

**BÖLGESEL RENK KONTRASTI VE RENK DAĞILIMI BİLGİSİ
KULLANARAK KARMAŞIK AĞ DESTEKLİ BELİRGİN ALAN TESPİTİ**

ALPER AKSAÇ

**YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ**

**TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**TEMMUZ 2013
ANKARA**

Fen Bilimleri Enstitü onayı

Prof. Dr. Ünver Kaynak
Müdür

Bu tezin Yüksek Lisans derecesinin tüm gereksinimlerini sağladığını onaylarım.

Doç. Dr. Erdoğan Doğdu
Anabilim Dalı Başkanı

Alper AKSAÇ tarafından hazırlanan BÖLGESEL RENK KONTRASTI VE RENK DAĞILIMI BİLGİSİ KULLANARAK KARMAŞIK AĞ DESTEKLİ BELİRGİN ALAN TESPİTİ adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Yrd. Doç. Dr. Tansel ÖZYER
Tez Danışmanı

Tez Jüri Üyeleri

Başkan : Yrd. Doç. Dr. Esra Kadioğlu ÜRTİŞ _____

Üye : Doç. Dr. Bülent TAVLI _____

Üye : Yrd. Doç. Dr. Tansel ÖZYER _____

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Alper AKSAÇ

Üniversitesi : TOBB Ekonomi ve Teknoloji Üniversitesi
Enstitüsü : Fen Bilimleri
Anabilim Dalı : Bilgisayar Mühendisliği
Tez Danışmanı : Yrd. Doç. Dr. Tansel ÖZYER
Tez Türü ve Tarihi : Yüksek Lisans – Temmuz 2013

ALPER AKSAÇ

**BÖLGESEL RENK KONTRASTI VE RENK DAĞILIMI BİLGİSİ
KULLANARAK KARMAŞIK AĞ DESTEKLİ BELİRGİN ALAN TESPİTİ**

ÖZET

Bilgisayarla görü alanında dikkati çeken alanların bulunması işlemi gittikçe önemi artan popüler bir konudur. Son yıllarda, bu konu üzerine birçok araştırma çalışması yapılmış ve çeşitli yöntemler geliştirilmiştir. Tespit edilen alanlar belirgin nesne tespiti, resimden çıkarım yapma, resmin yeniden boyutlandırılması, resim ve video sıkıştırma gibi birçok bilgisayarla görü uygulamasında girdi olarak önemli rol oynamaktadır.

Önerilen modellerde, insan görselliğinde dikkati çeken piksellerin/bölgelerin belirginleştirilmesi ve arka planda kalan piksellerin/bölgelerin bastırılması ile dikkat çekme haritaları üretilir. Yapılan önceki çalışmalarda dikkat çekme haritası oluşturmak için genel olarak resmin renk, biçim, doku, parlaklık ve konum bilgileri kullanılır.

Bu tezde ise, süperpiksel yöntemiyle bölütleme yapılan resimde karmaşık ağ, histogram, renk, konum ve alan bilgisi kullanılarak dikkat çekme haritası oluşturulması üzerine çalışılmıştır. Önerilen yöntem önceki yöntemlerin halka açık veri kümesi sonuçları ile karşılaştırılarak, başarısı ve tutarlılığı ortaya konulmuştur.

Anahtar Kelimeler: resim bölütleme, süperpiksel, belirgin alan tespiti, dikkat çekme haritası, karmaşık ağ

University : **TOBB Economics and Technology University**
Institute : **Institute of Natural and Applied Sciences**
Science Programme : **Computer Engineering**
Supervisor : **Assistant Prof. Dr. Tansel ÖZYER**
Degree Awarded and Date : **Master of Science – July 2013**

ALPER AKSAÇ

**COMPLEX NETWORK BASED SALIENT REGION DETECTION USING
REGIONAL COLOR CONTRAST AND COLOR DISTRIBUTION**

ABSTRACT

The process of salient regions detection in the field of computer vision has increasingly become very popular topic. In recent years, lots of research works have been carried out and lots of various techniques have been developed on this topic. Detected regions plays an important role in many computer vision applications such as salient object detection, image retrieval, image retargeting - seam carving, image and video compression as an input.

In the proposed frameworks, saliency maps are generated by highlighting foreground pixels/regions attracting human visual attention and suppressing background pixels/regions. In previous studies, to generate saliency maps are generally used to image's color, orientation, texture, luminance and location information.

In this thesis, it has been studied on generating saliency maps by using complex network, histogram, color, location and area information in the segmented image with superpixel method. The proposed system's success and consistency are illustrated by comparing with the results of previous methods which have been gathered with a publicly available dataset.

Keywords: image segmentation, superpixel, salient region detection, saliency map, complex network

TEŐEKKÜR

Bu tezin hazırlanmasında yardım ve katkılarıyla beni yönlendiren deęerli Hocam Yrd. Doç. Dr. Tansel Özyer'e, yüksek lisans eğitimim boyunca bana deęerli katkılarda bulunan TOBB Ekonomi ve Teknoloji Üniversitesi Bilgisayar Mühendislięi bölümü Hocalarıma, tez çalışmalarım boyu bana büyük yardımları olan deęerli arkadaşım Orkun Öztürk'e, yüksek lisans çalışmalarım boyunca bana maddi konularda destek verip verimli bir şekilde çalışmama katkıda bulunan TÜBİTAK'a ve en önemlisi beni bugünlere getiren deęerli aileme teşekkürü borç bilirim.

İÇİNDEKİLER

TEZ BİLDİRİMİ.....	iii
ÖZET.....	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER	vii
ŞEKİLLERİN LİSTESİ	ix
TABLOLARIN LİSTESİ.....	x
ALGORİTMALARIN LİSTESİ	xi
KISALTMALAR	xii
SEMBOL LİSTESİ	xiii
1. GİRİŞ	1
2. İLGİLİ ÇALIŞMALAR	4
2.1. Belirginlik Tespiti.....	4
2.2. Süperpiksel Yöntemiyle Resim Bölütleme	7
2.2.1. Algoritma	9
2.3. Karmaşık Ağ.....	11
2.3.1. Topluluk Çıkarımı	12
2.3.2. Kalite Fonksiyonu	13
2.3.2.1. Normalize Kesim.....	13
2.3.2.2. Modülerlik	14
2.3.3. Topluluk Çıkarma Yöntemleri	14
2.3.3.1. Girvan ve Newman'ın Bölücü Algoritması.....	14
2.3.3.2. Hızlı Açıgözlü Algoritması	16

2.3.3.3. Etiket Yayılımı	17
2.4. Özellik Noktalarının Tespiti	18
2.4.1. Moravec Köşe Tespit Yöntemi	18
2.4.2. Harris Köşe Tespit Yöntemi.....	19
2.4.3. Shi-Tomasi Köşe Tespit Yöntemi.....	21
3. BÖLGESEL BİLGİ KULLANARAK KARMAŞIK AĞ DESTEKLİ BELİRGİN ALAN TESPİTİ.....	22
3.1. Genel Bakış	22
3.1.1. SLIC Resim Bölütleme	23
3.1.2. Renk Kontrastı	23
3.1.3. Karmaşık Ağ Kontrastı	24
3.1.4. Dağılım.....	25
3.2. Metodoloji	25
3.2.1. SLIC Resim Bölütleme	25
3.2.2. Renk Kontrastı	26
3.2.3. Karmaşık Ağ Kontrastı	29
3.2.4. Dağılım.....	33
3.2.5. Dikkat Çekme Haritası Oluşturma	34
4. DENEYLER.....	37
5. SONUÇ	43
KAYNAKLAR	44
ÖZGEÇMİŞ	50

ŞEKİLLERİN LİSTESİ

Şekil 1.1.	Genel Bakış.....	2
Şekil 2.1.	Popüler yöntemlerden dikkat çekme haritaları.....	5
Şekil 2.2.	SLIC algoritması sonuçları.....	7
Şekil 2.3.	SLIC algoritmasının 1 döngü sonrası.....	10
Şekil 2.4.	SLIC algoritmasının 10 döngü sonrası.....	10
Şekil 2.5.	Dendogram ile ağ gösterimi.....	17
Şekil 2.6.	Etiket yayılım yöntemi ile ağ etiketlenmesi.....	17
Şekil 2.7.	R fonksiyonu bölgeleri.....	21
Şekil 3.1.	Yöntemin akış şeması.....	22
Şekil 3.2.	Doğruluk haritalarının ortalaması.....	28
Şekil 3.3.	Merkez noktalarına göre dikkat çekme haritaları.....	29
Şekil 3.4.	Karmaşık ağ kontrastı yaklaşımı.....	30
Şekil 3.5.	Renk ve karmaşık ağ kontrastı birleşimi.....	33
Şekil 3.6.	Yöntemin aşamalarında başka sonuçlar.....	36
Şekil 4.1.	Doğruluk ve dikkat çekme haritalarının karşılaştırması.....	37
Şekil 4.2.	Süperpiksel sayısına göre yöntemin başarı oranı.....	38
Şekil 4.3.	Önerilen yöntemdeki adımların karşılaştırılması.....	39
Şekil 4.4.	Önerilen yöntemin önceki çalışmalarla karşılaştırılması.....	40
Şekil 4.5.	Önerilen yöntemin önceki çalışmalarla karşılaştırılması.....	40
Şekil 4.6.	Popüler çalışmalardan ve önerilen yöntemden çıktılar.....	42

TABLULARIN LİSTESİ

Tablo 4.1. Yöntemlerin çalışma süreleri.....	39
--	----

ALGORİTMALARIN LİSTESİ

Algoritma 2.1.	SLIC Sözde Kodu.....	11
Algoritma 3.1.	Karmaşık Ağ Kontrastı Sözde Kodu.....	32
Algoritma 3.2.	Önerilen Yöntemin Sözde Kodu.....	35

KISALTMALAR

Kısaltma	Açıklama
İGS	İnsan Görme Sistemi
DoG	Gauss Farkı
CIELab	CIE tarafından 1976 yılında tanımlanan renk uzayı formülü
SLIC	Basit Lineer Yinelemeli Öbekleme
CPU	Merkezi İşlem Birimi
GPU	Grafik İşlemci Ünitesi

SEMBOL LİSTESİ

Bu çalışmada kullanılmış olan simgeler açıklamaları ile birlikte aşağıda sunulmuştur.

Simge	Açıklama
N	Resimdeki piksel sayısı
k	Süperpiksel sayısı
Q	Modülerlik ölçütü
λ	Özdeğer vektörü
c	Bölgenin ortalama renk değeri
p	Bölgenin ortalama konum değeri
d_{ij}^{Bha}	i ve j bölgeleri arasındaki Bhattacharyya uzaklığı
d_i^{Cent}	i bölgesinin olası merkeze uzaklığı
Sal_i^{Cont}	i bölgesinin renk kontrastı değeri
Sal_i^{CNCont}	i bölgesinin karmaşık ağ kontrastı değeri
Sal_i^{Dist}	i bölgesinin dağılım kontrastı değeri
\overline{Sal}_i^{Cont}	i bölgesinin genel kontrast değeri
Sal_i	i bölgesinin belirginlik değeri
w_j	j bölgesinde bulunan toplam piksel sayısı
w_{ij}^C	i ve j bölgeleri arasındaki renk farkı ağırlığı
w_{ij}^P	i ve j bölgeleri arasındaki konum farkı ağırlığı
f_{ij}^{adj}	i ve j bölgeleri arasındaki komşuluk değeri
t_c	Renk fark eşiği
t_p	Konum fark eşiği
σ_c^2	Renk kısıtı
σ_p^2	Konum kısıtı
α	Renk kontrastı kısıtı
β	Karmaşık ağ kontrastı kısıtı
γ	Dağılım kontrastı kısıtı

1. GİRİŞ

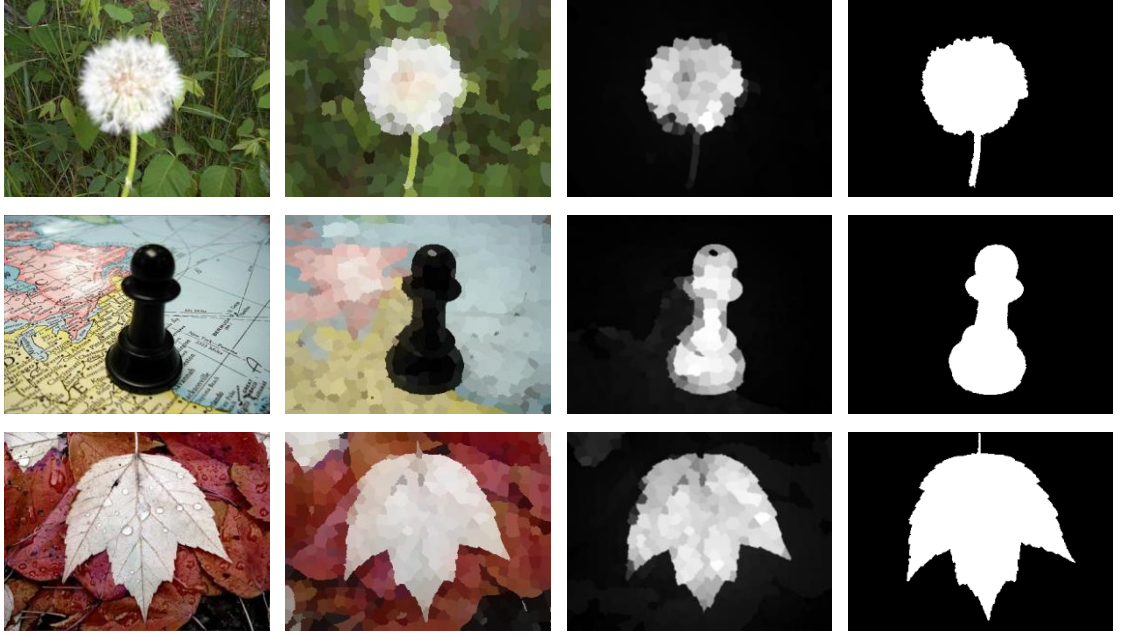
İnsan görme sistemi (İGS)(Human Visual System - HVS) [1] resimde bulunan belirgin bölgeleri diğer kısımlardan hızlı ve kolay bir şekilde ayırabilmektedir. Bu sistemi aynı şekilde otonom sistemlere uygulama, resimde bulunan nesnelere insan görme sistemindeki gibi analiz etme birden çok bilim dalını ilgilendiren bir problemdir. Gerçeğe yakın sonuçlar elde etmek için sinir bilimi [2,3], kavramsal psikoloji [4,5], bilgisayarla görü [6,7] gibi alanların bu probleme olan yaklaşımlarını ve çözümlerini dikkate almak gerekir.

Belirginlik görsel tekillik, ön görülemezlik, seyreklik gibi kavramların resimdeki renk, eğim, kenar ve sınır özniteliklerinde yer almasıyla meydana gelmektedir. İnsan görme sistemi resimdeki bazı alanlara daha çok yoğunlaşır ve dikkat eder. Resimdeki önemli alanların ve nesnelere konumlarını bulan belirginlik tespiti, bu insan görme sistemindeki seçme işlemi gibi çalışır. Dikkat çeken bölgeler tespit edildikten sonra, bilgisayarla görü alanındaki nesne tanıma [8], resim bölütleme [9-11], resimden çıkarım yapma [12,13], resim ve video sıkıştırma [14,15], resim boyutlandırma [16-20], gibi birçok içerik tabanlı uygulamada girdi olarak kullanılmaktadır.

Algısal araştırmalar [21-23] görsellikte belirginliği etkileyen en önemli etkeni kontrast olarak belirtmiştir. Yapılan önceki çalışmalarda kontrast tanımı, pikseller/bölgeler arasındaki renk değişimi, bütünlüğü ve dağılımı, kenar, eğim, uzamsal sıklık (spatial frequency), sıklık grafiği, çok boyutlu tanımlayıcılar veya bunların değişik kombinasyonları ile ifade edilmektedir.

İnsan görme sistemi dikkatini çeken nesneye odaklandığı zaman, belirgin nesneyi analiz etmeye çalışır, diğer kısımları inceleme safhasına katmaz. Bu görme sistemini temel olarak oluşturulan dikkat çekme bölgelerini tespit eden yöntemler iki kategoriye ayrılır [24]: yukarıdan aşağıya (pikseller/bölgeler arasındaki çevresel farka bakar) ve aşağıdan yukarı (pikseller/bölgeler arasındaki genel renk dağılımı ve tekilliğine bakar).

Bunun yanında karmaşık ağ konusundaki ilerlemeler bu konunun resim bölütleme algoritmalarına uygulanmasına olanak sağlamıştır. Yöntem, resimdeki bölütlenen alanların topluluk olarak ele alınarak kümelenmesi mantığına dayanır.



Şekil 1.1. Genel Bakış

Şekil 1.1’de probleme yönelik genel bakış sunulmuştur ve soldan sağa: orijinal resim, bölütleme sonucu resim, önerilen yöntemin sonucu dikkat çekme haritası ve doğruluk haritası.

Bu tez kapsamında belirgin alanların tespiti konusuna yapılan katkılar aşağıdaki gibi sıralanabilir;

- Kullanıcıdan hiçbir girdi almayan, interaktif olmayan bütünüyle otomatik çalışan resimdeki belirgin alanları tespit eden aşağıdan yukarı kategorisinde metot önerilmiştir.
- Kullanılan veri kümesinden hiçbir ön bilgi almayan, öğrenme ve eğitim verisi olmadan çalışan bir metot geliştirilmiştir.

