

**GEODEZİK MESAFE TABANLI ETKİLEŞİMLİ RESİM BOLÜMLEMESİ**

**ORKUN ÖZTÜRK**

**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

Orkun ÖZTÜRK tarafından hazırlanan GEODEZİK MESAFE TABANLI ETKİLEŞİMLİ RESİM BOLÜMLEMESİ 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.

Orkun ÖZTÜRK

**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

## **ORKUN ÖZTÜRK**

### **GEODEZİK MESAFE TABANLI ETKİLEŞİMLİ RESİM BOLÜMLEMESİ**

#### **ÖZET**

Kullanıcı etkileşimli ön plan çıkarımının, arka plandan ayrılan nesnelerin başka bir arka plana taşınarak resim montajlarının yapılması ya da gereksiz arka plan bilgisinin elenmesi sonucu sadece istenilen bölgenin elde edilerek bu bölge üzerinde (medikal) görüntü işlemenin yapılması gibi çeşitli alanlarda kullanımı mevcuttur. Yaygın kullanımda, etkileşimli bölümlenme tanım olarak; kullanıcıların arka plandan soyutlamak istedikleri nesneye ve arka plana ait alanlarda yer alan bazı görüntü öğelerini ayrı ayrı işaretlemeleri yoluyla geriye kalan (kullanıcı tarafından işaretlenmemiş) piksellerin arka ya da ön plandan hangisine ait olduklarının tespit edilmesidir.

Bu tez çalışmasında, kullanıcı tarafından işaretlenen ön ve arka plan piksellerinden yola çıkarak, işaretlenmemiş piksellerin hangi bölgeye ait olduklarının tahmini üzerinde çalışılmış ve yeni bir metot geliştirilmiştir. Önerilen metotta ilk olarak işlenecek olan resim bir ön aşamadan geçirilerek birbirine yakın renk ve konumdaki pikseller gruplandırılmış (süper piksel) ve sınıflandırılması gereken piksel sayısı büyük oranda azaltılmıştır. İşaretlenmemiş (süper)piksellerin, klasik yaklaşımdan farklı olarak ön veya arka plana ait olma olasılıkları, işaretlenmiş (süper)piksellere olan konumsal yakınlıkları da gözetilerek hesaplanmıştır. Bu olasılık değerleri kullanılarak işaretli/siz pikseller arasındaki en kısa yol mesafeleri bulunarak belli bir mesafede yer alan piksellerin ön veya arka plana atanması gerçekleştirilmiştir. Bunun sonucunda atanamayan piksellerin kendilerine konum olarak yakın atanmış piksellere göre olasılık tahmini yapılmış ve bölümlenme işlemi tamamlanmıştır. Elde edilen sonuçlar bu tür işaretlemeyi kullanan metotlarla nicelik olarak herkesin kullanımına açık olan bir veri kümesi üzerinde karşılaştırılmıştır.

**Anahtar Kelimeler:** etkileşimli resim bölümlenmesi, ön plan/nesne çıkarımı, yarı denetimli bölümlenme, fazla bölümlenme, uzaysal değişken, olasılık yoğunluk tahmini, geodezik mesafe

**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**

**ORKUN ÖZTÜRK**

## **GEODESIC DISTANCE BASED INTERACTIVE IMAGE SEGMENTATION**

### **ABSTRACT**

Interactive image segmentation or foreground extraction is an important component of image editing tools. Studies on separating foreground from background quickly, practically and with little error has grown enormously over the last decade. Interactive image segmentation has usage in photo mounting, (medical) image processing etc. In general, interactive segmentation by definition is classifying pixels as either foreground or background which are not indicated by users using the information of labeled pixels. Labeling both foreground and background pixels separately is not the only way of expressing object of interest. Other than that, there are ways like drawing a bounding box on object or scribbling object boundaries. However, proposed methods differ according to the user interaction. For example, a method working for the former interaction type will not work for the bounding box where only background pixels are indicated. That is why only methods using the former and the last interaction type is considered in this work.

In this thesis, I worked on estimating the label of unlabeled pixels by using foreground and background information obtained from user and developed a new method. The proposed method first does over segmentation on image and groups similar pixels. Thus, pixel count is greatly reduced. Then, probability of pixels being belonged to foreground or background are calculated by taking into account the spatial location of pixels to the scribble locations. Later, by using these probabilities geodesic distances to user scribbles are calculated. Pixels having distance below a certain value are labeled. Finally, remaining unlabeled pixels are labeled by calculating probabilities using the spatially closest labeled pixel information. Obtained results are compared with the methods using the same marking technique quantitatively on a publicly available dataset.

**Keywords:** Interactive image segmentation, foreground/object extraction, semi supervised segmentation, over segmentation, spatially varying, probability density estimation, geodesic distance

## TEŐEKKÜR

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



## İÇİNDEKİLER

TEZ BİLDİRİMİ.....	iii
GEODEZİK MESAFE TABANLI ETKİLEŞİMLİ RESİM BÖLÜMLEMESİ.....	iv
ÖZET.....	iv
ABSTRACT.....	vi
TEŞEKKÜR.....	viii
İÇİNDEKİLER .....	ix
ŞEKİLLERİN LİSTESİ .....	xi
TABLOLARIN LİSTESİ.....	xii
KISALTMALAR .....	xiii
SEMBOL LİSTESİ .....	xiv
1. GİRİŞ .....	1
2. İLGİLİ ÇALIŞMALAR .....	3
2.1. Çizge Kesim Metodu .....	3
2.2. Uyarlanabilir GMMRF modeli.....	6
2.3. Rastgele Yürüyüşler .....	7
2.4. Geodezik Bölümleme .....	8
2.5. İletimsel Bölümleme .....	9
2.6. Kısıtlı Rastgele Yürüyüşler .....	10
2.7. Geodezik Çizge Kesim Metodu .....	12
3. SÜPER PİKSEL BÖLÜMLEMESİ.....	16
3.1. Süper piksel Kafesleri .....	16
3.2. Turbo Pikseller .....	18
3.3. Enerji Eniyilemesi ile Süper piksel Çıkarımı .....	19

