_B_48[i]← 20 * Log10(255/toplam_RMSE_B_48[i]
/* Dosyaya yazdırılır
Yazdır(PSNR_R_48[i], PSNR_G_48[i], PSNR_B_48[i],
        toplam_RMSE_R_48[i],toplam_RMSE_G_48[i],
        toplam_RMSE_B_48[i])

```

```

repeat
    /* Yukarıdaki işlemler palet 24 ve palet 12 içinde uygulanır.
End PSNR_RMSE

/* Görüntülerin table kaydedilmesi.
Procedure Resimleri_Kaydet(integer palet, integer R, integer G, integer B)
    Goruntu_No_Hesap() ;
    for i=0 to toplam_piksel do
        write(goruntu_no,i,R,G,B)
    repeat
end Resimler_Kaydet
Procedure Goruntu_No_Hesap
    Read(Goruntu_Master_count)
    Goruntu_no ← Goruntu_Master_count / 5400 + 1
end Goruntu_No_Hesap
Procedure Kullanılacak_Goruntu_Kaydet
    Resimleri_Kaydet(48,renklerS111,renklerS112,renklerS113);
end Kullanılacak_Goruntu_Kaydet

/* Öğrencilere verilecek görev listesi
Procedure Ogrenci_Gorev_Listesi
    /* Bu tez uygulamasında Crystal Reports kullanılarak öğrenci görev
        kartları basılabilir.

    /* Bu raporda, bir ana rapor (main_report) ve toplam fon adeti
        kadar alt rapor (sub_report) kullanılmaktadır. Raporda Crystal
        Reports özellikleri ile tablolar ilişkilendirilerek listeler
        oluşturulabilir.
end Ogrenci_Gorev_Listesi

```

/* Görüntülerin renk piksel bazında renk değiştirme işlemi

Procedure Butonlu_Resim_Goster(pictureBox resim_adi, integer S1,
integer S2, integer S3, integer palet_secimi)

integer satir, sutun, n1

satir $\leftarrow$ 0

sutun $\leftarrow$ 0

n1 $\leftarrow$ 0

buton buton1

for i = 0 to i < toplam_piksel do

if (i % 150 = 0) then

n1 $\leftarrow$ satir

satir $\leftarrow$ satir + 1

sutun $\leftarrow$ 0

endif

buton1.Click $\leftarrow$ (buton_sec)

buton1.BackColor = Color.FromArgb(S1,S2,S3)

n1 $\leftarrow$ n1 + 36

sutun $\leftarrow$ sutun + 1

repeat

end Butonlu_Resim_Goster

Procedure buton_sec

Palet_Secimi(48,6,8)

end buton_sec

Ek3. Türkçe-İngilizce Terimler Sözlüğü

Türkçe	İngilizce
Animasyon	:Animation
Ana Bileşenler Analizi	:Principal Components Analysis(PCA)
Bellek	:Memory
Bilgi	:Information
Düğme	:Button
En büyük işaret gürültü oranı	Peak signal to noise ratio (PSNR)
En küçük kareler ortalamasının karekökü	:Root minumum Square Error (RMSE)
Fare	:mouse
Gerçek Renk	:True color
Giriş Birimleri	:Input devices
Görüntü	:Image
Görüntü işleme	:Image processing
Grafik sistemler	:Graphics systems
Gri seviye	:Gray-Scale
İkili görüntü	:binary images
İlişkiler	:Relationship
İşlemci	:Processor
İşletim sistemi	:Operating system
İz maskesi	:Shadow mask
Kullanıcı arayüzü	:User interface
Oyun çubuğu	:Joystic

Özdeğerler	:Eigenvalues
Özvektörler	:Eigenvectors
Piksel	:Pixel
Piksel bellek	:Frame buffer
Renk azaltma	:Color reduction
Renk sistemi	:Full Colour
Sınıf	:Class
Simülasyon	:Simulation
Tasarım	:Dizayn
Yazılım	:Software

ÖZGEÇMİŞ

Nil KARACA

adres: Güzelyalı Mahallesi 22 Sokak No: 2/5 Konak/İZMİR 35280

tel: (0232) 2463971

e-mail: nil.karaca@ege.edu.tr

Kişisel Bilgiler	Milliyeti	: T.C.
	Doğum Yeri / Tarihi	: İzmir / 29.03.1968
Eğitim	2005 -	Ege Üniversitesi Uluslararası Bilgisayar Enstitüsü (UBE) <i>Bilgi Teknolojileri Ana Bilim Dalı</i>
	1985- 1989	Ege Üniversitesi Fen Fakültesi Matematik Bölümü
	1982-1985	İzmir Karataş Lisesi
İş Deneyim	1989-	Ege Üniversitesi Analist Programcı (Uzman)