- Resim bölütlemesi yapılarak çalışma zamanı düşürülmüş, yüksek çözünürlüklü resimlerde de verimli ve etkili çalışan algoritma oluşturulmuştur.
- Kontrast bilgisi bölgelerin birbirleriyle olan tekillik ilişkisi ve mekânsal dağılımı üzerinden elde edilmiştir.
- Önerilen metot resmin merkezinde bulunmayan belirgin nesnelere de çalışabilecek şekilde geliştirilmiştir.
- Karmaşık ağ teorisi resim bölütleme kullanılarak, belirgin alanların tespiti konusuna uygulanmıştır. Çizge tabanlı bölütleme algoritmalarından farklı bir yaklaşım izlenmiştir.
- Herkes tarafından kullanılan popüler veri kümesi ile test edilerek yöntemin başarısı öncekilerle karşılaştırılmıştır.

Bu tez çalışması şu şekilde düzenlenmiştir: Bölüm 1’de, tez çalışmasının temelleri ve yapılan çalışmayı kısaca özetleyen giriş bölümü bulunmaktadır. Bölüm 2’de, kullanılan yöntemler hakkında bilgi verilerek, yapılan çalışmalara değinilmiştir. Bölüm 3’te bu çalışmada geliştirilen belirgin alanların tespiti yöntemi anlatılmıştır. Bölüm 4’te yapılan deneylerden bahsedilirken, önceki yöntemlerle karşılaştırılmasına bakılmıştır, çıkan tüm bu sonuçlar yorumlanmıştır. Sonuncu Bölüm 5’de ise bu tez çalışmasında yapılan çalışma ve elde edilen sonuçlar özetlenip, gelecek çalışmaların ne doğrultuda olacağı belirtilerek tez sonlandırılmıştır.

2. İLGİLİ ÇALIŞMALAR

Bu bölümde tez çalışmasında geliştirilen yöntemin temellerinde kullanılan kavramlar ile ilgili bilgiler verilecektir.

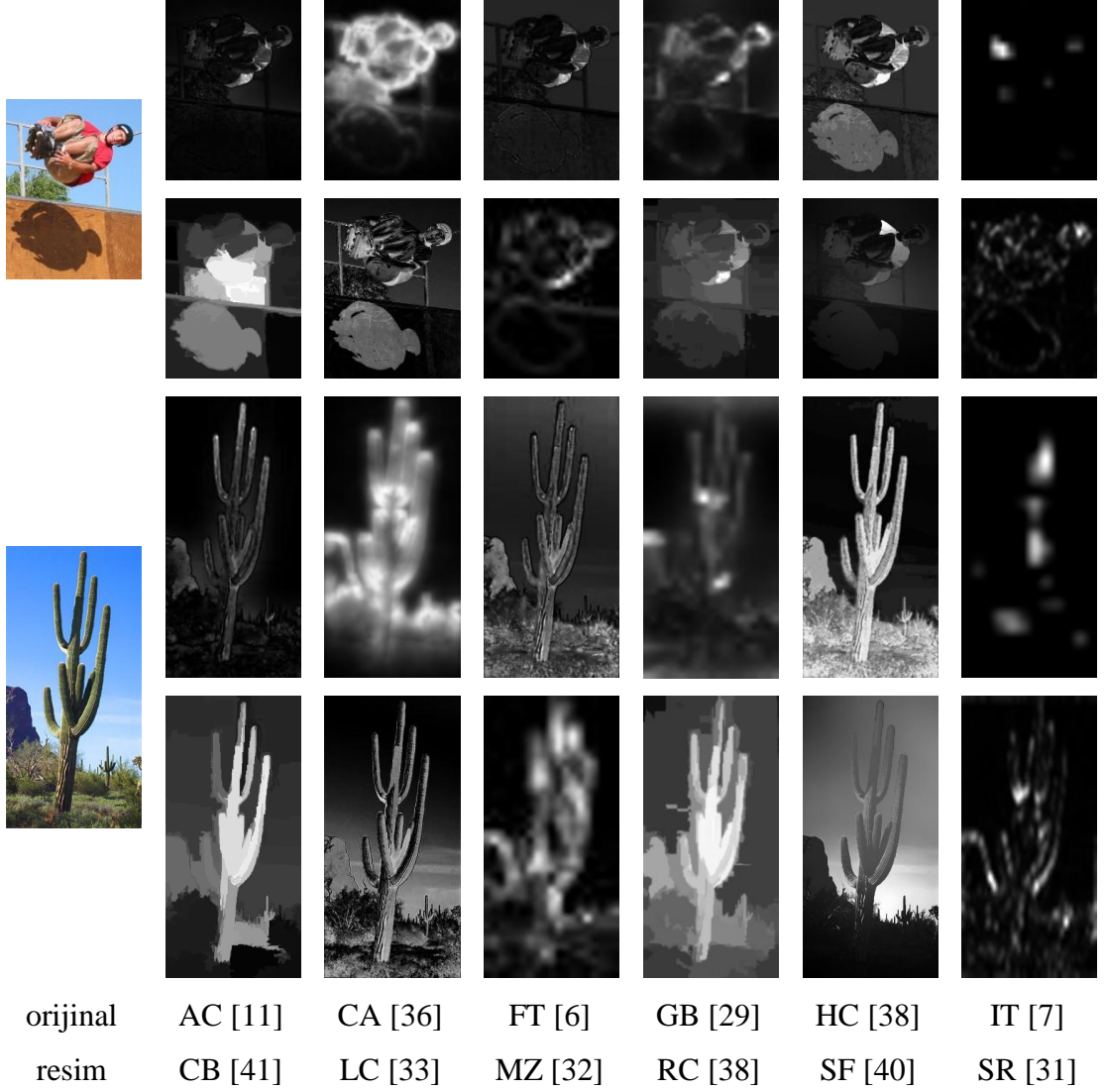
2.1. Belirginlik Tespiti

Önceki bölümde bahsedildiği üzere, belirginlik tespiti metotları yukarıdan aşağıya ve aşağıdan yukarı olmak üzere iki ana kategoride gruplanabilir.

Yukarıdan aşağıya yöntemler [25-27], amaca yönelik, yavaş ve görev bağımlı olup belirginlik bilgisini işlemek için ön bilgiye ihtiyaç duyar ve bu sebeple bellekte çok fazla bilgi tutar. Eğitim safhasında elde ettiği ön bilgiyi, test aşamasında kullanır.

Diğer yöntem olan aşağıdan yukarı ise, veri güdümlü, hızlı ve dikkat öncesi olup ön bilgiye ihtiyaç duymaz, bellek bağımlı değildir. Ön bilgi kullanmadan renk, kenar, doku, parlaklık gibi resimdeki temel bilgileri kullanarak belirginliği onu çevreleyen komşuları arasından tespit etmeye çalışır.

Aşağıdan yukarı yöntemlerde, ilk çalışmalarda [28] renk, parlaklık ve yön bilgileri bir araya getirilmiştir. Sonrasında buna Difference of Gaussians (DoG) kullanılarak merkezilik bilgisi eklenmiştir. [7] Sonrasında, göze çarpan kısımları daha çok vurgulamak için bu çalışma genişletilmiştir. [29] Frekans tabanlı yöntemlerde ise Fourier dönüşümü kullanılarak belirginlik bulunmaya çalışılır. [30,31] Fakat bir önceki yöntemlere nazaran nesnenin belirginliğini ve bütünlüğünü daha iyi ortaya koymasına karşın daha bulanık ve nesnenin tümü yerine kenarlarını daha çok vurgulamaktadır. Şekil 2.1'de bu durumlar gözlemlenebilir. Aşağıdan yukarı yöntemlerde renk uzayı üzerine kurulu yöntemlerde kendi içinde yerel ve genel kontrast analizi olmak üzere ikiye ayrılır.



Şekil 2.1. Popüler yöntemlerden dikkat çekme haritaları

Yerel yöntemler incelenilen pikselin/bölgenin sadece komşu olduğu pikselleri/bölgeleri değerlendirir. Piksel bazında inceleme [32,33], çok boyutlu DoG [34] ve histogram analizi yöntemleri [35] önerilmiştir. Bu yaklaşım daha az bulanık çıktılar üretse de, resimdeki gürültü ve kenar gibi yüksek frekanslı içeriklere daha duyarlıdır. [6]

Genel yöntemler [35-43] ise kontrast analizini resimdeki tüm pikselleri/bölgeleri birbirleriyle karşılaştırarak gerçekleştirir. Sonuç olarak daha iyi sonuç üretseler de piksel tabanlı yaklaşımlarda resim ölçeği büyük olduğunda zaman karmaşıklığı çok fazla olmaktadır. Son zamanlarda bölütleme algoritmalarının gelişmesi, bu algoritmaların belirgin nesne tespitinde kullanılmasına olanak sağlamıştır. Genel

yöntemlerde piksel karşılaştırması yerine bölgesel karşılaştırma yapmak bu yöntemin öncekilerine [38-43] göre tutarlılık ve zaman yönünden daha başarılı olduğunu göstermiştir.

Büyük ölçekli resimlerin de kolaylıkla işlenebilmesine olanak sağlayan bölgesel yaklaşımlar bölütleme algoritmasının başarısına bağımlı olduğu için çok ölçekli, histogram tabanlı kontrast, renk dağılımı gibi yaklaşımlar izlenmiştir.

Algoritmaların çıktısında elde edilen belirginlik haritaları bulanık ve daha çok yerel özellikleri ortaya koyduğundan belirgin nesnenin kenarlarını daha çok vurgulamaktadır. Nesneyi bir bütün olarak vurgulamazlar, sadece dikkatin nereye yoğunlaşması gerektiğine dair ön bilgi verirler. Bu sebeple diğer yöntemler bölütleme, tespit, sıkıştırma vb. gibi içerik tabanlı uygulamalar için daha çok uygundur.

Resimdeki belirgin alanlar genel olarak aşağıdaki özellikleri içermektedirler;

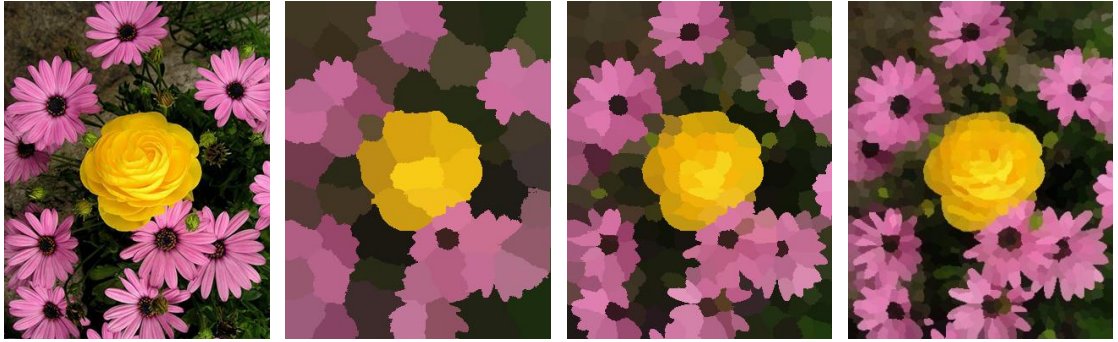
- Belirgin nesneye ait kendisini çevreleyen alanlardan renk bakımından yüksek kontrastlık gösterirler ve resimdeki renk dağılımları nadirdir. Nadir bulunan özellikler daha belirgindir.
- Bu belirgin alanlar resim sınırlarından daha çok resmin merkezinde bulunmaktadır. İnsan görme sistemi ilk olarak resmin merkezine odaklanır.
- Belirgin alan yani resimdeki öne çıkan kısım arka plandaki kısma göre daha küçüktür.
- Arka plan kısmı resim alanına daha çok dağılmıştır. Belirgin alan ise daha küçük ve kompakt olarak bütün haldedir.
- Belirgin alanın kontrast değeri daha çok yakınında bulunan alanlar tarafından belirlenir. Uzakta bulunan alanların etkisi çok fazla yoktur.

2.2. Süperpiksel Yöntemiyle Resim Bölütleme

Resimlerden anlamlı verilerin çıkartılabilmesi ve bilgisayarlı sistemler tarafından yorumlanabilmesi için resimdeki önemli kısımların çıkartılması gerekmektedir. Görüntü bölütleme uzun yıllardır bilgisayarla görü alanında üzerinde çalışılan popüler bir problemdir ve üzerine birçok çalışma yapılmıştır.

Son yıllarda ise, bu konuda çizge tabanlı yöntemler genel eniyilemeyi sağladıklarından öne çıkmaktadırlar. [44] Çizge tabanlı yöntemler bölütlemeyi döngüsel bir yaklaşım ile alt çizgelere ayırarak sağlarlar. Yukarıdan aşağıya yaklaşımın izlenmesi zaman karmaşıklığını arttırmasının yanı sıra karmaşık yüzeylerde detay kaybına sebep olmaktadır.

Şekil 2.2’de bölütleme algoritmasının 64, 256 ve 1024 süperpiksel alanına göre ayrılmış sonuçları yer almaktadır. Süperpiksel alanlarının düzenli, bütün bir yapı sergilediğini ve kenarları iyi kapsadığını görmekteyiz.



orijinal resim

64 bölüt

256 bölüt

1024 bölüt

Şekil 2.2. SLIC algoritması sonuçları

Son yapılan çalışmalarda, detayların korunması ve zaman karmaşıklığının azaltılması için süperpiksel yaklaşımı önerilmiştir. Süperpiksel benzer özellikler sergileyen piksellerin birleştirilip onların tek bir piksel ile ifade edilmesi yöntemidir.

Süperpiksellerin eşit dağılımlı ve dışbükey olması düzgün dağılımlı çizgelerin oluşturulmasını sağlar. Ayrıca düzenli yapıda süperpiksel yaklaşımının izlenmesi,

aşırı bölütlemeye sebep olsa da çizge parçalanmasında oluşabilecek az bölütleme gibi hataların önüne geçmektedir. Kolay algoritması, hız ve sonuçlar bakımından bu konuda son zamanlarda yapılmış en iyi çalışma simple linear iterative clustering (SLIC) yöntemi gösterilebilir. [45]

SLIC pikselleri gruplarken pikseller arasındaki renk benzerliğini ve yakınlığını kullanmaktadır. Bunu sağlarken renk ve konum düzlemini ifade eden 5 boyutlu labxy alanından faydalanır. Buradaki $[l \ a \ b]^T$ vektörü CIELAB renk uzayını ve $[x \ y]^T$ vektörü ise piksellerin konum düzlemini ifade etmektedir.

$$d_{lab} = (l_i - l_j)^2 + (a_i - a_j)^2 + (b_i - b_j)^2 \quad (2.1)$$

$$d_{xy} = (x_i - x_j)^2 + (y_i - y_j)^2 \quad (2.2)$$

$$D = \sqrt{d_{lab} + \frac{m^2}{S^2} d_{xy}} \quad (2.3)$$

Algoritma sadece resmin kaç süperpiksel alanına bölütleneceğini k parametresi olarak alır. Toplam N pikselden oluşan bir resimdeki her bir süperpiksel alanı yaklaşık olarak $\frac{N}{k}$ kadar pikselden oluşmaktadır. Her bir süperpiksel alanının merkezi

$S = \sqrt{\frac{N}{k}}$ ızgara aralığı ile ifade edilmektedir. Bölütlemeye iki piksel arasındaki

mesafenin S uzaklığından fazla olmaması beklenmektedir. S aralık büyüklüğündeki her bir bölgenin merkezi $C_i = [l_i \ a_i \ b_i \ x_i \ y_i]^T$ ve $i = [1, k]$ ile olarak tanımlanır. Denklem 2.3'deki diğer değişken m ise süperpikselin yoğunluğunu belirtmektedir. Büyük m değerleri uzaklık değişkeninin formüldeki etkisini arttıracığı için daha sıkı ve düzgün yapıda süperpiksel alanı elde edilir. CIELAB renk uzayı için bu değişkenin alacağı minimum ve maksimum değer aralığı $[1, 20]$ olarak belirtilmiştir.

2.2.1. Algoritma

İlk olarak, her biri S genişliğinde olacak şekilde resimdeki süperpiksel bölgelerinin merkezleri $C_i = [l_i \ a_i \ b_i \ x_i \ y_i]^T$ belirlenir. Bölgelerin merkezleri belirlenirken 3×3 komşulukta en düşük eğim konumunda olması sağlanır. Böylelikle, merkezlerin kenar üzerinde ve gürültülü piksel olması engellenir. Resmin eğimi aşağıdaki formül ile hesaplanır:

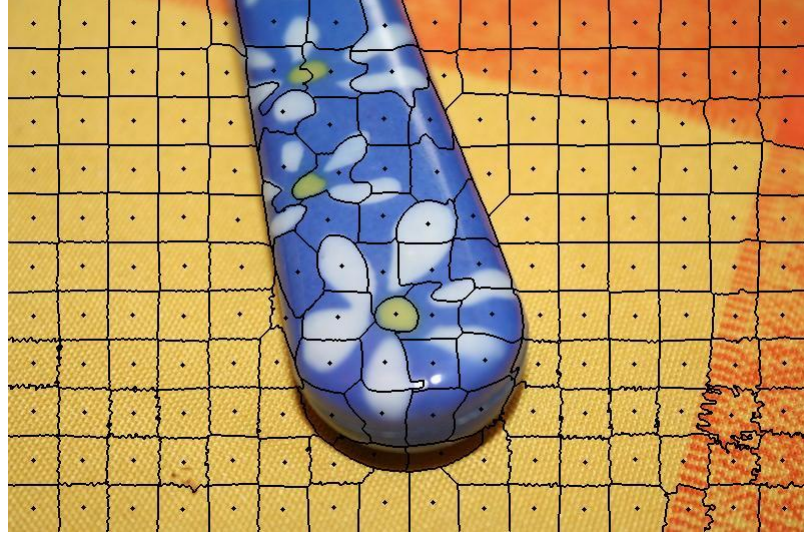
$$G(x, y) = \|I(x + 1, y) - I(x - 1, y)\|_2^2 + \|I(x, y + 1) - I(x, y - 1)\|_2^2 \quad (2.4)$$

Bu formülde, $I(x, y)$ (x, y) piksel konumundaki lab vektörünü ve $\|\cdot\|$ ise L_2 normundaki Öklid uzaklığını ifade eder.