3.4.	Basit Doğrusal Yinelemeli Kümeleme (SLIC) ile Süper piksel Çıkarımı ..	20
3.4.1.	Uzaklık Ölçüsü.....	21
3.4.2.	Algoritma .....	22
3.4.3.	Zaman Karmaşıklığı.....	22
3.5.	Diğer Yöntemler.....	23
4.	GEODEZİK BÖLÜMLEME .....	25
4.1.	Olasılık Yoğunluk Tahmini.....	25
4.1.1.	Çekirdek Yoğunluk Tahmini.....	25
4.1.1.1.	Renk modellemesi .....	26
4.1.2.	Hızlı Gauss Dönüşümü .....	27
4.2.	Geodezik Mesafe .....	27
4.2.1.	Hızlı Yürüyüş Metodu .....	28
4.3.	Metot .....	29
5.	GEODEZİK MESAFE TABANLI, İKİLİ BÖLÜMLEME ALGORİTMASI ..	31
5.1.	Yöntem .....	32
5.2.	Uygulama ve Çalışma Zamanları .....	38
5.2.1.	Uygulama .....	38
5.2.2.	Çalışma Zamanları .....	39
6.	DENEYLER.....	40
6.1.	Değerlendirme .....	51
7.	SONUÇ .....	52
	KAYNAKLAR .....	53
	ÖZGEÇMİŞ .....	56

## ŞEKİLLERİN LİSTESİ

Şekil 2.1.1.	Çizge kesimi için oluşturulan çizge.....	4
Şekil 2.1.2.	Maksimum akışı sağlayan yolun gösterimi.....	5
Şekil 2.5.1.	Tümevarımsal ve iletimsel çıkarımın karşılaştırılması.....	9
Şekil 2.6.1.	Sınır bölgelerine ek yumuşak kısıtların eklenmesi.....	11
Şekil 2.7.1.	Çizge kesim metodunun erken kesim problemi.....	12
Şekil 2.7.2.	Geodezik metodunun farklı işaretleme yerlerine göre verdiği sonuçlar	13
Şekil 3.1.1.	Maksimum kesim ile en iyi yolun bulunması.....	17
Şekil 3.1.2.	Dinamik programlama ile en iyi yolun bulunması.....	18
Şekil 3.2.1.	Turbo piksel bölümlenmesinin aşamaları.....	19
Şekil 3.3.1.	Örtüşen parçaların gösterimi.....	19
Şekil 3.3.2.	HGD ile kaynak ve hedeflerin kümelenerek eşleştirilmesi.....	27
Şekil 4.2.1.	Yüzeyin zaman içerisinde dışarıya doğru yayılması.....	28
Şekil 4.3.1.	İki piksel arasındaki 4-komşuluk üzerinden en kısa ağırlıklı yol.....	29
Şekil 5.1.0.	Geodezik bölümleme metodunun iyi çalışmadığı örnekler.....	31
Şekil 5.1.1.	Süper piksel bölümlenmesi örnekleri.....	32
Şekil 5.1.2.	Piksel konumlarına göre farklı oyf elde edilmesi.....	34
Şekil 5.1.3.	Uzaysal değişken bant genişliği kullanmanın etkisi.....	34
Şekil 5.1.4.	Konumsal olarak sabit ve değişken oyf tahmini sonuçları.....	35
Şekil 5.1.5.	Geodezik mesafe hesaplaması sonucu elde edilen ön plan olasılık.....	36
Şekil 5.1.6.	Geodezik mesafe hesaplaması sonucu elde edilen bölümleme.....	37
Şekil 5.1.7.	Sınıf değerleri kesin olmayan piksellerin gösterimi.....	37
Şekil 5.1.8.	Piksel düzeyinde bölümleme ve morfolojik iyileştirme.....	38
Şekil 5.2.1.	CUDA iş parçacığı ve hafıza modeli.....	39
Şekil 6.1.0.	Grabcut veri kümesi.....	40
Şekil 6.2.0.	Grabcut veri kümesi üzerinde alınan sonuçlar.....	49
Şekil 6.3.0.	Farklı işaretleme türüyle elde edilen sonuçlar.....	50

## TABLULARIN LİSTESİ

Tablo 5.1.	Uygulamanın ortalama çalışma süreleri.....	39
Tablo 6.1.	Bölünme Hata Oranları.....	41

## KISALTMALAR

Kısaltma	Açıklama
<b>SLIC</b>	Basit Doğrusal Yinelemeli Kümeleme
<b>oyf</b>	Olasılık Yoğunluk Fonksiyonu
<b>GPU</b>	Grafik İşlemci Ünitesi
<b>CPU</b>	Merkezi İşlem Birimi
<b>HGD</b>	Hızlı Gauss Dönüşümü
<b>HYM</b>	Hızlı Yürüyüş Metodu
<b>GMMRF</b>	Gaussian Mixture Markov Random Field

## SEMBOL LİSTESİ

Bu çalışmada kullanılmış olan simgeler açıklamaları ile birlikte aşağıda sunulmuştur.

<b>Simge</b>	<b>Açıklama</b>
$F$	Ön plan sınıfı
$B$	Arka plan sınıfı
$\Omega_F$	Ön plan kümesi
$\Omega_B$	Arka plan kümesi
$P$	Piksel kümesi
$P_F$	Ön plan sınıfına aitlik olasılığı
$P_B$	Arka plan sınıfına aitlik olasılığı
$D_F$	Ön plan sınıfına olan geodezik mesafe
$D_B$	Arka plan sınıfına olan geodezik mesafe
$K$	Çekirdek fonksiyonu
$\sigma$	Bant genişliği

## 1. GİRİŞ

Etkileşimli bölümler, denetimsiz bölümlerden farklı olarak resmi oluşturan piksellerin kullanıcı girdisi yardımıyla gruplara ayrılmasıdır. Denetimsiz bölümlerde ise farklı olarak resmi oluşturan anlamlı bölgeler kullanıcı girdisi olmadan tespit edilmeye çalışılmaktadır. Arka planı karışık olmayan resimlerde oldukça iyi sonuçlar veren denetimsiz metotlar olmasına rağmen arka planın karışık olduğu resimlerde bu yöntemlerin istenilen sonuçları vermediği görülmüştür. Az miktarda kullanıcı girdisi ile etkileşimli yöntemlerin karışık arka planları ayırmada denetimsiz çalışan metotlara göre çok daha iyi sonuçlar verdiği görülmüştür. Ayrıca, denetimsiz bir yöntemle istenilen düzeyde bir bölümler elde edilse dahi amaç resimdeki bir objenin arka plandan ayrılması ise yine kullanıcı etkileşimi ile arka plandan ayrılmak istenen nesnenin belirtilmesi gerekmektedir. Ek olarak, arka plandan ayrılmak istenen nesne birden fazla bölgeden oluşabilir. Bu durumda denetimsiz yöntemler objeyi birden fazla bölgeye ayıracaktır. Böyle bir durumda bu bölgelerin aynı objeye ait olduklarının yine açıkça belirtilmesi gerekmektedir. Karmaşık resimlerde, resmi oluşturan anlamlı bölgeler sübjektif bir yapı sergileyebilir. Bu nedenlerle etkileşimli bölümler, denetimsiz bölümlerden farklı bir konu olarak bilgisayarla görü alanında çalışılmaktadır. Esasında etkileşimli bölümler de kendi içerisinde ikili ya da çoklu olmak üzere ikiye ayrılabilir. İkili sınıflandırmada kullanıcılar ön ve arka plan olmak üzere iki ayrı sınıf belirtirken, çoklu sınıflandırmada ikiden fazla sınıf belirtilir. Buna göre resim belirtilen sınıf sayısı kadar birbiriyle kesişmeyen bölgelere ayrılır. Bu tez çalışmasında ikili sınıflandırma için çalışan bir yöntem geliştirilmiştir.

Bu tez çalışmasında, renkli resimlerin etkileşimli bölümleri üzerinde çalışan bir algoritma geliştirilmiştir. Piksel tabanlı yaklaşımlardan farklı olarak resim, hızlı ve sınır bölgelerine riayet eden bir fazla bölümler algoritması ile piksel gruplarına ayrılmıştır. Klasik yaklaşımlardan farklı olarak uzaysal yakınlığı da gözetilen renk modellemesi yapılmıştır. Önerilen yöntemin çalışma zamanını hızlandırmak için uygulamanın bazı kısımları grafik işlem ünitesini kullanan paralel bir kütüphane

kullanılarak gerekleřtirilmiřtir. Algoritmanın performansı umumi olarak eriřeme aık bu alandaki alıřmalarda kullanılmak üzere sunulmuř bir veri kümesi üzerinde manüel olarak ıkartılmıř ön plan resimleriyle ne kadar örtüřtüęü ölçülmüř ve benzer alıřmalarla bir kıyas yapılmıřtır.

Tez alıřması řu řekilde düzenlenmiřtir. Bölüm 1 tez konusunu oluřturan problemin tanımını derinleřtiren ve bu problemin özümü için önerilen metodun kabaca hangi ařamalardan oluřtuęunu anlatan giriş bölümünden oluřmaktadır. Bölüm 2 de problemin özümü için önerilmiř aynı etkileřim yöntemini kullanan ve yeni ufuklar aan alıřmalar incelenmiřtir. 3. bölümde önerilen yöntemin ön ařamasını oluřturan fazla bölümlene algoritmaları incelenmiř ve kullanılan yöntemin neden seçildięi anlatılmıřtır. Bölüm 4 de önerilen algoritmanın temelini oluřturan ve bölüm 2 de bahsedilen geodezik bölümlene metodunun anlatımı derinleřtirilmiřtir. Bölüm 5 de önerilen algoritma detaylandırılırken 6. bölümde deneyler ve elde edilen sonuçlara yer verilmiřtir. Tez alıřması 7. ve son bölüm ile yapılan alıřmaya ait izlenimlerin aktarılmasıyla ve gelecek alıřmaların ne doęrultuda olacaęı belirtilerek sonlandırılmıřtır.



## 2. İLGİLİ ÇALIŞMALAR

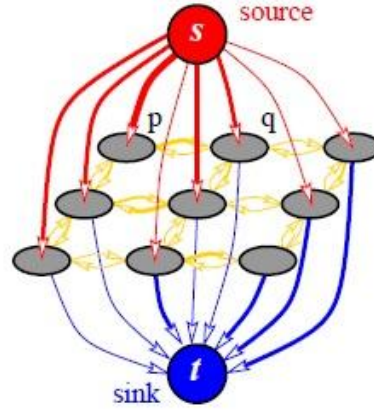
Bu bölümde iki sınıflı, etkileşimli bölümlenme yöntemleri kronolojik olarak kısaca özetlenmiştir. Bu bölümde özetlenen çalışmalar, problemin çözümüne yönelik önemli kazanımlar sağlayan ve popüler olmuş yöntemlerdir. Yapılan incelemelere göre bu yöntemler çizge kesim, rastgele yürüyüş ve geodezik tabanlı olarak göze çarpmaktadır. Diğer önerilen yöntemler bu metotları temel alarak ortaya atılmıştır.