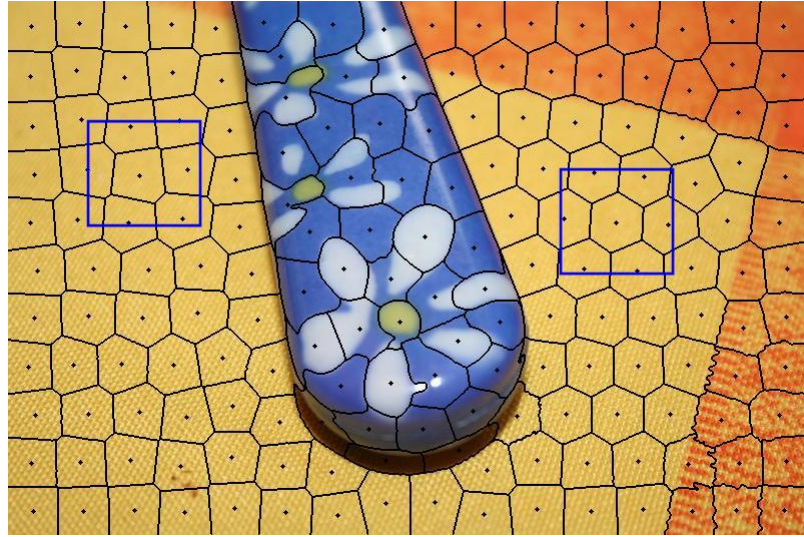
İlk atama işleminden sonra, bütün bölgelerin kendi arama alanlarında kalan pikseller merkezdeki pikseller ile Denklem 2.3 kullanılarak karşılaştırılır. Karşılaştırılan pikseller en iyi eşleşme sağlanan bölgeye atanır. Bu karşılaştırma K-means öbekleme algoritmasında olduğu gibi tüm pikseller ile yapılmaz, sadece bölge merkezinin $2S \times 2S$ komşuluğunda kalan pikseller ile yapılır.

Bu yerel arama sayesinde, daha düzenli süperpikseller oluşturulmakta ve arama zaman karmaşıklığı azaltılmaktadır. Bütün bölgeler için piksel ataması yapıldıktan sonra, bölgeler için yeniden merkez hesaplaması yapılır. Ve piksellerin yakın bölgelere atanması işlemi beş kez tekrarlanarak döngü bitirilir.

Süperpiksel öbeklemesinin sonunda, bölgelere atanmamış veya bölgesi ile komşuluğu olmayan pikseller kalmış olabilir. Bütünlüğü sağlamak için, bu pikseller kuyruk tabanlı taşma doldurma (flood fill) algoritması kullanılarak kendisine komşu yakın bölgeye atanır ve bölütleme tamamlanır.



Şekil 2.3. SLIC algoritmasının 1 döngü sonrası



Şekil 2.4. SLIC algoritmasının 10 döngü sonrası

Şekil 2.3 ve 2.4'de [45] SLIC algoritmasının sırasıyla 1 ve 10 döngü sonrasındaki bölütlemelerinin sonuçlarını görmekteyiz. Resimlerde siyah nokta ile süperpiksel bölgelerinin veya alanlarının merkezleri işaretlenmiştir. İki piksel arasındaki maksimum uzaklığın S genişliğinde olması süperpiksel bölgelerinin bütünlük yapısını korumasını sağlamıştır. Resimdeki mavi kareler algoritmanın $2S \times 2S$ genişliğindeki arama penceresini göstermektedir. Bu arama alanı içindeki pikseller bölge merkezi ile karşılaştırılarak yakınlığa göre gruplanmaktadır.

Algoritma 2.1: SLIC Sözdde Kodu

- 1: **Girdi:** Süperpiksel sayısı
 - 2: **Çıktı:** Bölütlenmiş resim
 - 3: Girdi olarak alınan süperpiksel sayısı göre S değerini hesapla. Her biri S genişliğinde olacak şekilde süperpiksel bölgelerinin merkezlerini $C_i = [l_i \ a_i \ b_i \ x_i \ y_i]^T$ belirle.
 - 4: $n \times n$ komşulukta Denklem 2.4'ü kullanarak merkezleri tekrar konumlandır.
 - 5: **repeat**
 - 6: **for** bütün merkezler için C_i **do**
 - 7: İşlem yapılan merkez noktasının $2S \times 2S$ komşuluğunda kalan piksellerden hangileri Denklem 2.3 sonucu merkez noktasıyla benzeşiyorsa, bu pikseli merkez noktasının kümesine ekle.
 - 8: **end for**
 - 9: Bölgelerin yeni merkezlerini ve eski merkezlerle olan E hata farkı değerini hesapla.
 - 10: **until** $E \leq threshold$
 - 11: Bağlı olmayan veya kümeye atanmamış pikselleri yakın küme değerlerine ata.
-

2.3. Karmaşık Ağ

Çizge teorisinin bir parçası olarak yer alan ağ teorisi bilgisayar ve ağ biliminin bir alanıdır. Birçok disiplinde farklı uygulamalarına rastlanabilir. Karmaşık ağların analizinde, ağın bağlantı karakteristikleri irdelenerek önemli düğümlerin, toplulukların, bağların ve ilgili alanların çıkarımı yapılır. Bu tür bir analiz, ağların küresel evrim davranışları için iyi bir bakış açısı kazandırır.

Birçok gerçek ağ yapısı bünyesinde topluluk yapısı barındırır ve içerisindeki bazı gruplara diğerlerinden daha kuvvetli bağlıdır. Topluluk çıkarımının yapılması ve karakteristiğinin belirlenmesi, aynı topluluğu paylaşan öğelerin çoğu özelliğinin benzerlik göstermesinden dolayı önemlidir.

Karmaşık ağ bağlamındaki en yaygın yapısal problem toplulukların tespit edilmesidir. Toplulukların tespit edilmesi problemi, ağdaki yapısal olarak ilişkili grupların bulunması problemiyle yakından alakalıdır. Bu yapısal olarak ilişkili gruplar, topluluk olarak adlandırılmaktadır. Bu problemler, geleneksel olarak yapılan çalışmalar da çizge bölümlenme olarak fazlasıyla yer edinse de, karmaşık ağlar boyut açısından burada yapılan çalışmalardakilerle karşılaştırılmayacak derecede büyüktür. Öte yandan ağlar, topluluk keşfinin verimliliğini önemli derecede artıracak kadar içeriğe sahip olabilirler. Bu da topluluk belirlemede birbirinden farklı birçok algoritmanın geliştirilmesinde etkili olmuştur. Örneğin, içerik topluluklardaki düğümlerle ilintili olabilir, bunun ağdaki kümelemenin kalitesini artırmada ne kadar önemli olduğu [46]'de gösterilmiştir.

2.3.1. Topluluk Çıkarımı

Karmaşık ağ analizi, araştırmacılar için ağdaki grupları (küme, topluluk) belirlemek adına bir dizi araç sağlar. Biçimsel olmayan tanımıyla, bir ağda yer alan bir topluluk, ağın geri kalanına kıyasla kapsadığı düğümlerin birbirlerine daha sıkı bir biçimde bağlı olması şeklindedir. [47,48] Bu tanım birkaç değişik yolla, genellikle bir kalite fonksiyonuyla genelleştirilmiştir. Kalite fonksiyonu, ağın topluluklara bölümünün iyiliğini ölçer. Bu kalite ölçütlerinden bazıları, örneğin Modülerlik [49] ve Normalize Kesim [50], diğerlerine göre daha yaygın olmalarına rağmen, tek bir ölçüt tüm durumlar için uygulanabilir olmadığından hiçbiri evrensel geçerlilik kazanmamıştır. Topluluk çıkarımı için ortaya atılan algoritmalar problemlere yaklaşımlarına ya da performans karakteristiklerine göre birçok önemli boyuta göre değişebilir. Algoritmaların yaklaşımındaki bir önemli nokta da belirli bir kalite metriğini optimize edip etmemeleridir. Ağların topluluklara bölümü için birçok algoritma geliştirilmiştir.

2.3.2. Kalite Fonksiyonu

Bir çizgenin kümelere ayrılmasının iyiliğini artırmak için literatürde çeşitli kalite fonksiyonları ve ölçütleri ortaya atılmıştır.

2.3.2.1. Normalize Kesim

A, ağın veya çizgenin komşuluk matrisi; $A(i, j)$, i ve j düğümleri arasındaki kenarın ağırlığı; V , ağdaki veya çizgedeki köşeler ya da düğümler olmak üzere bir grup $S \subset V$ köşesinin normalleştirilmiş bölümlendirilmesi:

$$Ncut(S) = \frac{\sum_{i \in S, j \in \bar{S}} A(i, j)}{\sum_{i \in S} degree(i)} + \frac{\sum_{i \in S, j \in \bar{S}} A(i, j)}{\sum_{j \in \bar{S}} degree(j)} \quad (2.6)$$

denklemiyle tanımlanmıştır [50,51].

Bir başka deyişle, bir grup S düğümünün normalleştirilmiş bölümlendirilmesi, S 'nin ve çizgenin geri kalanı \bar{S} 'nin toplam köşe ağırlığı ile normalleştirilmiş, S 'yi çizgenin geri kalanına bağlayan köşelerin ağırlıkları toplamıdır.

Bir grup $S \subset V$ düğümünün iletkenliği yakından ilişkilidir ve şöyle tanımlanmıştır [52]:

$$Conductance(S) = \frac{\sum_{i \in S, j \in \bar{S}} A(i, j)}{\min(\sum_{i \in S} degree(i), \sum_{i \in \bar{S}} degree(i))} \quad (2.7)$$

$\{V_1, \dots, V_k\}$ şeklinde k kümeye bölünmüş bir çizgenin normalize kesimi ya da iletkenliği, her bir $V_i \{i = 1, \dots, k\}$ kümenin normalize kesiminin ya da iletkenliğinin toplamıdır [53].

2.3.2.2.Modülerlik

Modülerlik, son zamanlarda bir çizgenin kümelenmesinin iyiliğini ölçmek adına oldukça yaygın hale gelmiştir. Modülerliğin avantajlarından biri çizgenin bölümlendirildiği küme sayısı bakımından bağımsız olmasıdır. $\{V_1, \dots, V_k\}$ şeklinde k kümeye bölünmüş bir çizgenin modülerliği;

$$Q = \sum_{i=1}^k (e_{ii} - a_i^2) \quad (2.8)$$

denklemiyle tanımlanmıştır. Formüldeki, e_{ii} i kümesinde bulunan kenar sayısını; a_i^2 ise aynı derece de rastgele üretilmiş çizgede bir kenarının i kümesinde olması beklenen kenar sayısını ifade eder. Modülerlik değeri 0 ile 1 arasındadır ve 1'e yaklaştıkça topluluğun şans eseri var olmadığı kesinlik kazanır. Modülerliğin 0,3 değerinden büyük ve eşit durumlarda ağ yapısındaki ilişkilerin önemli ve anlamlı olduğu yapılan testler ile ortaya konulmuştur. Topluluk çıkarımı yapılırken tüm olasılıklar değerlendirildiği için arama işleminin hesaplama maliyeti çok yüksektir: Modülerlikte bu Q değerini eniyilemek NP-Zor bir problemdir. [54] Bu sebeple, maksimum modülerlik değerine yakın sonuçlar veren hesaplama zamanı çok fazla olmayan çeşitli algoritmalar önerilmiştir. [47]

2.3.3. Topluluk Çıkarma Yöntemleri

2.3.3.1.Girvan ve Newman'ın Bölücü Algoritması

Newman ve Girvan kenar arasındalığı (betweenness) fikrini kullanarak, topluluk çıkarımı için bölücü bir algoritma önermişlerdir. Kenar arasındalığı ölçütü, yüksek kenar arasındalığı puanlı kenarların farklı toplulukları bağlayan kenarlar olmaları ihtimalinin yüksek olması üzerine kurulmuştur. Yani, topluluklar arası kenarların, topluluk içi kenarlardan daha yüksek kenar arasındalığı puanlarına sahip olmaları

tasarlanmıştır. Bundan dolayı, yüksek arasındalık puanlı bu kenarların belirlenip çıkarılmasıyla, sosyal ağ topluluk bileşenlerine ayrılabilir.

En kısa yol arasındalığı, kenar arasındalığı ölçütüne verilebilecek örneklerden biridir. Buradaki düşünce, az sayıda topluluk arası kenar olmasından dolayı, farklı topluluklara ait düğümler arasındaki en kısa yollar, bu az sayıdaki topluluklar arası kenarlar üzerinden geçmeye zorlanacaktır şeklindedir. Ayrıca, rassal yürüyüş arasındalığı ile akım debisi arasındalığı kenar arasındalığı iki diğer farklı örneğidir. Rassal yürüyüşte, herhangi iki düğümü bağlayan yol seçimi, en kısa yolda olduğu gibi kesel değil rassal yürüyüşün bir sonucudur. Akım debisinde ise devre teorisinden yola çıkılmıştır. Öncelikle, ağ sanal olarak bir direnç ağına dönüştürülerek, ağdaki her bir düğüm birim dirençle yer değiştirmiştir ve de iki düğüm kaynak ve hedef olarak belirlenmiştir. Daha sonra, her bir kenar için arasındalık, tüm olası düğüm çiftlerinin seçilerek üzerlerinden geçen mutlak akımların toplamı hesaplanarak ölçülmüştür.

Algoritmalarının genel şekli şu şekildedir:

1. Ağdaki her bir kenar için herhangi bir ölçütü kullanarak arasındalığı hesapla.
2. En yüksek puanlı kenarı bul ve ağdan çıkart.
3. Geri kalan kenarlar için arasındalığı yeniden hesapla.
4. Adım 2'den tekrar et.

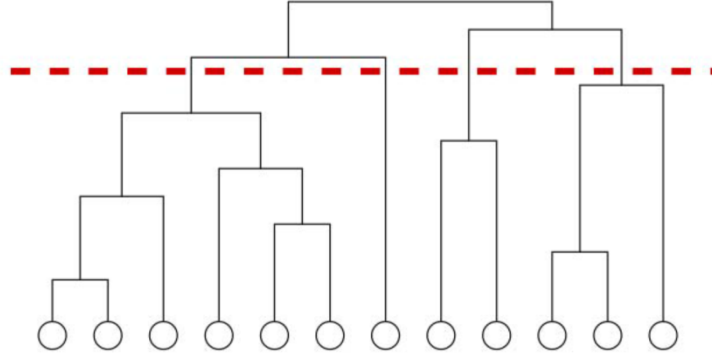
Yukarıdaki işlem yeteri kadar az sayıda topluluk elde edilene kadar devam ettirilir. Ayrıca, toplulukların hiyerarşik yapısının da elde edilmesi işlemin diğer bir sonucudur. Kenar arasındalığı için farklı ölçütlerin kullanılmasının muhtelif topluluk yapıları ortaya çıkaracağı fikrinin aksine, deneyler hangi arasındalık ölçütünün kullanıldığının o kadar da önemli olmadığı gerçeğini ortaya çıkarmıştır. Tekrar hesaplama adımı devam ettiği sürece, farklı ölçütler tarafından oluşturulan sonuçların birbirlerinden çok az farklılık gösterdikleri saptanmıştır. Tekrar hesaplama adımının nedeni şu şekilde açıklanabilir: Eğer kenar arasındalığı puanları sadece bir kere

hesaplansaydı ve kenarlar azalan sırada ağdan çıkarılsaydı bu puanlar güncellenemeyecek ve de kenarların çıkarımından sonra oluşan yeni ağ yapısını yansıtamayacaklardı. Bu nedenle, yeniden hesaplama adımı, tatmin edici sonuçların elde edilmesi açısından algorithmadaki en önemli adım olarak göze çarpmaktadır. Bu yöntemin en büyük dezavantajı yüksek hesaplama maliyetidir: Açıkça tüm kenarlar için arasındalığı hesaplanması $O(mn)$ zaman alırken, tüm algoritma m döngü için $O(m^2n)$ son olarak $O(n^3)$ zaman gerektirmektedir. Burada n çizgedeki düğüm sayısını, m ise çizgedeki kenar sayısını ifade etmektedir. [47]

2.3.3.2.Hızlı Ağgözlü Algoritması

Newman modülerliği eniyilemek için ağgözlü hiyerarşik toplayıcı bir kümeleme algoritması önermiştir. Algoritmanın altında yatan temel yapı, bir topluluğu temsil eden küme grupları daha geniş topluluklar elde etmek için başarılı bir şekilde ikili olarak birleştirilir öyle ki; her bir birleşimde ağın modülerliği artırılmaktadır esasına dayanır. Başlangıçta her bir düğüm bir topluluğa aitken, her bir adımda birleşimi modülerliği en çok artıran topluluklar seçilir. Burada en azından bir kenarı paylaşan topluluklar göz önünde bulundurulmalıdır. Çünkü hiçbir kenarı paylaşmayan toplulukların birleşimi modülerliğin artırılmasına yönelik bir sonuç vermemektedir. Bundan dolayı bu adım $O(m)$ zamanda gerçekleştirilir. Ayrıca, mevcut bölümlenmede her bir çift topluluk tarafından paylaşılan kenarların oranını muhafaza eden veri yapısı da sunulmuştur. Bu veri yapısını güncellemek en kötü durumda $O(n)$ zaman almaktadır. Toplamda $n - 1$ tekrarlama (birleşim) olduğundan, algoritma seyrek çizgeler için $O((m + n)n)$ veya $O(n^2)$ zaman gerektirmektedir. Burada n çizgedeki düğüm sayısını, m ise çizgedeki kenar sayısını ifade etmektedir. [55] Max-heaps gibi etkili veri yapıları kullanılarak bu algoritmanın zaman karmaşıklığı $O(md(\log n))$ olarak geliştirilmiştir. Burada d ağacın yüksekliğini temsil etmektedir. Seyrek ve hiyerarşik ağlarda $m \sim n$ ve $d \sim \log n$ olarak ifade edilebildiği için algoritma $O(n \log^2 n)$ zaman karmaşıklığında çalışabilmektedir. [47]

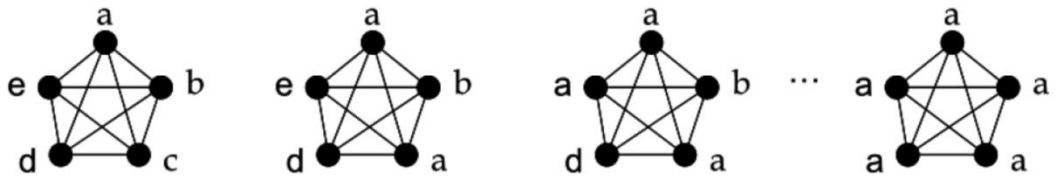
Şekil 2.5’da [56] dendogram ile ağ yapısı tanımı yapılmıştır. Resimdeki dairelerin her biri ikişerli bağlanmış ağ düğümünü temsil etmektedir. Tek bir topluluk olana kadar ikişerli eşleşmeye devam edilir. Kırmızı kesikli çizgi maksimum modülerliği ifade etmektedir ve topluluk eşleşmesi bu sınıra kadar devam ettirilir.



Şekil 2.5. Dendogram ile ağ gösterimi

2.3.3.3. Etiket Yayılımı

Etiket yayılımı yöntemi [57] düşük zaman karmaşıklığı toplulukların tespitinde avantaj sağlamaktadır. Başlangıçta, her kümeye farklı etiket verilir. Her döngüde, bütün kümeler sırayla taranarak, her kümeye komşularının çoğu tarafından paylaşılan etiket atanır. Eğer tekil bir çoğunluk yoksa çoğunluktan rastgele bir etiket seçilir. Sonuç itibariyle, bazı etiket çoğunlukları zamanla kaybolurken bazıları tüm ağ boyunca yayılmaya devam eder. Çizgedeki her bir küme çoğu komşusunun sahip olduğu aynı etiketi paylaşıyorsa döngü sonlandırılır.



Şekil 2.6. Etiket yayılım yöntemi ile ağ etiketlenmesi

Şekil 2.6’de soldan sağa doğru ağ yapısındaki düğümlerin teker teker etiketlenmesi ve yöntemin çalışma şekli gösterilmiştir. Tekil bir ağırlık olmadığı için ilk başta rastgele seçilen değer yüksek kenar yoğunluğundan dolayı yöntem sonunda baskın hale gelip tüm düğümleri kendisiyle aynı kümede olmasına sebep olmuştur.

Komşu kümeler tekil çoğunluk içermiyorsa, yöntem sonuçları da tekil değildir. Böyle durumlarda rastgele etiket ataması yapıldığı için algoritmanın her çalıştırılmasında farklı sonuçlar elde edilecektir. Yöntem her döngü için $O(m)$ zaman karmaşıklığına sahiptir ve döngü sayısına göre bu karmaşıklık artmaktadır. Buradaki m çizgedeki kenar sayısını ifade etmektedir. Çizge büyüklüğü bu çalışma zamanını etkilemediği için büyük çizgelerin hızlı analiz edilmesinde kullanılmaktadır.

2.4. Özellik Noktalarının Tespiti

Özellik noktalarının başka bir deyişle anahtar noktaların kolay tespit edilebilmesi için ayırt edici noktalar olması gerekmektedir. Bu bağlamda, kenar veya diğer düzlemsel bölgeler yerine köşe noktalarından seçilen örnekler daha fazla özelliğe sahip olacaktır.

Köşe tespit algoritmaları kenar tespiti ve parlaklık tabanlı olmak üzere ikiye ayrılmaktadır. Kenar tespiti tabanlıda önde gelen algoritmalara CSS ve ECSS; parlaklık tabanlı da ise Moravec, Harris ve Shi-Tomasi yöntemleri örnek verilebilir. Kenar tespit tabanlı algoritmalar parlaklık tabanlı olanlara göre köşe tespitinde daha başarılı olsalar da çalışma zamanları yüksektir.

2.4.1. Moravec Köşe Tespit Yöntemi

Yöntemin temeli [58], seçilen bir pikselin komşuluğundaki alanın çevresindekilerle karşılaştırılmasına dayanır. Komşuluk alanının aralarında 45° açı farkı bulunan dört farklı yönde kaydırılmasıyla elde edilen çevre alandaki piksellerin parlaklığı ile kaydırılmadan önceki alandakilerle farkının karşılaştırılmasına göre köşe noktalar tespit edilmektedir. Yöntemin formülü aşağıdaki gibidir:

$$E(u, v) = \sum_{x,y} w(x, y) [I(x + u, y + v) - I(x, y)]^2 \quad (2.10)$$

$$C = \min (E(0,1), E(1,1), E(1,0), E(-1,1)) \quad (2.11)$$

denkleminde $w(x, y)$ büyüklüğündeki komşuluk alanındaki piksel parlaklık değerleri ile, alanın (u, v) yönünde kaydırmalar ile elde edilen alandaki piksellerin parlaklıkları arasındaki farkı göstermektedir. Dört yöndeki kaydırmanın en küçük değeri köşe bulmada ölçüt olarak kullanılacaktır. Eğer bu değer yerel maksimum ve belirli bir eşik seviyesinden yüksek olduğu zaman köşe noktası olarak kabul edilecektir.

2.4.2. Harris Köşe Tespit Yöntemi

Harris köşe tespit algoritması [59] Moravec yönteminin üç eksiğinin giderilmesiyle oluşturulmuştur.

İlki, Moravec sadece 45°'lik kaydırmalar ile hesaplama yaptığından dolayı dönüşten bağımsız değildir. Harris önerdiği yöntemde tüm yönlerdeki değişime bakılması gerektiğini belirtmiştir. Bunu Taylor açılımındaki tüm küçük yönlerdeki kaydırmaları hesaba katarak aşağıdaki şekilde geliştirmiştir:

$$E(u, v) = \sum_{x,y} w(x, y) [I_x u + I_y v - O(u^2, v^2)]^2 \quad (2.12)$$

$$E(u, v) = Au^2 + 2Cuv + Bv^2 \quad (2.13)$$

$$A = \sum_{x,y} w(x, y) I_x^2(x, y)$$

$$B = \sum_{x,y} w(x, y) I_y^2(x, y) \quad (2.14)$$

$$C = \sum_{x,y} w(x, y) I_x(x, y) I_y(x, y)$$

İkinci olarak, komşuluk alanını ifade eden $w(x,y)$ pencere fonksiyonu iki değer almasından dolayı algoritma gürültüye karşı dayanaksızdır. Harris bunun yerine Gauss fonksiyonunun kullanılmasını önermiştir.

$$w(x,y) = \exp\left(-\frac{(x^2 + y^2)}{2\sigma^2}\right) \quad (2.15)$$

Gauss fonksiyonunun kullanılmasıyla eşitlik matris formuna dönüştürülür.

$$E(u,v) \cong [u,v]M \begin{bmatrix} u \\ v \end{bmatrix} \quad (2.16)$$

$$M = \begin{bmatrix} A & C \\ C & B \end{bmatrix} \quad (2.17)$$

M matrisinin özdeğerleri (λ_1, λ_2) hesaplanarak pikselin sınıflandırılması yapılır. Son eksiklik, Moravec yöntemindeki gibi köşe noktaların yakın ölçüt değerlerine sahip olmasının önüne geçilerek giderilmiştir.

$$R = \det M - k(\text{trace} M)^2 \quad (2.18)$$

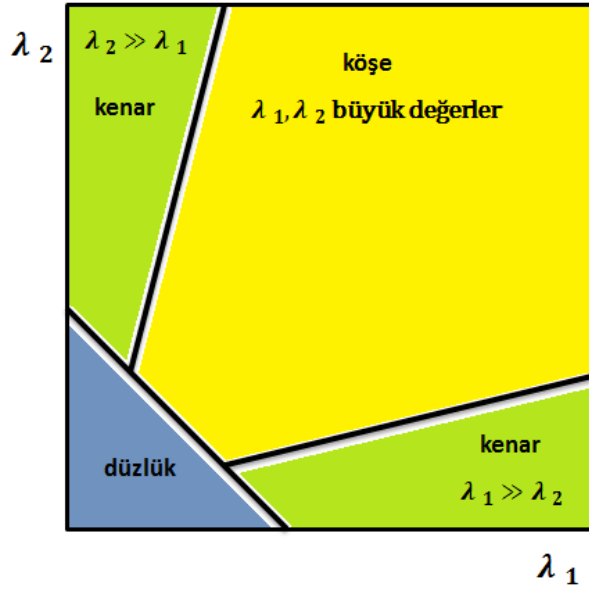
$$\det M = \lambda_1 \lambda_2 \quad (2.19)$$

$$\text{trace} M = \lambda_1 + \lambda_2 \quad (2.20)$$

Özdeğerlerin hesaplanmasının zaman karmaşıklığı fazla olduğundan dolayı daha kolay hesaplanabilen formül önerilmiştir:

$$R = AB - C^2 - k(A + B)^2 \quad (2.21)$$

denkleminde k sabit olup 0,04 ile 0,06 arasında değerler kullanılması önerilmiştir. R fonksiyonun büyük pozitif değerde olması köşeye, büyük negatif değerde olması kenara ve küçük değerler alması ise düz bölgeye karşılık gelmektedir. Şekil 2.7'da bu sonuçlara karşılık gelen alanların gösterimi yapılmıştır.



Şekil 2.7. R fonksiyonu bölgeleri

2.4.3. Shi-Tomasi Köşe Tespit Yöntemi

Shi-Tomasi köşe tespit yöntemi [60] Harris algoritmasının geliştirilmesi ile oluşturulmuştur. Sadece seçme ölçütü farklı olmasına rağmen Harris yöntemine göre daha iyi sonuçlar vermektedir.

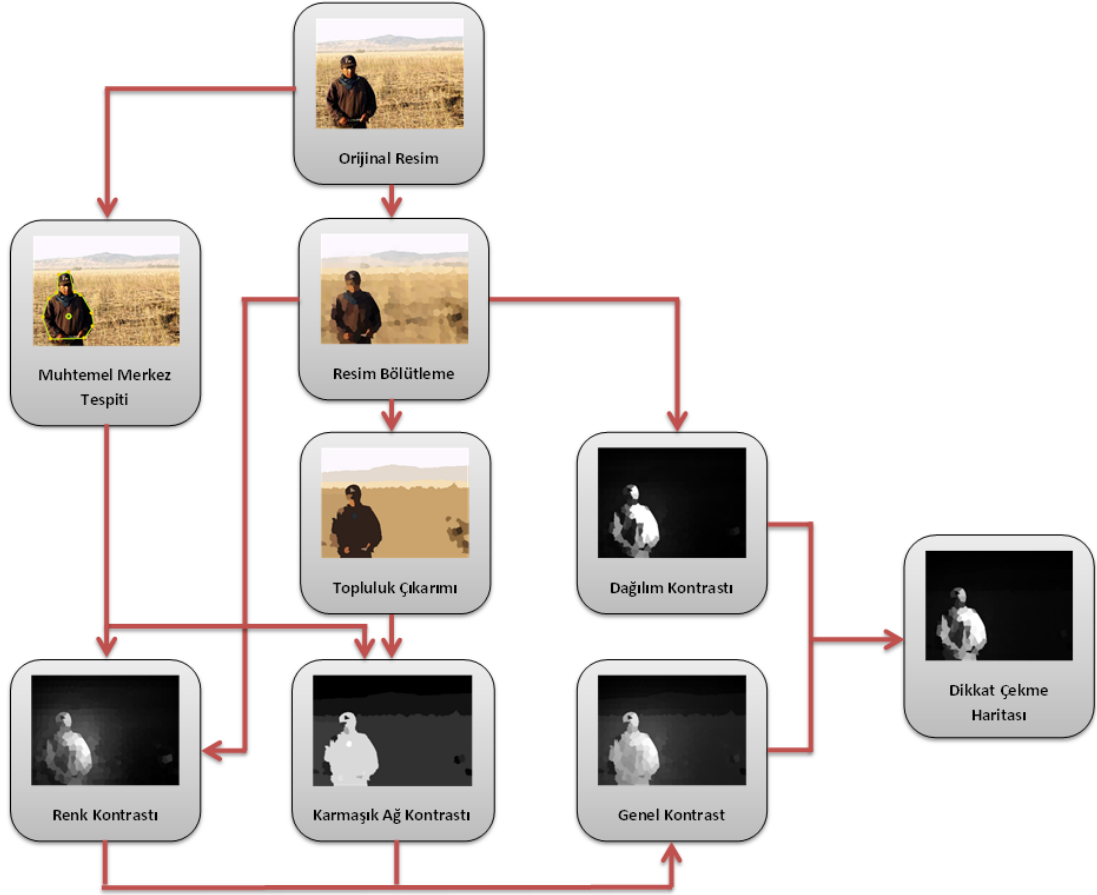
Harris yöntemi doğruluktan ödün vererek zaman karmaşıklığı düşük çalışmaktadır. Shi-Tomasi yönteminin zaman karmaşıklığı daha yüksek olmasına rağmen özdeğerleri hesapladığı ve sadece özdeğerlerin kendisini değerlendirdiği için doğruluğu Harris yöntemine göre daha yüksektir.

$$R = \min(\lambda_1, \lambda_2) \quad (2.22)$$

denkleminde R fonksiyonun büyük pozitif değerde olması köşeye, (λ_1, λ_2) değerlerinin birisinden küçük olması kenara ve (λ_1, λ_2) değerlerinin ikisinde de küçük olması ise düz bölgeye karşılık gelmektedir.

3. BÖLGESEL BİLGİ KULLANARAK KARMAŞIK AĞ DESTEKLİ BELİRGİN ALAN TESPİTİ

3.1. Genel Bakış



Şekil 3.1. Yöntemin akış şeması

Önerdiğimiz algoritma ilk olarak resmi süperpiksellerine bölütleme yaptıktan sonra kontrast hesaplaması için bölütleme yapılan süperpikseller üzerinden tekillik, karmaşık ağ tekilliği ve dağılım ölçütlerini kullanarak belirginlik haritası oluşturur. Algoritmanın izlediği adımlar Şekil 3.1’de gösterilmiş ve aşağıda belirtilmiştir.

3.1.1. SLIC Resim Bölütleme

Buradaki amaç resimdeki değerli kısımları korurken, gürültüye sebep olabilecek alakasız ve az öneme sahip kısımlardan ayrıştırarak resmi temel özelliklerine göre parçalara ayırmaktır. Resimdeki piksellerin renk, konum vb. gibi özellikleri kullanılıp türdeş alanlar oluşturularak öbeklenmesi bize bu temel özelliklerin korunmasını sağlar. Ayrıca bu öbeklemenin alanlar arasındaki keskin kontrast geçişleri, sınır ve kenar özellikleri vb. değerleri korunarak, önceden belirlenen şekil ve boyut konusundaki kısıtları sağlanarak sağlanmalıdır. [45]

Sonuç olarak elde edilen sıkı bir yapıdaki, iyi bir örüntü sergileyen, temel özelliklerini kaybetmeden oluşturulan süperpiksel alanları kontrast hesaplamasında bize daha gülbüz ve tutarlı sonuçlar elde etmemizi sağlayacaktır.

3.1.2. Renk Kontrastı

Kontrast ölçütlerimizden ilki olan renk kontrastı, süperpiksel bölgelerinin birbirleriyle olan benzerlik ilişkilerini inceleyen ve incelenilen bölgelerden hangilerinin diğerlerinden daha farklı olduğunu tespit etmemizi sağlayan bir ölçüttür. Başka bir deyişle, bu bölgeler arasında tekiliği vurgulayan, çok sık seyretmeyen, olağan dışı nadir özellikler sergileyen bölgeleri bulma arayışıdır.