### 2.1. Çizge Kesim Metodu

2001 yılında Boykov ve diğerleri tarafından önerilen çizge kesim metodu [1] etkileşimli bölümlenme konusunda atılan en büyük adımlardan biri olarak literatüre geçmiştir. Bu alt bölümde bu metodun problemin çözümüne nasıl yaklaştığı anlatılmaktadır. Çizge kesim metodu, 2.1.1 ile ifade edilen maliyet fonksiyonunu minimize etmeye çalışır.

$$E(L) = \sum_{p_i \in P} R_{L_i}(p_i) + \lambda \sum_{(p_i, p_j) \in N} B(p_i, p_j) |L_i - L_j| \quad (2.1.1)$$

Buna göre;  $L = (L_i)$  her bir piksele karşılık gelen ikili ( $L_i \in F | L_i \in B$ ) sınıf vektörü,  $R_{L_i}(p_i)$  piksel sınıfına bağlı değişen bölge maliyet terimi,  $B(p_i, p_j)$  ise birbirlerine komşu piksellerin sınıfına bağlı sınır maliyet terimidir.  $\lambda \geq 0$  ise bölge ve sınır maliyetleri arasındaki ödünleşmeyi sağlayan etkendir.  $P$  resmi meydana getiren pikselleri temsil eden küme iken,  $N$  ise komşu pikselleri içeren kümedir. Genellikle  $R$  her bir pikselin bireysel olarak ön veya arka plana aitliğini ifade ederken,  $B$  birbirine komşu olan piksellerin benzerliğini ifade etmektedir. Şekil 2.1.1 [2] de ifade edildiği gibi her bir piksel bireysel olarak ön ve arka planı temsil eden düğümlere bağlıdır ( $R$  terimi). Bunun yanı sıra komşu pikseller de birbirlerine bağlanmıştır ( $B$  terimi). Yukarıdaki maliyet fonksiyonunun minimize edilmesi problemi maksimum akış/minimum kesim problemine dönüştürülerek çözülmesi fikri ortaya çıkmıştır.



Şekil 2.1.1. Çizge kesimi için oluşturulan çizge

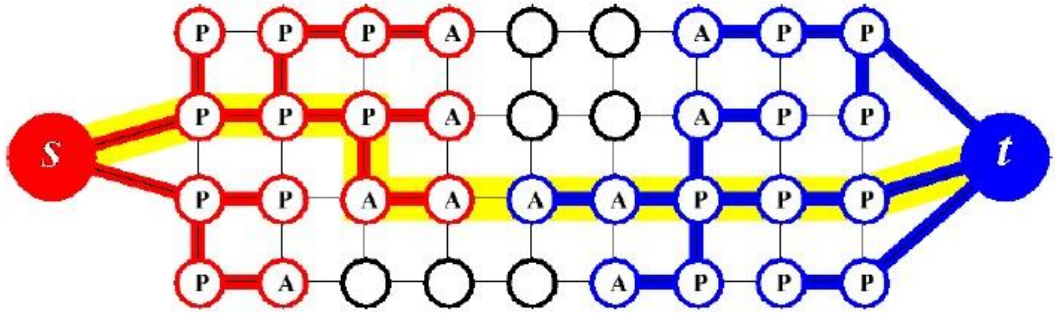
Buna göre çizgeyi ikiye ayıran minimum kesim, maliyet fonksiyonunun minimize edilmesine yaklaşık bir sonuç vermektedir. Kesimin maliyeti, bağlı olduğu piksellerden bir tanesi ön diğeri de arka plana ait olan kenarların ağırlıkları toplamından oluşmaktadır (2.1.2).

$$|C| = \sum_{e \in C} w_e \quad (2.1.2)$$

Maksimum akış/minimum kesim probleminin hızlı çözümü için Boykov ve diğeri bir algoritma geliştirmişlerdir. [2] Bu algoritmaya göre kökleri kaynak  $s$  ve hedef  $t$  düğümleri olan, birbiriyile çakışmayan, iki ayrı arama ağacı oluşturulmaktadır:  $S$  ve  $T$ . Buna göre  $S$  ağacındaki ebeveynden çocuğa doğru olan her bir kenar doymamış iken,  $T$  ağacında çocuktan ebeveyne doğru olan kenarlar doymamıştır.  $S$  veya  $T$  de yer almayan düğümler ise serbest olarak işaretlenmiştir. Arama ağaçlarındaki düğümler aktif ve pasif olmak üzere ikiye ayrılmıştır. Aktif düğümler dış sınır düğümleri iken, pasif düğümler içteki düğümlerdir. Buradaki kilit nokta; aktif düğümler doymamış yollar üzerinden yeni çocuklar edinerek arama ağaçlarını genişletirler. Artan yol, arama ağaçlarının birinde yer alan aktif düğümün öteki ağaçta yer alan bir düğüme komşu olmasıyla açığa çıkar. Algoritma aşağıda belirtilen üç adımı yinelemeli olarak tekrar eder.

- “büyüme” aşaması: arama ağaçları birbirlerine komşu olana kadar büyürler (yol bulunana kadar).
- “çoğalma” aşaması: bulunan yol artırılır, arama ağaçları ormanlara dönüşür.
- “edinme” aşaması: arama ağaçları yeniden düzenlenir.

Büyüme aşamasında ağaçlar genişletilir. Aktif düğümler komşu doymamış kenarlar aracılığı ile serbest düğümlerden yeni çocuklar edinirler. Yeni elde edinilen düğümler arama ağaçlarının aktif düğümleri olurlar. Bir aktif düğümün tüm komşu düğümleri keşfedildikten sonra aktif düğüm pasif hale getirilir. Büyüme aşaması aktif bir düğüm karşı ağaca bağlı komşu bir düğümle karşılaştığı anda durur. Bu durumda kaynaktan hedefe doğru bir yol elde edilir. (Şekil 2.1.2 [2]).



Şekil 2.1.2. Maksimum akışı sağlayan yolun gösterimi

Çoğalma aşamasında büyüme aşamasında bulunan yol artırılır. Olası en fazla akış bu yoldan taşındığı için bu yolda yer alan bazı düğümler doymuş hale gelir. Böylece, arama ağaçlarındaki bazı düğümler kimsesiz kalabilir. Yani, bu düğümleri ebeveynlerine bağlayan kenarlar doymuş hale geldiği için artık geçerli değildirler. Esasında, bu aşama arama ağaçlarını, ormanlara dönüştürebilir.

Edinme aşamasında kimsesiz düğümlere yeni ebeveynler bulunmaya çalışılır. Yeni ebeveyn düğümü kimsesiz düğümle aynı set içerisinde yer almalıdır. Ayrıca, yeni ebeveyn doymamış bir kenar üzerinden bağlanılmalıdır. Eğer böyle bir düğüm bulunamazsa kimsesiz düğüm serbest olarak işaretlenir. Böyle bir durumda, bu düğümün önceki tüm çocukları kimsesiz kalır. Aşama kimsesiz düğüm kalmayınca

kadar devam eder.  $S$  ve  $T$  de yer alan bazı kimsesiz düğümler serbest kaldığından, edinme aşaması bu kümelerin küçülmesine neden olur.

Edinme aşamasından sonra algoritma büyüme aşamasında geri döner. Algoritma, arama ağaçları büyüemeyecek duruma gelinceye kadar devam eder. Bu maksimum akışın elde edildiğinin göstergesidir. Buna karşılık gelen minimum kesim,  $S$  ve  $T$  kümelerine bağlı düğümlerden tespit edilir.

## 2.2. Uyarlanabilir GMMRF modeli

2001 de Boykov ve diğerleri tarafından önerilen çizge kesim metodunda algoritmanın verimli çalışması için  $\lambda$  gibi bazı parametrelerin iyi ayarlanması gerekiyordu. 2004 yılında Blake ve diğerleri [3] tarafından yapılan bu çalışmada el ile ayarlanan sabitlerin resim verisinden tahmin edilmesine yönelik bir çalışma yapılmıştır. Buna göre bir pikselin ön ya da arka plana ait olması aşağıdaki şartlı olasılığa bağlı olarak modellenmiştir.

$$p(\theta|z) = \sum_k p(\theta, k|z) \quad (2.2.1)$$

$$p(\theta, k|z) = \frac{1}{p(z)} p(z|\theta, k) p(\theta, k) \quad (2.2.2)$$

$z$  resmi meydana getiren piksellerin renklerini (RGB) oluşturan vektördür.  $k$  saklı değişkenleri temsil eden karışım indeksidir.  $\theta$  piksellerin ön ya da arka plana aitliğini temsil eden saklı değişkendir. Buna göre her bir pikselin  $0 \leq \theta \leq 1$  oranında ön ve  $1 - \theta$  oranında da arka plana ait olduğu varsayılmaktadır. Ancak buradaki problemin çözümünde her piksel için  $\theta_n = 0, 1$  dir.

Renk gözlemlerini modellemek için basit bir yaklaşım ele alınmıştır: Her bir piksel için ön plan renginin  $K$  tane Gauss karışım bileşiminin bir tanesinden rastgele meydana geldiği varsayılmıştır. Arka plan için de aynı varsayım kullanılmıştır.  $\theta_n = 0, 1$  ikili değeri her bir piksel için bir önceki alt bölümde bahsedilen çizge kesim yöntemi kullanılarak bulunmuştur.

### 2.3. Rastgele Yürüyüşler

2006 yılında Grady [4] tarafından etkileşimli bölümlenme için rastgele yürüyüşler algoritması önerilmiştir. Bu algoritmaya göre işaretlenmemiş her bir pikselin işaretli olan farklı sınıfa ait piksellerden hangisine ilk olarak ulaşacağı sorusuna yanıt aranmıştır.

$$L_{ij} \begin{cases} d_i & i = j \\ -w_{ij} & (v_i, v_j) \in N \\ 0 & \text{diğer durumlarda} \end{cases} \quad (2.3.1)$$

$$A_{e_{ij}v_k} \begin{cases} +1 & i = k \\ -1 & j = k \\ 0 & \text{diğer durumlarda} \end{cases} \quad (2.3.2)$$

Resim çizge olarak ele alınmış ve şu tanımlamalar yapılmıştır:  $G = (E, V)$  çizgesi  $v \in V$  düğümlerinden ve  $e \in E \subseteq V \times V$  kenarlarından meydana gelmektedir. Kenar ağırlıkları  $w_{ij} = \exp(-\beta(g_i - g_j))$  ile bir düğümün derecesi  $d_i = \sum w_{ij}$  ile temsil edilmiştir.  $g_i$  ile piksel yoğunluğu ifade edilmektedir.

$$D[x] = \frac{1}{2}(Ax)^T I(Ax) = \frac{1}{2}x^T Lx = \frac{1}{2} \sum_{e_{ij} \in E} w_{ij} (x_i - x_j)^2 \quad (2.3.3)$$

Yukarıdaki ifadeyi minimize eden bir harmonik  $x$  fonksiyonunun bulunması ile problem formulize edilmiştir.  $L$  matrisi yarı pozitif tanımlı bir matris olduğu için  $D[x]$  in sadece kritik noktaları, minimum sonucu verecektir. Düğümleri işaretli  $V_M$  ve işaretli  $V_U$  ( $V_M \cup V_U = V$  ve  $V_M \cap V_U = \emptyset$ ) olarak ayırırsak, (2.3.3) de belirtilen ifade aşağıdaki şekilde ayrıştırılabilir.

$$D[x_U] = \frac{1}{2} [x_M^T \ x_U^T] \begin{bmatrix} L_M & A \\ A^T & L_U \end{bmatrix} \begin{bmatrix} x_M \\ x_U \end{bmatrix} = \frac{1}{2} (x_M^T L_M x_M + 2x_U^T A^T x_M + x_U^T L_U x_U) \quad (2.3.4)$$

Yukarıdaki ifadenin  $x_U$  ya bağlı türevini alıp kritik noktaları bulmamız aşağıdaki ifadeyi elde etmemizi sağlar.

$$L_U x_U = -A^T x_M \quad (2.3.5)$$

Böylelikle doğrusal denklemlerden oluşan  $|V_U|$  bilinmeyenli bir sistem elde edilmiştir. Her bir sınıf için bu doğrusal sistemlerin çözülmesiyle işaretli piksellerin ataması gerçekleştirilmiştir.