Bu yaklaşım önceki kontrast tabanlı birçok uygulamada [38-43] kullanılsa da, bu uygulamalarının çoğunun piksel tabanlı olması istenilen sonuçların elde edilmesine engel olmuştur. Çoğu piksel tabanlı çalışma piksellerdeki gürültüye, karmaşık doku içeren yapılardaki ani kontrast değişimlerine, vb. gibi analiz sonuçlarındaki başarıyı düşürecek olumsuzluklara ek olarak yüksek çalışma zamanlarına sahiptir. Ayrıca, çok boyutlu yöntemlerde işleyişlerinden ötürü bu bilgileri kaybedebilmeleri yanı sıra bulanık sonuçlar vermektedirler. Önerilen algoritmada süperpiksel bölütlemesinin sağladığı türdeşlik, sıkı ve simetrik yapı gibi özellikleri bahsedilen problemlerin üstesinden gelinmesine yardımcı olmuştur.

3.1.3. Karmaşık Ağ Kontrastı

Çok boyutlu yaklaşımlar ürettikleri belirginlik haritalarını üst üste toplayıp yani farklı sonuçların iyi yönlerini bir araya getirip düzgün olarak dikkat çekme haritaları elde etmelerine karşın yöntemlerin çalışma zamanları oldukça fazladır.

Önceki adımın sonucu olarak elde edilen dikkat çekme haritasında belirgin alanların parlaklığı bir bütün olarak ifade edilememektedir. Bu problem belirgin bölgelerin aşırı bölütlenmesinin sonucu olarak görülebilir. Sonuç olarak, tek bir bölge olarak ifade edilmeyen belirgin bölge farklı parlaklıklara sahip olarak parçalı ve yamalı bir yapıda gözükmemektedir. Bir önceki bölgesel tabanlı yaklaşımlarda da [38,40] aynı problemle karşılaşmıştır ve bazılarında [42] bölgeleri aynı parlaklıkta ifade edebilen yumuşatma üzerine dayalı yöntem önerilerinde bulunulmuştur. Bu yumuşatma yöntemi, aynı renk benzerliği sergileyen fakat aşırı bölütleme sonucu farklı bölgelere düşmüş piksel grupları arasındaki parlaklık farkını azaltıp birbirlerine yaklaştırarak daha türdeş alan sonuçları elde edebilmektedir.

Önerilen yöntemde, aynı şekilde yumuşatma üzerine dayalı bir yaklaşım izlense de içerik ve uygulama olarak farklı bir yöntem yaklaşımı benimsenmiştir. Temel yaklaşım aşırı bölütlemeye uğramış bölgelerin tekrar birleştirilmesi üzerinedir. Bunu sağlamak için, karmaşık ağ teorisinde bulunan toplulukların çıkarımı üzerinde çalışılan probleme uyarlanmıştır. Süperpiksel bölütlerinin her biri ayrı küme olarak değerlendirilip bunlar üzerinden topluluk çıkarımı yapılmıştır. Yöntem sonucu olarak, aşırı bölütleme sonucu ayrı düşmüş bölgeler aynı topluluk altında tekrar birleştirilmiştir. Bir nevi bölgesel birleştirme uygulanarak hangi süperpiksel bölgelerinin bir küme topluluğu altında gruplanacağı tanımlanmıştır.

Birleştirme aşamasından sonra, yeni oluşturulan bölgeler arasında başka bir deyişle topluluklar arasındaki tekillik incelenerek nadir bulunan, seyrek seyreden ve göze çarpan bölgeler tespit edilir. Bir önceki adımda bulunan yöntemle birebir bir yaklaşım izlenir. Bu yöntem sonucu belirgin bölgeler daha toplu ve bütün olarak

göze hitap etmektedir. Elde edilen belirginlik haritası önceki yöntem sonucuyla birleştirilerek daha tutarlı sonuçlar elde edilir.

3.1.4. Dağılım

Belirgin bölge olmayan kısımlar yani arka planlar bütün resim boyunca yüksek değişiklik göstererek dağılmıştır. Belirgin bölgeler yani ön plan kısımları ise resme saçılmadan bir bütün olarak yer alır. Son ölçüt olan dağılımda, belirgin bölgeler için bütünlük ve konum verisini kullanarak bir tanımlama yapılmıştır. Böylelikle, tekillik özelliği sergileyen belirgin bölgelerin bir arada buldukları zaman daha belirgin olması gerektiği için tekillik özelliği sergileyen dağınık arka plan bölgeleri için kısıt sağlanmış olur. [35,36]

3.2. Metodoloji

Bu bölümde, daha önce aşamalarından bahsettiğimiz yöntemin detaylı incelemesi ve yorumlanması bulunmaktadır.

3.2.1. SLIC Resim Bölütleme

Renk kontrastını hesaplamanın en kolay yolu bütün pikselleri birbiriyle karşılaştırmaktır. Ama bu yöntemin $O(m^2)$ zaman karmaşıklığı vardır. Buradaki m resimdeki tüm piksel sayısıdır. Bu zaman karmaşıklığını azaltmak için histogram tabanlı bir hesaplama [38] veya resim bölütleme yöntemi [38-43] ([38] bu çalışmada hem piksel tabanlı yaklaşım hem de bölgesel yaklaşım önerilmiştir) tercih edilebilir.

Son yıllarda geliştirilmiş en iyi ve hızlı resim bölütleme algoritması olarak SLIC [45] gösterilebilir. SLIC bölütleme yöntemi [45], resmi kullanıcıdan aldığı süperpiksel sayısı kadar bölüme ayırır. Belirli bir boyuta ve düzene sahip süperpiksel bölgeleri, kendisine özellik olarak yakın pikselleri temsil etmesi ve hesaplamaların sadece resimde bulunan bu bölgeler üzerinden yapılması çalışma zamanını büyük ölçüde

düşürmektedir. Ayrıca bölütleme sayesinde piksellerin barındırdığı gereksiz ayrıntılar ve gürültüden de soyutlanma sağlanır.

SLIC algoritmasında [45] insan görme sistemini iyi karakterize etmesi ve cihaz bağımsız olmasından dolayı CIELAB renk uzayı kullanılması tercih edilmiştir. Süperpiksel bölütleri $R_i, i = 1, 2, 3, \dots, k$ ile ifade edilmektedir. Buradaki k bölüt sayısını temsil etmektedir ve önerilen yöntemde $k = 500$ seçilmiştir. Bu seçim bölütleme algoritmasının çalışma zamanı ve önerilen yöntemin bu bölüt sayısındaki başarı sonucu arasındaki ödünleşmeye göre yapılmıştır.

$$c_i = \frac{\sum_{I \in R_i} I_c}{|R_i|}, p_i = \frac{\sum_{I \in R_i} I_p}{|R_i|} \quad (3.1)$$

denkleminde c_i ve p_i sırasıyla süperpiksel bölgelerin içinde piksellerin renk ve uzaysal konumlarının ortalamasını; I_c I pikselindeki üç boyutlu CIELAB renk uzayı vektörünü; I_p I pikselindeki iki boyutlu XY konum vektörünü; $|R_i|$ ise süperpiksel bölgesi içerisindeki piksel sayısını ifade etmektedir.

3.2.2. Renk Kontrastı

Kontrast ölçütü, i süperpiksel bölgesindeki bilginin diğer j süperpiksel bölgelerinin bilgileriyle karşılaştırılması sonucu hesaplanır.

$$d_{ij}^{Bha} = \sqrt{1 - \sum_{u=1}^{4096} \sqrt{Hist_i^u Hist_j^u}} \quad (3.2)$$

denkleminde i ve j süperpiksel bölgeleri arasındaki Bhattacharyya uzaklığı hesaplanır. Bhattacharyya uzaklığı, iki bölge arasındaki renk histogram benzerliğini ölçer. Histogram karşılaştırmasında Bhattacharyya uzaklığı istatistiksel bir yaklaşım izlediğinden Öklid uzaklığına göre daha iyi sonuç verir. Renk histogramı hesabı için

CIELAB renk uzayı kullanılmıştır ve renk uzayının her bir bileşeni (L, a, b) 16 bölmeye ayrılmıştır. Her bölüt alanı bünyesinde toplamda $16 \times 16 \times 16 = 4096$ bölme bulundurmaktadır. Denklemdeki $Hist_i^u$ ve $Hist_j^u$ i ve j süperpiksel bölgelerinin bölgelerdeki bölme veya piksel sayısına göre normalleştirilmiş histogram değerlerini ifade etmektedir. Denklem sonucu sıfıra ne kadar yakın olursa bölgeler arasındaki benzerlik o kadar fazla olur.

$$w_{ij}^C = \|c_i - c_j\|_2^2 d_{ij}^{Bha} \quad (3.3)$$

$$w_{ij}^P = \exp\left(-\frac{1}{2\sigma_p^2} \|p_i - p_j\|_2^2\right) \quad (3.4)$$

$$Sal_i^{Cont} = \sum_{j=0, i \neq j} w_j w_{ij}^C w_{ij}^P \quad (3.5)$$

denkleminde w_{ij}^C i ve j süperpiksel bölgeleri arasındaki renk benzerliğini, c_i, c_j bölgelerin ortalama renk bilgisini, denklem 3.2'deki d_{ij}^{Bha} bölgeler arasındaki renk histogram benzerliğini, w_{ij}^P konumsal kısıt için kullanılan Gauss ağırlığını, p_i, p_j bölgelerin ortalama pozisyon bilgisini, w_j karşılaştırma yapılan bölgedeki toplam piksele sayısını, $\|\cdot\|$ ise L_2 normundaki Öklid uzaklığını ifade etmektedir.

w_j değeri büyük süperpiksel bölgelerinin daha çok kontrast farkı oluşturmasını sağlar. Süperpiksel yönteminde bölge büyüklükleri genellikle birbirlerine yakın olsa da sınır ve kenar kısımların kontrast geçişleri, karmaşık dokuya sahip bölgelerde farklı boyutlarda süperpiksel yapılarının oluşmasına sebep olur.

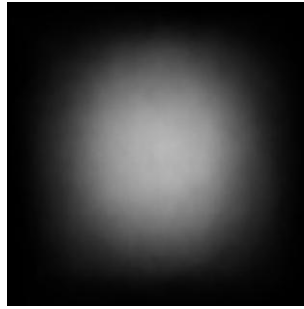
w_{ij}^P konumsal kısıtı yakın bölgelerin incelenilen bölgeye daha çok katkı yapmasını sağlar. $\sigma_p^2 = 0.4$ değeri deneysel olarak seçilmiştir ve p değerleri $[0,1]$ arasına normalleştirilmiştir. Büyük σ_p değerleri konumsal ağırlıklandırmanın etkisini azaltıp, uzak bölgelerinde incelenilen bölgeye katkı yapmasını sağlayacaktır.

$$d_i^{Cent} = \exp\left(-\frac{1}{2\sigma_p^2} \|p_i - c\|_2^2\right) \quad (3.6)$$

$$Sal_i^{Cont} = d_i^{Cent} \sum_{j=0, i \neq j} w_j w_{ij}^C w_{ij}^P \quad (3.7)$$

denkleminde d_i^{Cent} i süperpiksel bölgesinin olası belirginlik bölgesi merkezine olan uzaklığı için kullanılan Gauss ağırlığı konum kısıtı, c ise olası belirginlik bölgesinin merkezini ifade etmektedir.

Sal_i^{Cont} i süperpiksel bölgesinin belirginliğinin son değeri bu bölgenin belirginlik bölgesinin merkezine olan uzaklık kısıtı ile hesaplanmaktadır. Belirgin nesnenin resmin merkezinde yer alması gerektiği varsayımı üzerine önceki yöntemler [41,42] bu konum kısıtı hesaplamasını resmin merkez noktaları üzerinden yapmaktaydı. Şekil 3.2’de kullanılan veri tabanında bulunan doğruluk haritalarının ortalamasını aldığımızda belirginliğin merkeze yakın olduğunu görmekteyiz. Bu varsayım üzerine yapılan çalışmalar ve sonuçlar [27] bunu doğrulasa da resmin tam merkezinde yer almayan durumlarda mevcuttur. Böyle durumlarda bile daha tutarlı ve güvenilir sonuçlar elde etmek için resimdeki belirgin bölgenin konumunu yaklaşık olarak tahmin eden bir yöntem uygulanmıştır.

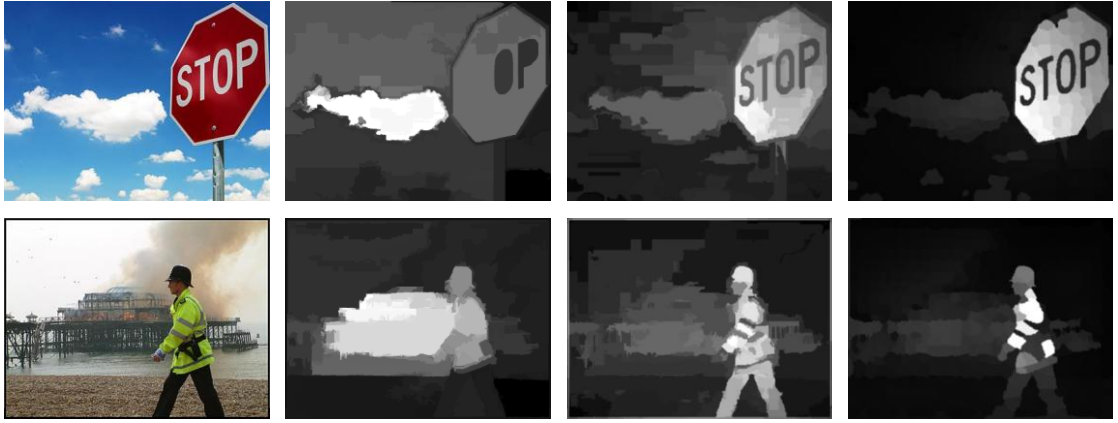


Şekil 3.2. Doğruluk haritalarının ortalaması

Harris köşe tespit yönteminin [59] hata verdiği durumlarda bile iyi sonuçlar veren onun geliştirilmiş modeli olan Shi-Tomasi köşe tespit yöntemiyle [60] belirgin bölge üzerinde önemli ilgi noktaları tespit edilerek olası merkez tespit edilir. Resmin

kenarlarına yakın tespit edilen noktalar gürültüye sebep olma ihtimaline karşın değerlendirilmemektedir.

Olası merkez noktası bilgisi yardımıyla tanımlanan Gauss konum kısıtı ile belirgin alan ile aynı kontrast bilgisine veya ondan daha fazla kontrast değerine sahip arka plan bölgelerinin parlaklık değerleri azaltılarak belirgin alana daha fazla vurgu yapılmaktadır. Şekil 3.3’de (2) numaralı yöntemde merkez tabanlı yaklaşım izlenirken, (3) numaralı yöntemde merkezilik kullanılmamıştır. (4) numaralı önerilen yöntemde ise muhtemel merkez noktası bilgisi kullanılmıştır. Resimlerde de görüleceği üzere önerilen yöntem ile merkezde bulunmayan belirginlik bölgeleri için iyi sonuçlar elde edilmiştir.



(1) orijinal resim

(2) CB [41]

(3) RC [38]

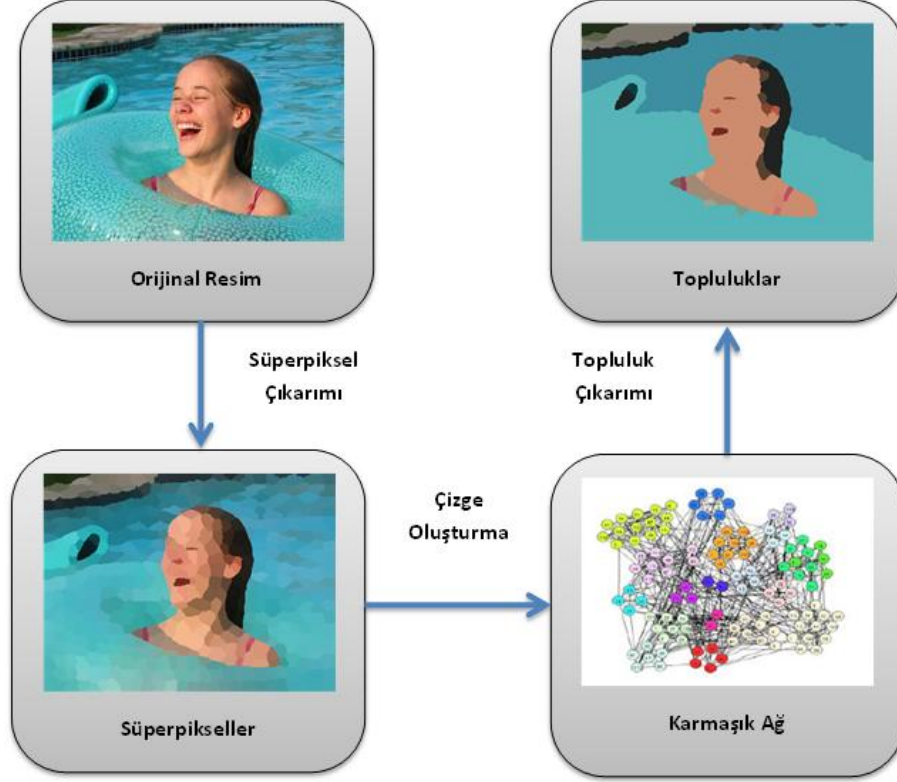
(4) önerilen

Şekil 3.3. Merkez noktalarına göre dikkat çekme haritaları

3.2.3. Karmaşık Ağ Kontrastı

Kontrast ölçütü sonrasında elde edilen dikkat çekme haritasında belirgin olması gereken alanlar aynı parlaklık değerini göstermeyebilir. Bunun sebebi aynı özellikleri sergileyen bölgelerin süperpiksel kısıtlamaları sonucu aşırı bölütlenmesinden dolayı kaynaklanmaktadır.