#### 2.4. Geodezik Bölümleme

Bai ve diğerleri [5] tarafından 2007 yılında önerilen metotta kullanıcı tarafından işaretlenen ön plan ( $F$ ) ve arka plana ( $B$ ) ait piksellere olan geodezik mesafeler hesaplanarak pikseller sınıflandırılmıştır. Geodezik mesafe  $d(p)$ ; işaretlenmiş piksellerden  $p$  pikseline olan tüm ağırlıklı yolların en küçük parçası olarak tanımlanmıştır (rastgele yürüyüşlerde farklı olarak ortalama mesafe hesaplanmaktadır).  $\Omega_F$  ön plan olarak işaretlenmiş,  $\Omega_B$  de arka plan olarak işaretlenmiş piksellerin kümesi olsun. Bu iki kümeden resimdeki diğer tüm piksellere ( $x$ ) olan ağırlıklı geodezik mesafeler 2.4.1 ile hesaplanmaktadır.

$$D_l(x) = \min_{s \in \Omega_l} d(s, x), \quad l \in \{F, B\} \quad (2.4.1)$$

$$d(s_1, s_2) := \min_{C_{s_1, s_2}} \int_0^1 |W(x) \cdot C_{s_1, s_2}(x)| dx \quad (2.4.2)$$

$C_{s_1, s_2}(p)$ ;  $s_1, s_2$  piksellerini  $p = 0$  ve  $p = 1$  için birbirine bağlayan yollardan biridir.  $W$ ; komşu pikselleri birbirine bağlayan ağırlıkları sembolize etmektedir. Ağırlıklar piksellerin ön plana bağlı olma olasılıkları arasındaki değişim derecesi ile hesaplanmıştır. ( $W = \nabla P_F(x)$ ). Olasılık değerleri işaretlenmiş piksellerden olasılık yoğunluk fonksiyonlarının hesaplanması ile elde edilmiştir (2.4.3).

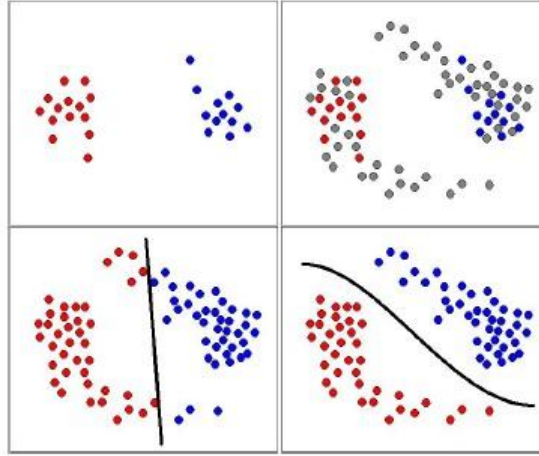
$$P_F(x) = \frac{\Pr(x|F)}{\Pr(x|F) + \Pr(x|B)} \quad (2.4.3)$$

Bu metriğe göre bir piksel işaretli bir piksele olasılık değerinin çok fazla değişmediği bir yol üzerinden yakındır. Bu şekilde her bir piksel için ön ve arka plan piksellerine olan geodezik mesafeler hesaplanarak en yakın olan sınıfa atamaları

gerçekleştirilmiştir. Bu yöntem önerilen metoda temel teşkil ettiği için başlı başına Bölüm 4 de ayrıntılı olarak ele alınmıştır.

## 2.5. İletimsel Bölümlenme

İletimsel bölümlenme [6] Duchenne ve diğerleri tarafından 2008 yılında önerilmiştir. Bu çalışmada öncelikle, kullanılan iletimsel çıkarımın klasik yaklaşım olan tümevarımsal çıkarımdan nasıl ayrıldığı anlatılmıştır.



Şekil 2.5.1. Tümevarımsal çıkarım (sol) ve İletimsel çıkarım (sağ) karşılaştırması

Makine öğrenmesindeki temel konulardan biri sınıfları bilinen örneklerden sınıfları bilinmeyen yeni örneklerin sınıflarının tahmin edilmesidir. Bunun için girdi ve çıktı çiftleri eğitim verisi olarak kullanılır ve yeni girdi verilerinin çıktısı tahmin edilmeye çalışılır. Tümevarımsal çıkarımda yeni gelecek olan girdi kümesi önceden bilinmemektedir. Bu nedenle öğrenme algoritması eğitim verisinden, tüm girdi ve çıktı uzayı için bir eşleme yapmayı öğrenmelidir. Yeni test girdileri geldiği zaman, öğrenilen fonksiyon, bu girdileri karşılık gelen çıktılarla eşleştirmelidir. İletimsel çıkarımda ise farklı olarak eğitim verisinin yanı sıra girdi verisi de önceden bilinmektedir. Sonuç olarak, iki adımdan oluşan öğrenme aşaması (girdi -çıkıtı eşleşmesini öğrenme ve yeni girdiler üzerinde test etme) tek bir aşamaya (girdi

verisine karşılık gelen çıktıları bulma) indirilebilir. Şekil 2.5.1 [6] de iletimsel çıkarımın tümevarımsal çıkarıma karşı bazı avantajları gösterilmiştir.

$$\min_f = \sum_{i \in T} c_i [Y_i - f(X_i)]^2 + \int_M \|\nabla f\|^2 p^s dV \quad (2.5.1)$$

İletimsel çıkarımda, girdi kümesini çıktı kümesine eşleyen bir düz fonksiyon  $f(X_i)$  bulunmaya çalışılır. Bu fonksiyon eğitim verisinde çıktı  $Y_i$  ye yakındır ve sadece düşük yoğunluklu bölgelerde değerinin değişmesine izin verilir. Bunun için, ne kadar düşük yoğunlukta fonksiyonun büyük değişimler göstereceğini ayarlayan bir  $s \geq 0$  serbest parametresi vardır. 2.5.1 de  $c_i$  katsayıları,  $(X_i, Y_i)$  eğitim noktalarını ne ölçüde uydurmak istediğimizi ayarlamamızı sağlamaktadır. Entegral terimi  $f$  fonksiyonunun sadece düşük yoğunluklu bölgelerde değişken olmasına iten terimdir. Burada  $c_i = +\infty$ ,  $f(X_i) = Y_i$  olmasına neden olur. Formüldeki  $p$  yoğunluğu sembolize etmektedir.