Önerilen yöntem karmaşık ağ kontrastı sonucu elde edilen dikkat çekme haritası ile önceki aşamada elde edilen sonucu kullanarak bölgeler için gereken yumuşatmayı ve benzer alanlarda aynı parlaklık değerlerinin sergilenmesini sağlar.



Şekil 3.4. Karmaşık ağ kontrastı yaklaşımı

İlk olarak süperpiksel bölgeleri kullanılarak bir çizge oluşturulur. Artık piksel tabanlı yaklaşım izlemediğimizden dolayı zaman karmaşıklığımız sadece süperpiksel bölge sayısı üzerinden olacaktır. Böylelikle, gürültülerden ve gereksiz bilgilerden yoksun daha işler yapıda olan anlamlı bir çizge oluşturulabilir. Çizge kümeleri arasındaki ağırlık fonksiyonu ise süperpiksel bölgelerinin renk ortalamaları arasındaki fark ile ifade edilmektedir. Bu ağırlık fonksiyonu bir küme diğer küme arasındaki benzerliği ortaya koyar. Şekil 3.4’de [56] karmaşık ağ kontrastı yaklaşımının akış şeması verilmiştir.

$$w_{ij}^c = \|c_i - c_j\|_2 \quad (3.8)$$

$$w_{ij}^p = \|p_i - p_j\|_2 \quad (3.9)$$

$$f_{ij}^{adj} = \begin{cases} 1, & w_{ij}^c < t_c \cap w_{ij}^p < t_p \\ 0, & w_{ij}^c \geq t_c \cup w_{ij}^p \geq t_p \end{cases} \quad (3.10)$$

denkleminde w_{ij}^c bölgeler arasındaki renk farkını, w_{ij}^p bölgeler arasındaki konum farkını, t_c renk fark eşiğini, t_p konum fark eşiğini, f_{ij}^{adj} bölgeler arasındaki komşuluğu, $\|\cdot\|$ ise L_2 normundaki Öklid uzaklığını ifade etmektedir.

t_c değeri ile renk farkı benzerliği belirli bir değerin altında olan bölgeler arasında bir ilişki ve incelemeye değer olduğu kontrol edilmektedir. t_p değeriyle ise bölgeler arasındaki uzaklığın ve birbirlerine olan katkılarının kontrol edilmesi sağlanır. Böylelikle renk benzerliği bulunan fakat aralarında fiziksel olarak komşuluk bulunmayan gibi durumlarda komşuluğun önüne geçilir. Yapılan deneyler sonucu $t_c = 0,02$ ve $t_p = 0,2$ değerleri yöntem için iyi sonuç verdiği gözlemlenmiştir. Burada renk ve uzaklık değerleri $[0,1]$ arasına normalleştirilmiştir.

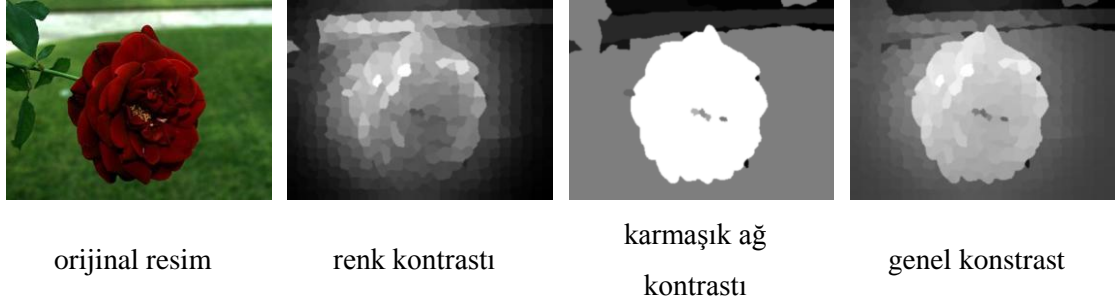
Çizge oluşturma aşamasından sonra çizgedeki komşulukları ve ağırlıkları kullanarak sayısı önceden bilinmeyen topluluğa ayırma işlemine geçilir. Topluluk çıkarım işleminde hızlı açgözlü algoritması yapılan ödünleşme sonucu tercih edilmiştir. Etiket yayılım yöntemi çok hızlı olmasına rağmen modülerliği göz önünde bulundurmadığı için seçilen yönteme göre daha az kesin sonuçlar vermektedir. Girvan ve Newman'ın bölücü algoritması ise zaman karmaşıklığının fazla olması sebebiyle tercih dışı kalmaktadır. Sonuç itibariyle zaman ve başarı ödünleşmesi neticesinde sonuçlarından da görülebileceği üzerine hızlı açgözlü algoritması topluluk çıkarım işleminde daha başarılıdır.

Aşağıdaki algoritma da çizge oluşturma ve topluluk çıkarımı aşamaları gösterilerek yöntemin açıklaması yapılmıştır ve sözde kodu verilmiştir.

Algoritma 3.1: Karmaşık Ağ Kontrastı Sözde Kodu

- 1: **Girdi:** t_c ve t_p kısıtları
 - 2: **Çıktı:** Çıkarım yapılmış topluluk
 - 3: G çizgesi oluştur.
 - 4: **foreach** i süperpiksel bölütü için **do**
 - 5: **foreach** j süperpiksel bölütü için **do**
 - 6: w_{ij}^c ve w_{ij}^p değerini hesapla.
 - 7: **if** $w_{ij}^c < t_c$ ve $w_{ij}^p < t_p$ **then**
 - 8: G çizgesine i ve j bölütleri için w_{ij}^c ağırlığında kenar ekle.
 - 9: G çizgesindeki her köşe için topluluk ata.
 - 10: **repeat**
 - 11: **while** 1==1 **do**
 - 12: Toplulukları birleştir ve Q modülerlik değerini hesapla.
 - 13: **if** Q yeterli maksimum değerine ulaşırsa **then**
 - 14: **break**
 - 15: **until** G çizgesi özgün bir topluluk ifade eder
-

Çizgedeki topluluklar tespit edildikten sonra bunlar arasındaki kontrast değişimlerine bakılarak belirginlik alanı tespiti yapılır. Bir önceki aşamada elde edilen sonuçlarda belirginlik alanlarındaki bütünlük probleminin üstesinden gelmek için yumuşatma yöntemine ihtiyaç duyulmuştur. Parçala ve birleştir tarzı yaklaşım ile tespit edilen topluluklarda süperpiksel bölütmesinde olan aşırı bölütleme problemine çok rastlanmaz. Böyle bir durumda daha bütün ve parlak dikkat çeken bölgeler elde edilebilir. Şekil 3.5’de renk kontrastı sonucuna karmaşık ağ kontrastının eklenmesiyle elde edilen yumuşatma sonucu olan genel kontrast sonucu gösterilmiştir.



Şekil 3.5. Renk ve karmaşık ağ kontrastı birleşimi

$$Sal_i^{CNCont} = d_i^{Cent} \sum_{j=0, i \neq j} w_j w_{ij}^C w_{ij}^P \quad (3.11)$$

denkleminde bütün değişkenler denklem 3.7'deki aynı değerleri ve kısıtları ifade etmektedir. w_j değeri büyük süperpiksel bölgelerinin daha çok kontrast farkı oluşturmasını sağlar. Daha büyük yapıda bölütleme bölgeleri olduğundan bu değişkenin sağlamış olduğu etkiyi bu ölçütte daha iyi görebiliriz. Topluluk çıkarımı sonucunda elde edilen bölgeler, süperpiksel bölgelerinde karşılık geldiği indeks konumlarına tekrar hesaplanarak aradaki farklı bölge sayısı problemi giderilmiştir.

3.2.4. Dağılım

Önceden bahsedildiği üzere dağılımda, bölgenin sahip olduğu renk bilgisini resmin geri kalanında ne kadar sıklıkla seyrettiği tespit edilmeye çalışılır. Konumsal olarak düşük dağılım yani varyans gösteren bölgeler daha belirgin ve bütün yapıdadır. Bu sebeple daha çok dağınık yapı sergileyen ve yüksek varyans gösteren arka plan bölgeleri daha az parlak olarak resmedilmedir.

$$\mu_i = \sum_{j=0, i \neq j} w_{ij}^C p_j \quad (3.12)$$

$$w_{ij}^P = \|\mu_i - p_j\|_2^2 \quad (3.13)$$

$$N^C = \sum_{j=0, i \neq j} \exp\left(-\frac{1}{2\sigma_c^2} \|c_i - c_j\|_2^2 d_{ij}^{Bha}\right) \quad (3.14)$$

$$w_{ij}^C = \frac{1}{N^C} \exp\left(-\frac{1}{2\sigma_c^2} \|c_i - c_j\|_2^2 d_{ij}^{Bha}\right) \quad (3.15)$$

$$Sal_i^{Dist} = \sum_{j=0, i \neq j} w_{ij}^C w_{ij}^P \quad (3.16)$$

denkleminde w_{ij}^C i ve j süperpiksel bölgeleri arasındaki renk benzerliğini, μ_i işlem yapılan i bölgesinin sahip olduğu renk bilgisinin karşılaştırılan diğer bölgelerle olan ağırlıklı konum ortalaması, c_i, c_j bölgelerin ortalama renk bilgisini, N^C normleştirme değişkeni, denklem 3.2'deki d_{ij}^{Bha} bölgeler arasındaki renk histogram benzerliğini, w_{ij}^P renk bilgisinin ortama dağılımını, p_j j bölgesinin ortalama pozisyon bilgisini, $\|\cdot\|$ ise L_2 normundaki Öklid uzaklığını ifade etmektedir. N^C normleştirme değişkeni w_{ij}^C toplamının 1'e eşit olmasını sağlar. $\sigma_c^2 = 0$ olduğu zaman $w_{ij}^C = \frac{1}{k}$ olur. w_{ij}^C kısıtlı bölgeler arasındaki renk benzerliği dağılımının kontrol edilmesini sağlar. $\sigma_c^2 = 400$ değeri deneysel olarak seçilmiştir ve p değerleri [0,1] arasına normleştirilmiştir. Denklem resme dağılmış ve yüksek varyans gösteren renk faktörü bölgelerini hesaplayıp bunların belirginlik bölgesinde yer almamasını sağlar. Düşük varyanslı bölgeler daha derli toplu bir yapı sergiledikleri için belirginlik bölgesinde yar alırlar.

3.2.5. Dikkat Çekme Haritası Oluşturma

Dikkat çekme haritasının son halini hesaplamak için önceki adımlardaki ölçütlerin sonucu olan dikkat çekme haritaları kullanılır. Kontrast ve karmaşık ağ kontrastı ölçütleri için lineer, bunların sonucu ile dağılım ölçütlerinin bir araya getirilmesi için lineer olmayan tanımlama yapılmıştır. İlkinde belirgin alanların daha iyi ifadesi için lineer, sonrasında arka planların bastırılabilmesi için lineer olmayan ifade tanımlama kullanılmıştır. Tüm Sal_i^{Cont} , Sal_i^{CNCont} ve Sal_i^{Dist} ölçütleri kendi içerisinde [0,1] arasına normleştirilir ve bütün bir sonuç aşağıdaki tanım kullanılır:

$$\overline{Sal}_i^{Cont} = \alpha Sal_i^{Cont} + \beta Sal_i^{CNCont} \quad (3.17)$$

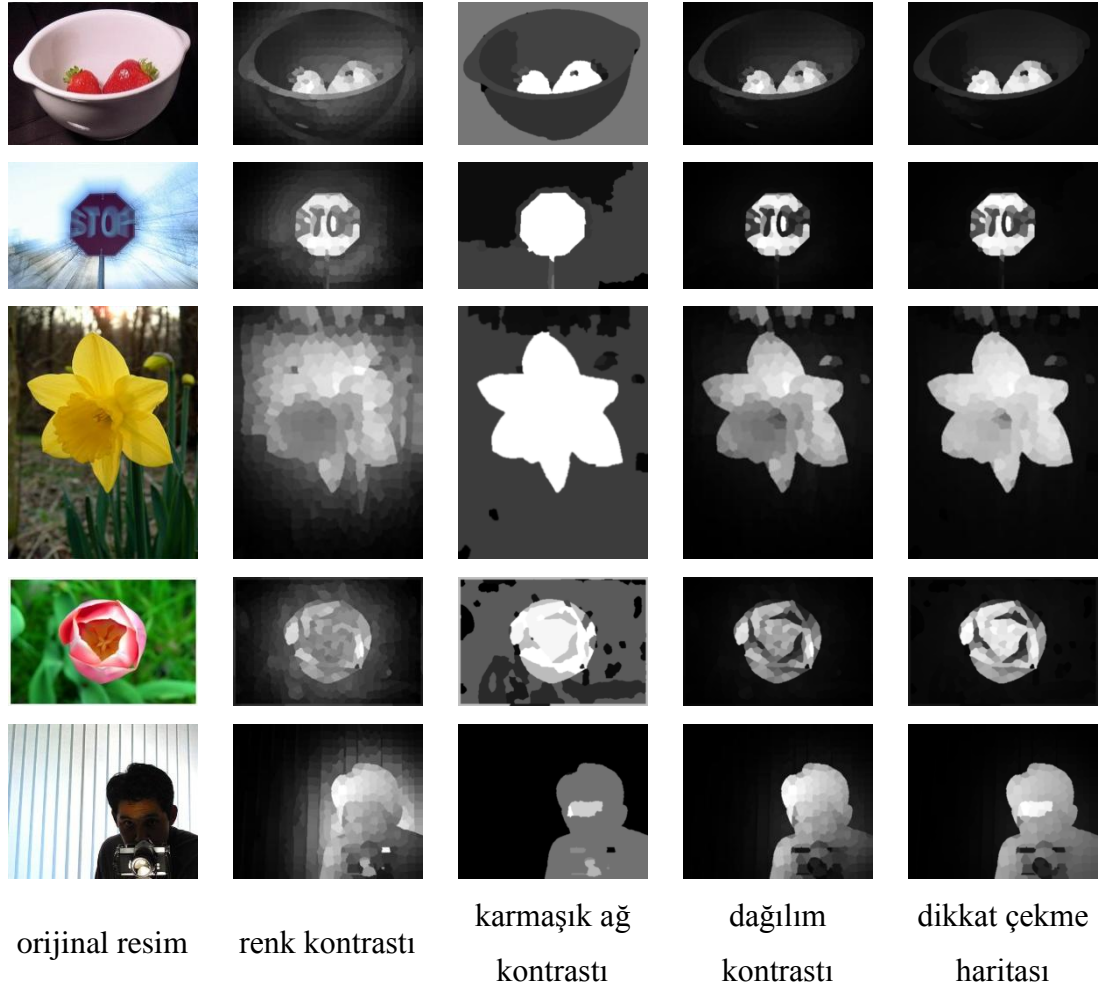
$$Sal_i = \overline{Sal}_i^{Cont} \exp(-\gamma Sal_i^{Dist}) \quad (3.18)$$

denklemden lineer olmayan tanımda üstel ifade kullanılması sebebi, yüksek dağılım göstermeyen kısımların düşük belirginlik göstermesi içindir. Bunu tanımlamak için maksimum değerinden de çıkarabilirdik ama aradaki ani değişimlerin daha iyi belirtilmesi için üstel fonksiyon kullanılmıştır. Buradaki $\alpha = 0,5$, $\beta = 0,5$ ve $\gamma = 2$ değerlerinin iyi sonuçlar verdiği deneysel olarak tespit edilmiştir.

Aşağıdaki algoritma da önerilen yöntemin aşamaları gösterilerek yöntemin açıklaması yapılmıştır ve sözde kodu verilmiştir.

Algoritma 3.2: Önerilen Yöntemin Sözde Kodu

- 1: **Girdi:** Süperpiksel sayısı, t_c ve t_p kısıtları
 - 2: **Çıktı:** Dikkat çekme haritası
 - 3: Algoritma 2.1 kullanılarak resmi bölütlemesi yapılır. Algoritmada orijinal resim kullanılır.
 - 4: Algoritma 3.1 kullanılarak topluluklar tespit edilir. Algoritmada hesaplamalar süperpiksel bölgeleri üzerinden yapılır.
 - 5: **foreach** i süperpiksel bölütü için **do**
 - 6: **foreach** j süperpiksel bölütü için **do**
 - 7: Denklemleri 3.7 kullanarak Sal_i^{Cont} değerini hesapla.
 - 8: **foreach** i topluluk bölütü için **do**
 - 9: **foreach** j topluluk bölütü için **do**
 - 10: Denklemleri 3.11 kullanarak Sal_i^{CNCont} değerini hesapla.
 - 11: **foreach** i süperpiksel bölütü için **do**
 - 12: **foreach** j süperpiksel bölütü için **do**
 - 13: Denklemleri 3.16 kullanarak Sal_i^{Dist} değerini hesapla.
 - 14: **foreach** i süperpiksel bölütü için **do**
 - 15: Denklemleri 3.18 kullanarak Sal_i değerini hesapla.
 - 16: Sal_i değerlerini kullanarak dikkat çekme haritasını oluştur.
-



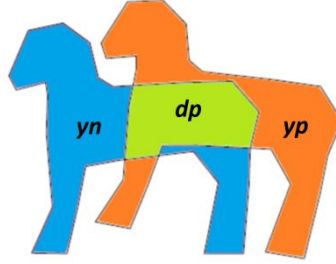
Şekil 3.6. Yöntemin aşamalarından başka sonuçlar

Şekil 3.6'da yukarıda önerilen yöntem için bahsettiğimiz algoritmanın adımlarından çeşitli çıktılar sunulmuştur.