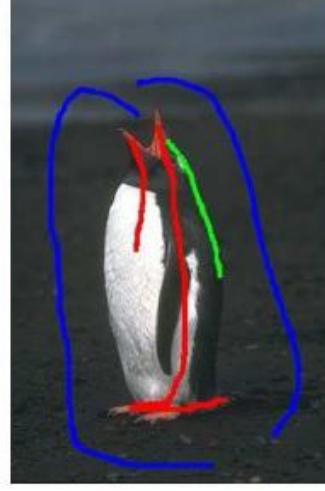
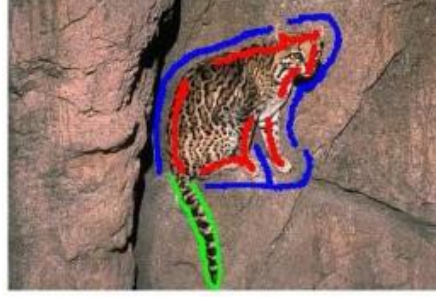
$$\sum_{j \in J} L_{k,j} F_j = - \sum_{i \in \{\text{eğitim pikselleri}\}} L_{k,i} Y_i \quad (2.5.2)$$

$F$ ,  $F_i = f(X_i)$  şeklinde tanımlanan bir vektör olsun. Buna göre  $L$  matrisinin her bir  $k$  satırı için yukarıdaki sistemin çözümü ( $J$ ) piksellerin hangi sınıfa ait olduklarının olasılıklarını verecektir. Burada rastgele yürüyüşlerdeki gibi bir  $L$  matrisi kullanılabilir. Basit bir şekilde  $F_j$  0.5 değeri ile eşiklenerek piksellerin sınıf atamaları gerçekleştirilir.

## 2.6. Kısıtlı Rastgele Yürüyüşler

Yang ve diğerleri tarafından önerilen bu metotta [7] ön ve arka plan işaretlemelerinin yanı sıra kullanıcılar resim üzerine yumuşak kısıt da ekleyebilmektedirler. (Şekil 2.6.1 [7]). Yumuşak kısıt bu bölgede sınırların yer aldığını ifade etmekte ve sınırların yumuşak geçiş gösterdikleri tespit edilmesi zor olan yerlerde ortaya çıkarılmasına katkı sağlamaktadırlar.





Şekil 2.6.1. Sınır bölgelerine ek yumuşak (yeşil) kısıtların eklenmesi

$$p_i = \frac{1}{d_i} \sum_{e_{ij} \in \mathcal{E}} w_{ij} p_j \quad (2.6.1)$$

Rastgele yürüyüşler algoritması 2.6.1 ile verilen doğrusal sistemin çözümünü önermişti. Her  $v_i \in S_F$  için  $p_i = 1$  ve  $v_i \in S_B$  için  $p_i = 0$  olan sistemde her  $v_i \in V \setminus (S_F \cup S_B)$  için  $p_i$  değerleri bulunup, basitçe 0.5 ile eşiklenerek sınıf atamaları gerçekleştirilmişti. Kısıtlı rastgele yürüyüşlerde ise  $S_F$  ve  $S_B$  ye ek olarak  $S_S$  ve  $S_H$  sınıfları da eklenmiştir. Buna göre, yumuşak kısıt üzerinde yer alan bir düğümün ön veya arka plana bağlı olma olasılığının 0.5 e yakın olduğu ve  $[-\varepsilon, \varepsilon]$  gibi küçük bir değer aralığında olduğu varsayılmıştır. Sert kısıt üzerinde yer alan bir düğümün olasılığı ise 0.5 olarak belirlenmiştir. 2.6.1 deki doğrusal sistem aşağıdaki forma dönüştürülmüştür.

$$p_i = \frac{1}{d_i} \sum_{e_{ij} \in \mathcal{E}} w_{ij} p_j, \quad v_i \in V \setminus (S_F \cup S_B \cup S_H) \quad (2.6.2)$$

$$\text{s.t.} \begin{cases} p_i = 1 & v_i \in S_F \\ p_i = 0 & v_i \in S_B \\ p_i = 0.5 & v_i \in S_H \\ |p_i - 0.5| \leq \varepsilon & v_i \in S_S, \varepsilon \approx 0^+ \end{cases}$$

Yukarıdaki problem için yumuşak kısıttan dolayı etkili bir çözücü bulmak zor olduğu için problem aşağıdaki forma dönüştürülmüştür.

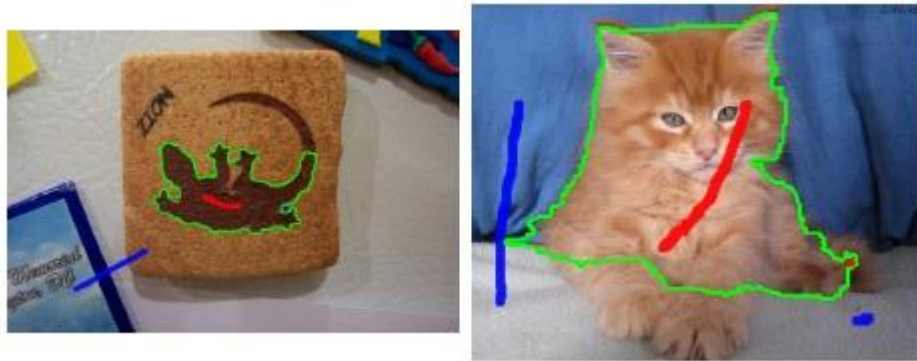
$$\min \sum_{e_{ij} \in \mathcal{E}} w_{ij} (p_i - p_j)^2 + \lambda \sum_{v_i \in S_S} (p_i - 0.5)^2 \quad (2.6.3)$$

$$\text{s.t. } \begin{cases} p_i = 1 & v_i \in S_F \\ p_i = 0 & v_i \in S_B \\ p_i = 0.5 & v_i \in S_H \end{cases}$$