4. DENEYLER

Önerilen yöntem 1000 resim ve doğruluk haritalarını içeren [6] veri tabanı kullanılarak literatürdeki popüler yöntemlerle karşılaştırılmıştır. Bazı yöntemlerin dikkat çekme haritaları önceden sağlanmıştı. [38] Kullanılan veri tabanı içerdiği doğruluk haritalarından dolayı nerdeyse tüm çalışmalarda yöntemlerin karşılaştırılması için kullanılan temel bir kaynaktır.

Performans değerlendirmesi diğer yöntemlerdeki [6,35,38,40-42] gibi hassasiyet (precision) ve anma (recall) oranları kullanılacaktır. Precision oranı dikkat çekme haritasında piksellerin ne kadarının doğru işaretlendiğini, anma oranı ise doğruluk haritasından ne kadarlık kayıp olduğunu gösterir. Şekil 4.1'de [45] mavi ile gösterilen kısım doğruluk haritasını, turuncu ile gösterilen kısım dikkat çekme haritasını temsil eden şekillerdir. Bu iki haritanın kesişimi piksellerin doğru tespit edildiğini (doğru pozitif - dp)(true positives - tp), turuncu kısım yanlış tespit edildiğini (yanlış pozitif - yp)(false positives - fp) ve mavi kısımda tespit edilememesinden dolayı olan hatayı (yanlış negatif - yn)(false negatives - fn) göstermektedir.



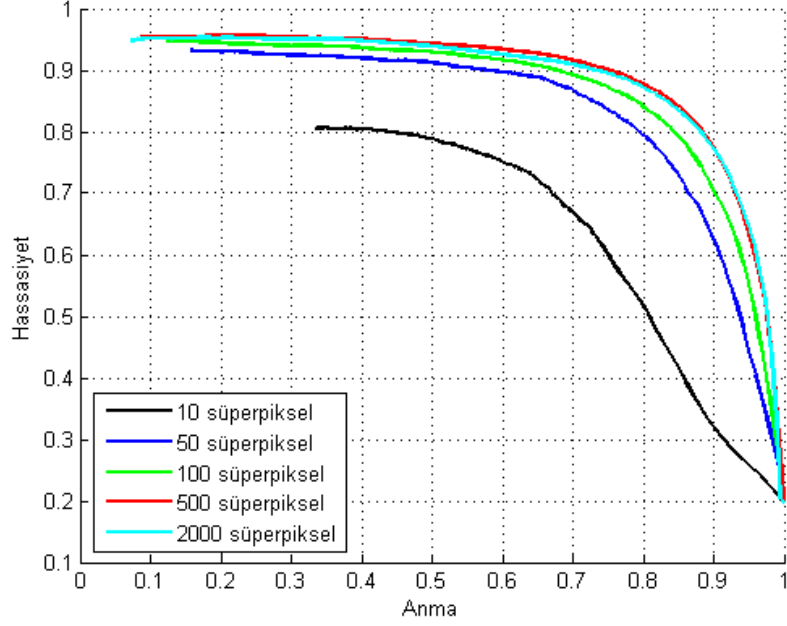
Şekil 4.1. Doğruluk ve dikkat çekme haritalarının karşılaştırımı

$$hassasiyet = \frac{\sum_{(x,y)} S(x,y)G(x,y)}{\sum_{(x,y)} S(x,y)} = \frac{dp}{dp + yp} \quad (4.1)$$

$$anma = \frac{\sum_{(x,y)} S(x,y)G(x,y)}{\sum_{(x,y)} G(x,y)} = \frac{dp}{dp + yn} \quad (4.2)$$

denkleminde S siyah-beyazdan oluşan dikkat çekme haritasındaki pikselleri, G ise doğruluk haritasındaki pikselleri tanımlamaktadır. Belirginliğin olduğu pikseller 1,

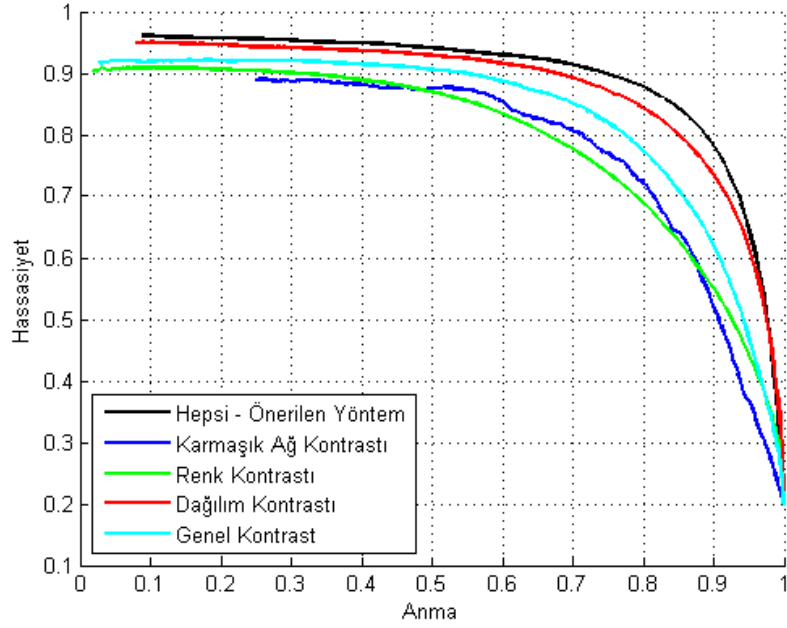
arka plan pikselleri ise 0 ile etiketlenmiştir. Yöntem sonucu elde edilen dikkat çekme haritası [0,255] değerleri arasındaki tüm eşik değerleri için siyah-beyaz (ikili) resme çevrilir. Hassasiyet ve anma değerlerinin 1000 resim için ortalaması alınır.



Şekil 4.2. Süperpiksel sayısına göre yöntemin başarı oranı

Şekil 4.2'de resmi bölütlere ayırırken neden 500 seçtiğimizin sonuçları yer almaktadır. 500 bölütün altındaki değerlerin anma oranları biraz daha hızlı düşmektedir. 500'ün üzerindeki değerlerde ise çok yakın sonuçlar elde edildiğinden ve çalışma süresinin artmasından dolayı yapılan ödünleşme sonucu bu sayı ideal olarak belirlenmiştir.

Şekil 4.3'de yöntemin basamaklarının kendi içerisindeki başarıları gösterilmiştir. Renk kontrastı ve karmaşık ağ kontrastının birleştirilmesiyle oluşturulan genel kontrast yöntemin tutarlılığını arttırmıştır. Karmaşık ağ kontrastı ile desteklenen renk kontrastı için uygulanan yumuşatma işleminin başarılı olduğu görülmüştür.



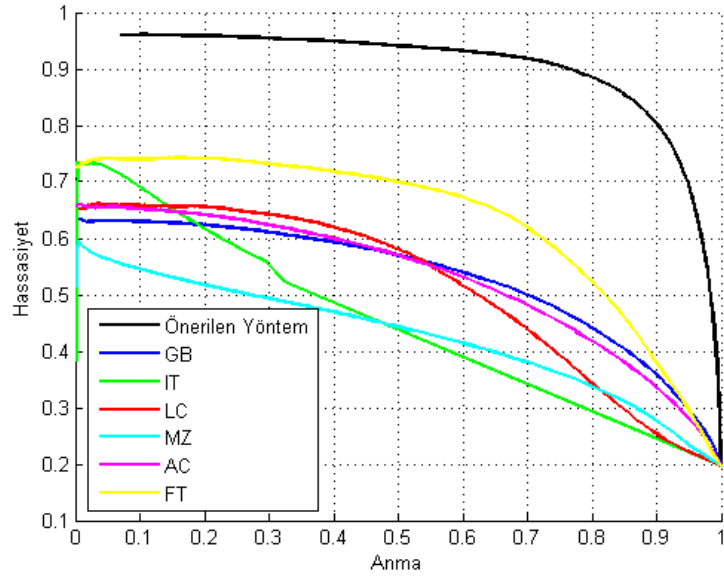
Şekil 4.3. Önerilen yöntemdeki adımların karşılaştırılması

Şekil 4.4 ve 4.5’de önerilen yöntemim diğer başarılı ve popüler yöntemlerle olan karşılaştırma sonuçları verilmiştir. Maksimum %96’lık hassasiyet değeri ile diğer yöntemlerden daha başarılı olduğu gösterilmiştir. Ayrıca Tablo 4.1’de verilen çalışma sürelerine bakıldığında, elde edilen sonucun bu çalışma süresi ile elde edilmiş olması önerilen yöntemin performans açısından başarılı olduğunu göstergesidir. Ancak, bazı durumlarda yeterli yumuşatmanın sağlanamaması yani belirginlik bölgelerinin bütün olarak daha parlak olmamasından dolayı belli bir süre sonra anma değerine bağlı olarak hassasiyet değeri düşmeye başlamaktadır.

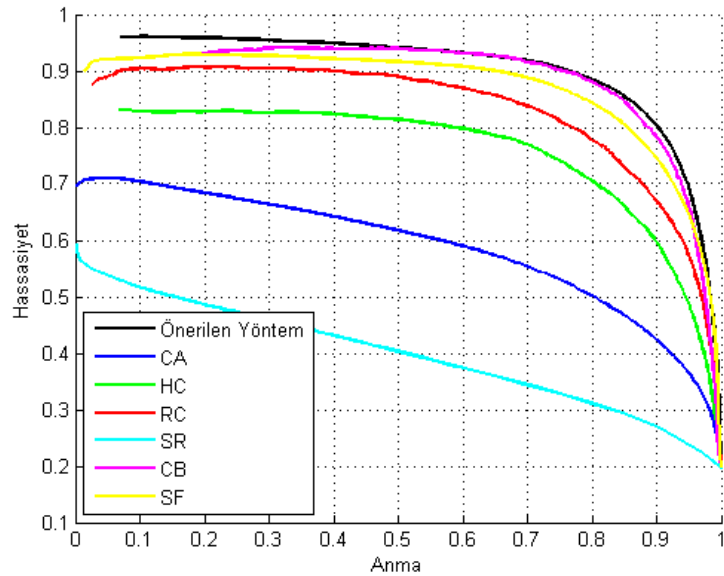
Tablo 4.1. Yöntemlerin çalışma süreleri

Yöntemler	CA[35]	FT[6]	HC[38]	RC[38]	SF[40]	CB[41]	Yöntem
Süreler(s)	51,1	0,013	0,012	0,145	0,158	5,57	0,173
Kodlama	Matlab	C++	C++	C++	C++	Matlab	C++
Yöntemler	GB[29]	SR[31]	AC[11]	LC[33]	IT[7]	MZ[32]	
Süreler(s)	1,578	0,059	0,01	0,013	0,12	0,063	
Kodlama	Matlab	Matlab	Matlab	C++	Matlab	C++	

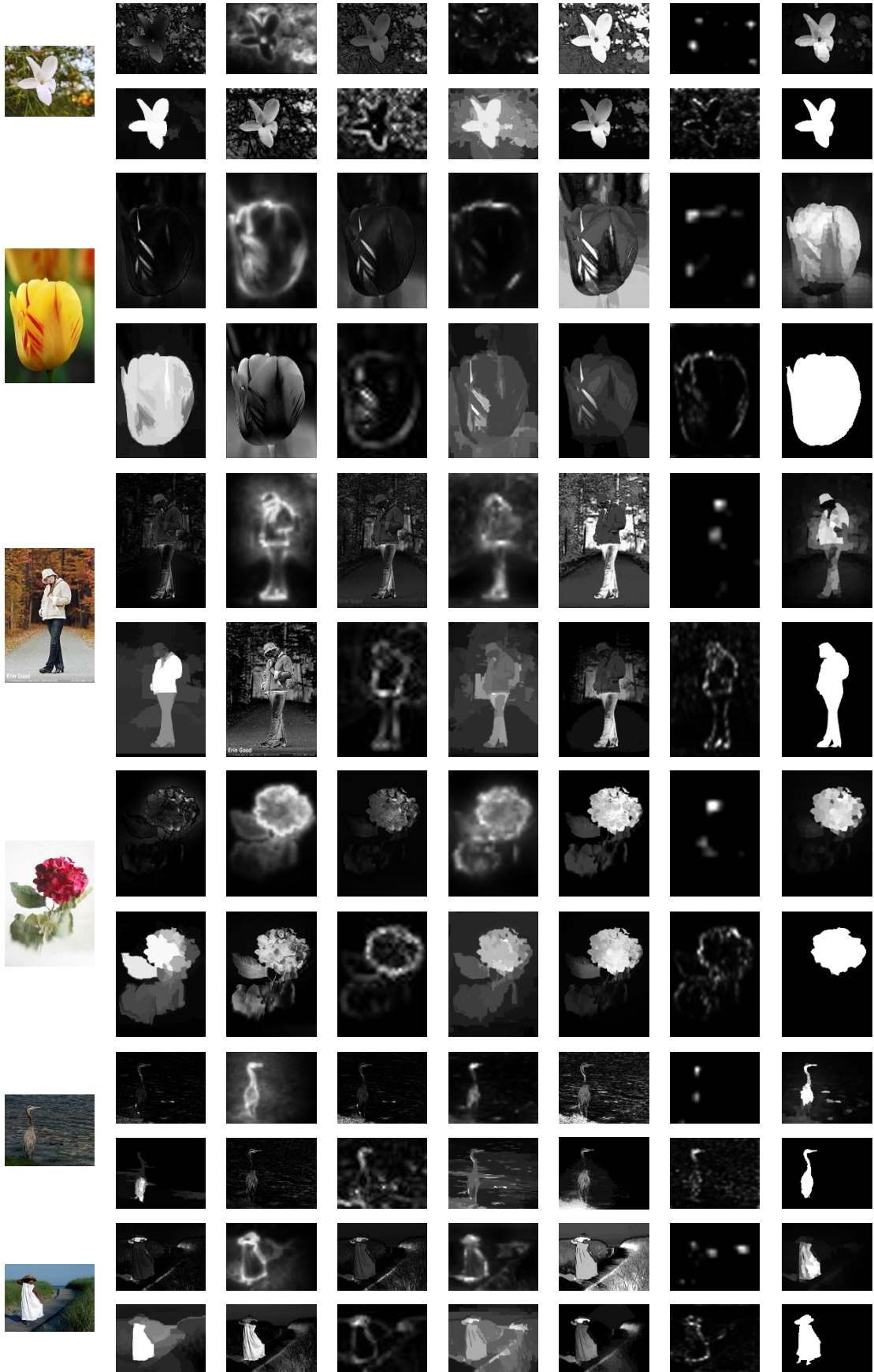
Uygulama C++ programlama dillerinde, Visual Studio 2012 ortamında Windows 8 platformu üzerinde geliştirilmiştir. Tüm testler Intel Core i5-2.4 GHz işlemcisine ve 16GB RAM kapasitesine sahip bilgisayar üzerinde yapılmıştır. Önerilen yöntemin çalışma zamanının %60'lık kısmını süperpiksel bölütlemesi almaktadır ve optimizasyon yapılmamış sonuçlardır. Resimde yöntemlerle ilgili çeşitli sonuçlar verilmiştir. Şekil 4.6'da problemle ilgili çeşitli çıktılar verilmiştir.

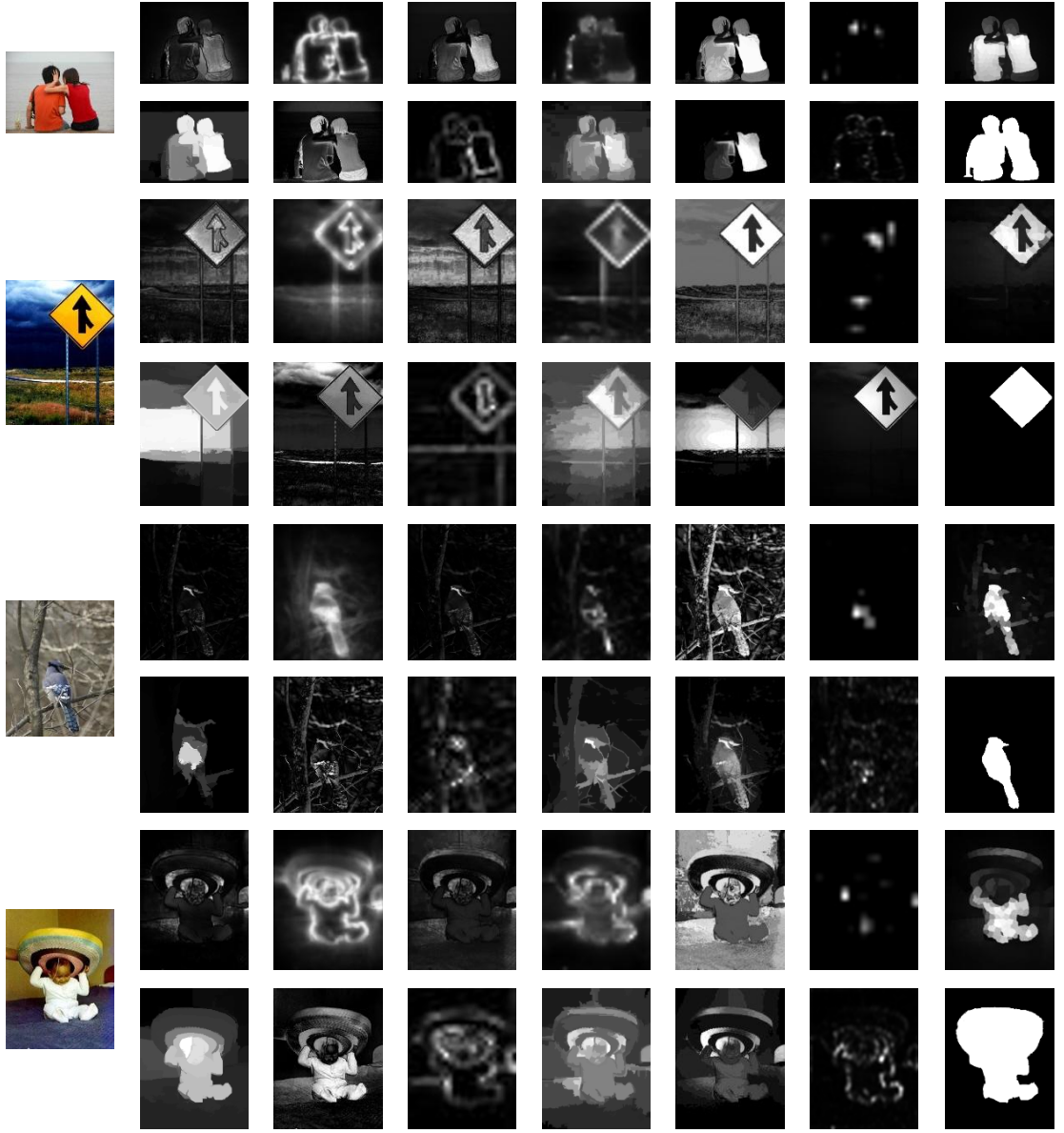


Şekil 4.4. Önerilen yöntemin önceki çalışmalarla karşılaştırılması



Şekil 4.5. Önerilen yöntemin önceki çalışmalarla karşılaştırılması





orijinal
resim

AC [11] CA [36] FT [6] GB [29] HC [38] IT [7]
CB [41] LC [33] MZ [32] RC [38] SF [40] SR [31]

önerilen
yöntem
doğruluk
haritası

Şekil 4.6. Popüler çalışmalardan ve önerilen yöntemden çıktılar

5. SONUÇ

Bilgisayarla görü alanındaki nesne tanıma, resim bölütleme, resim boyutlandırma, resimden çıkarım yapma, resim ve video sıkıştırma gibi birçok içerik tabanlı uygulamaların kullanım alanları giderek yaygınlaşmaktadır. Bu uygulamalarda girdi olarak kullanılan dikkat çekme haritalarının yöntemlerin başarılı sonuçlar almasında büyük önemi vardır. Bu amaçla belirgin alanların tespiti konusunda son yıllarda yapılan çalışmaların sayısı oldukça fazladır. Bu yapılan çalışmaların çoğundaki temel amaç, algoritma sonunda elde edilen dikkat çekme haritasını en iyi ve en hızlı şekilde elde etmektir. Aynı şekilde, bu alandaki sonuçların yukarıda bahsedilen bilgisayarla görü alanındaki birçok uygulamada temel kaynak teşkil etmesi ve öneminin artmış olması, yapılan çalışmalardan daha hızlı, tutarlı ve iyi sonuçlar elde etmek bu tezde yapılan çalışmanın amacını oluşturmaktadır. Önerilen yöntemin benzer çalışmaların sonuçlarını ne kadar geliştirdiği deneyler kısmında gösterilmiştir. Öncelikle, birçok yöntemin izlediği piksel tabanlı yaklaşımdan farklı olarak bölgesel tabanlı yaklaşım tercih edilmiştir. Yapılan çalışmalarda zaman ve sonuçlar açısından bölgesel tabanlı yaklaşımların çok daha başarılı oldukları deneylerle ortaya konulmuştur. Kontrast elde etmek için önerilen yaklaşımlara karmaşık ağ kullanılarak elde edilen kontrast sonuçları eklenerek daha tutarlı ve verimli çıktılar elde edilmiştir. Kontrast çıktılarını belirginliğin olabileceği bölgelerde daha çok vurgulayabilmek için kullanılan merkez tabanlı kısıt yerine belirgin bölgenin bulanabileceği noktayı tahmin eden yöntem ile merkez çevresinde bulunmayan veri kümeleri için daha doğru sonuçlara varılmıştır.

Yapılan bu çalışmanın bir sonraki aşamalarında çalışma zamanı ve doğruluğu açısından daha da geliştirilmesi hedeflenmektedir. Çalışma zamanı açısından problemin paralel programlaya elverişli olması açısından CPU mimarisine yerine GPU üzerinde geliştirme yapılması planlanmaktadır. Ayrıca belirgin bölgenin tahmin doğruluğunu arttırmak, karmaşık ağ desteğinin genişletilmesi ve bu alandaki diğer yaklaşımlarında uygulanması düşünülmektedir.

KAYNAKLAR

- [1] Palmer, Stephen E. Vision science: Photons to phenomenology. The MIT press, 1999.
- [2] Desimone, Robert, and John Duncan. "Neural mechanisms of selective visual attention." Annual review of neuroscience 18.1 (1995): 193-222.
- [3] Mannan, Sabira K., Christopher Kennard, and Masud Husain. "The role of visual salience in directing eye movements in visual object agnosia." Current biology 19.6 (2009): R247-R248
- [4] Teuber, Hans-Lukas. "Physiological psychology." Annual review of psychology 6.1 (1955): 267-296.
- [5] Wolfe, Jeremy M., and Todd S. Horowitz. "What attributes guide the deployment of visual attention and how do they do it?" Nature Reviews Neuroscience 5.6 (2004): 495-501.
- [6] Achanta, Radhakrishna, et al. "Frequency-tuned salient region detection." Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009.
- [7] Itti, Laurent, Christof Koch, and Ernst Niebur. "A model of saliency-based visual attention for rapid scene analysis." Pattern Analysis and Machine Intelligence, IEEE Transactions on 20.11 (1998): 1254-1259.
- [8] Rutishauser, Ueli, et al. "Is bottom-up attention useful for object recognition?." Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on. Vol. 2. IEEE, 2004.
- [9] Han, Junwei, et al. "Unsupervised extraction of visual attention objects in color images." Circuits and Systems for Video Technology, IEEE Transactions on 16.1 (2006): 141-145.
- [10] Ko, Byoung Chul, and Jae-Yeal Nam. "Object-of-interest image segmentation based on human attention and semantic region clustering." JOSA A 23.10 (2006): 2462-2470.

- [11] Achanta, Radhakrishna, et al. "Salient region detection and segmentation." *Computer Vision Systems*. Springer Berlin Heidelberg, 2008. 66-75.
- [12] Chen, Tao, et al. "Sketch2Photo: internet image montage." *ACM Transactions on Graphics (TOG)*. Vol. 28. No. 5. ACM, 2009.
- [13] Fu, Hong, Zheru Chi, and Dagan Feng. "Attention-driven image interpretation with application to image retrieval." *Pattern Recognition* 39.9 (2006): 1604-1621.
- [14] Christopoulos, Charilaos, Athanassios Skodras, and Touradj Ebrahimi. "The JPEG2000 still image coding system: an overview." *Consumer Electronics, IEEE Transactions on* 46.4 (2000): 1103-1127.
- [15] Guo, Chenlei, and Liming Zhang. "A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression." *Image Processing, IEEE Transactions on* 19.1 (2010): 185-198.
- [16] Wang, Yu-Shuen, et al. "Optimized scale-and-stretch for image resizing." *ACM Transactions on Graphics (TOG)*. Vol. 27. No. 5. ACM, 2008.
- [17] Zhang, Guo-Xin, et al. "A Shape-Preserving Approach to Image Resizing." *Computer Graphics Forum*. Vol. 28. No. 7. Blackwell Publishing Ltd, 2009.
- [18] Wu, Huisi, et al. "Resizing by symmetry-summarization." *ACM Transactions on Graphics (TOG)* 29.6 (2010): 159.
- [19] Ding, Meng, and Ruo-Feng Tong. "Content-aware copying and pasting in images." *The Visual Computer* 26.6-8 (2010): 721-729.
- [20] Avidan, Shai, and Ariel Shamir. "Seam carving for content-aware image resizing." *ACM Transactions on graphics (TOG)*. Vol. 26. No. 3. ACM, 2007.
- [21] Einhäuser, Wolfgang, and Peter König. "Does luminance-contrast contribute to a saliency map for overt visual attention?" *European Journal of Neuroscience* 17.5 (2003): 1089-1097.

- [22] Parkhurst, Derrick, Klinton Law, and Ernst Niebur. "Modeling the role of salience in the allocation of overt visual attention." *Vision research* 42.1 (2002): 107-123.
- [23] Reinagel, Pamela, and Anthony M. Zador. "Natural scene statistics at the centre of gaze." *Network: Computation in Neural Systems* 10.4 (1999): 341-350.
- [24] Wei Zhang; Wu, Q. M J; Guanghui Wang; HaiBing Yin, "An Adaptive Computational Model for Salient Object Detection," *Multimedia, IEEE Transactions on* , vol.12, no.4, pp.300,316, June 2010.
- [25] Fergus, Robert, Pietro Perona, and Andrew Zisserman. "Object class recognition by unsupervised scale-invariant learning." *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on. Vol. 2. IEEE, 2003.*
- [26] Parikh, Devi, C. Lawrence Zitnick, and Tsuhan Chen. "Determining patch saliency using low-level context." *Computer Vision–ECCV 2008. Springer Berlin Heidelberg, 2008. 446-459.*
- [27] Judd, Tilke, et al. "Learning to predict where humans look." *Computer Vision, 2009 IEEE 12th international conference on. IEEE, 2009.*
- [28] Koch, Christof, and Shimon Ullman. "Shifts in selective visual attention: towards the underlying neural circuitry." *Matters of Intelligence. Springer Netherlands, 1987. 115-141.*
- [29] Harel, Jonathan, Christof Koch, and Pietro Perona. "Graph-based visual saliency." *Advances in neural information processing systems. 2006.*
- [30] Guo, Chenlei, Qi Ma, and Liming Zhang. "Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform." *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008.*
- [31] Hou, Xiaodi, and Liqing Zhang. "Saliency detection: A spectral residual approach." *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on. IEEE, 2007.*

- [32] Ma, Yu-Fei, and Hong-Jiang Zhang. "Contrast-based image attention analysis by using fuzzy growing." Proceedings of the eleventh ACM international conference on Multimedia. ACM, 2003.
- [33] Zhai, Yun, and Mubarak Shah. "Visual attention detection in video sequences using spatiotemporal cues." Proceedings of the 14th annual ACM international conference on Multimedia. ACM, 2006.
- [34] Itti, Laurent, and Pierre F. Baldi. "Bayesian surprise attracts human attention." Advances in neural information processing systems. 2005.
- [35] Liu, Tie, et al. "Learning to detect a salient object." Pattern Analysis and Machine Intelligence, IEEE Transactions on 33.2 (2011): 353-367.
- [36] Goferman, Stas, Lihi Zelnik-Manor, and Ayellet Tal. "Context-aware saliency detection." Pattern Analysis and Machine Intelligence, IEEE Transactions on 34.10 (2012): 1915-1926.
- [37] Wang, Meng, et al. "Image saliency: From intrinsic to extrinsic context." Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. IEEE, 2011.
- [38] Cheng, Ming-Ming, et al. "Global contrast based salient region detection." Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. IEEE, 2011.
- [39] Ren, Zhixiang, et al. "Improved saliency detection based on superpixel clustering and saliency propagation." Proceedings of the international conference on Multimedia. ACM, 2010.
- [40] Perazzi, Federico, et al. "Saliency filters: Contrast based filtering for salient region detection." Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE, 2012.
- [41] Jiang, Huaizu, et al. "Automatic salient object segmentation based on context and shape prior." BMVC. Vol. 3. No. 4. 2011.

- [42] Fu, Keren, et al. "Salient object detection via color contrast and color distribution." *Computer Vision–ACCV 2012*. Springer Berlin Heidelberg, 2013. 111-122.
- [43] Xue, Yinzhu, Zhi Liu, and Ran Shi. "Saliency detection using multiple region-based features." *Optical Engineering* 50.5 (2011): 057008-057008.
- [44] Cigla, C., and A. A. Alatan. "Efficient super pixel extraction for image segmentation." *Signal Processing and Communications Applications Conference (SIU), 2012 20th*. IEEE, 2012.
- [45] Achanta, Radhakrishna, et al. "Slic superpixels." *École Polytechnique Fédéral de Laussanne (EPFL), Tech. Rep 149300* (2010).
- [46] Zhou, Yang, Hong Cheng, and Jeffrey Xu Yu. "Graph clustering based on structural/attribute similarities." *Proceedings of the VLDB Endowment* 2.1 (2009): 718-729.
- [47] Aggarwal, Charu C. *An introduction to social network data analytics*. Springer US, 2011.
- [48] Newman, Mark EJ, and Michelle Girvan. "Finding and evaluating community structure in networks." *Physical review E* 69.2 (2004): 026113.
- [49] Newman, Mark EJ. "Fast algorithm for detecting community structure in networks." *Physical review E* 69.6 (2004): 066133.
- [50] Shi, Jianbo, and Jitendra Malik. "Normalized cuts and image segmentation." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 22.8 (2000): 888-905.
- [51] Meila, Marina, and Jianbo Shi. "A random walks view of spectral segmentation." (2001).
- [52] Kannan, Ravi, Santosh Vempala, and Adrian Vetta. "On clusterings: Good, bad and spectral." *Journal of the ACM (JACM)* 51.3 (2004): 497-515.
- [53] Dhillon, Inderjit S., Yuqiang Guan, and Brian Kulis. "Weighted graph cuts without eigenvectors a multilevel approach." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 29.11 (2007): 1944-1957.

- [54] Brandes, Ulrik, et al. "On finding graph clusterings with maximum modularity." *Graph-Theoretic Concepts in Computer Science*. Springer Berlin Heidelberg, 2007.
- [55] Clauset, Aaron, Mark EJ Newman, and Cristopher Moore. "Finding community structure in very large networks." *Physical review E* 70.6 (2004): 066111.
- [56] Cuadros, Oscar, et al. "Segmentation of Large Images with Complex Networks." *Graphics, Patterns and Images (SIBGRAPI)*, 2012 25th SIBGRAPI Conference on. IEEE, 2012.
- [57] Raghavan, Usha Nandini, Réka Albert, and Soundar Kumara. "Near linear time algorithm to detect community structures in large-scale networks." *Physical Review E* 76.3 (2007): 036106.
- [58] Moravec, Hans P. *Obstacle avoidance and navigation in the real world by a seeing robot rover*. No. STAN-CS-80-813. STANFORD UNIV CA DEPT OF COMPUTER SCIENCE, 1980.
- [59] Chris Harris, Mike Stephens, *A Combined Corner and Edge Detector*, 4th Alvey Vision Conference, 1988, pp147-151.
- [60] Shi and C. Tomasi. *Good Features to Track*. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 593-600, June 1994.

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, Adı : AKSAÇ, Alper
Uyruğu : T.C.
Doğum tarihi ve yeri : 01.06.1988 Salihli
Medeni hali : Bekâr
Telefon : 0 (544) 674 61 13
E-posta : aaksac@etu.edu.tr

Eğitim

Derece	Eğitim Yeri	Mezuniyet Tarihi
Lisans	TOBB ETÜ Bilgisayar Mühendisliği	2011

İş Deneyimi

İş Deneyimi	Görev
2011-2013	TOBB ETÜ Ders ve Araştırma Asistanlığı
2010-2010	TOBB ETÜ Yazılım Programlama
2009-2009	TOBB ETÜ Yazılım Programlama
2008-2008	Ventura Yazılım Yazılım Programlama

Yabancı Dil

İngilizce

Almanca

Yayınlar

1. Uzun, Erkam, et al. "Kablosuz Algılayıcı Ağlarda Yasam Süresi Eniyilemesi ve Yerel Yönlendirme Ödünleşmesi Network Lifetime Maximization and Localized Routing Tradeoff in Wireless Sensor Networks."
2. Elhajj, Ahmed, et al. "Estimating the Importance of Terrorists in a Terror Network." Mining Social Networks and Security Informatics. Springer Netherlands, 2013. 267-283.
3. Aksaç, Alper, Erkam Uzun, and Tansel Özyer. "A real time traffic simulator utilizing an adaptive fuzzy inference mechanism by tuning fuzzy parameters." Applied Intelligence 36.3 (2012): 698-720.
4. Aksac, Alper, Orkun Ozturk, and Erdogan Dogdu. "A novel semantic web browser for user centric information retrieval: PERSON." Expert Systems with Applications 39.15 (2012): 12001-12013.
5. Aksaç, Alper, O. Ozturk, and T. Ozyer. "Real-time multi-objective hand posture/gesture recognition by using distance classifiers and finite state machine for virtual mouse operations." Electrical and Electronics Engineering (ELECO), 2011 7th International Conference on. IEEE, 2011.