Buradaki formülde  $\lambda$  her bir yumuşak kısıt düğümüyle 0.5 değeri arasındaki ödünleşmeyi sağlayan etkidir. Ayrıca, çizgedeki kenar ağırlıkları ( $w_{ij}$ ) orijinal rastgele yürüyüşler algoritmasına benzer şekilde hesaplanmıştır. Bu sistemin çözümü ile sınır bölgelerine daha duyarlı bir sınıflandırma yapılmıştır.

## 2.7. Geodezik Çizge Kesim Metodu

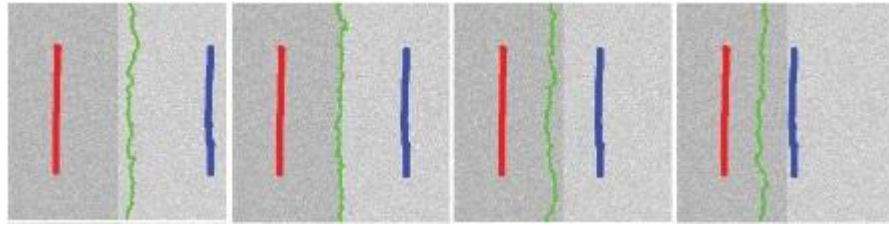
Price ve diğerleri tarafından 2010 yılında etkileşimli bölümlenme için geodezik çizge kesim metodu önerilmiştir. [8] Bu metotta hesaplanan geodezik mesafeler çizge kesim metodu içerisinde kullanılarak pikseller sınıflandırılmıştır. Önerilen metodun detayları aşağıda aktarılmaktadır.



Şekil 2.7.1. Çizge Kesim metodunun erken kesim problemi

2.1.1 ile ifade edilen formülde  $R$  ile piksellerin bireysel olarak ön ve arka plan sınıflarına hangi olasılıkla ait olduğu,  $B$  ile de birbirine komşu olan piksellerin aynı sınıfa ait olma olasılıkları modellenmişti.  $\lambda$  ile bu iki olasılık arasındaki ödünleşme kontrol ediliyordu. Bu metot genel olarak iyi sonuçlar vermesine karşılık temel zayıflığı, sınır teriminin daha kısa sınır bölgeleri üzerinde yoğunlaşmasıdır. (Şekil

2.7.1 [8]). Ancak, bu durumun dengeleyici terim  $\lambda$  nın deęerinin azaltılmasıyla üstesinden gelinebilir. Böylelikle bölge terimi daha baskın rol oynayacak ve bölümlene doğru sınırları kapsayacak şekilde gerçekleştirilecektir. Oysaki bölge teriminin ağırlığının artırılması bu sefer de ön plan rengine yakın arka planların ön plan olarak sınıflandırılmasına yol açabilmektedir. Bunun yanı sıra geodezik bölümlene ile renk modellemesine ek olarak geodezik mesafelerin hesaplanmasıyla daha etkili bir bölümlene sistemi ortaya çıkartılmıştır. Ancak geodezik bölümlenenin de bazı dezavantajları bulunmaktadır. Örneğin Şekil 2.7.2 [8] de de gösterildiği üzere sınıflandırma, işaretleme noktalarına baęlı bir durum sergilemektedir. Bu durumun genellikle ön plan ve arka plan renklerinin birbirine yakın olduęu ve piksellerin ön/arka plana baęlı olma olasılıklarının kararsız bir yapı sergilemesi durumunda açığa çıktığı gözlenmiştir. İkinci dezavantaj olarak geodezik metotta açık bir sınır bulma bileşeninin olmaması, ön ve arka plan renklerinin birbirinden oldukça farklı olması rağmen, nesne sınırlarını kesin olarak belirleyememesine neden olmaktadır. Price ve dięerleri tarafından önerilen metot ile bu iki metodun güçleri birleştirilmiş ve eksiklikleri giderilerek daha etkili bir sistem geliştirilmiştir.



Şekil 2.7.2. Geodezik metodunun farklı işaretleme yerlerine göre verdiği sonuçlar

Yukarıda da belirtildiği gibi, ön ve arka plan renklerinin ayırt edici olmadığı durumlarda geodezik bölümlene metoduna açıkça kenar bilgisini eklenmesi sonucu metodun performansı işaretleme bölgesine baęlı olmaktan kurtarılabilir. Böylece kullanıcılara işaretleme konusunda daha fazla özgürlüğün tanınması sağlanmaktadır. Öte yandan renk bilgisinin ayırt edici olduğu durumlarda, kısa kesim durumuna yol

açan kenar bilgisinin eklenmesi gereksizdir. Aşağıda geodezik mesafe ve kenar bilgisinin çizge kesim eniyilemesi çatısı altında nasıl birleştirildiği anlatılmaktadır.

İlk olarak, geodezik mesafe bilgisinin eklenmesi için 2.1.1 de yer alan  $R$  terimi aşağıda yer alan ifade ile değiştirilmiştir.

$$R_l(p_i) = s_l(p_i) + M_l(p_i) + G_l(p_i) \quad (2.7.1)$$

$M_l$  çizge kesim bölümlenmesinde kullanıldığı gibi küresel renk modelini temsil eder.  $G_l$  ile de geodezik mesafeler belirtilmiştir.  $s_l$  aşağıdaki gibi tanımlanmıştır.

$$s_l(p_i) = \begin{cases} \infty & p_i \in F \text{ veya } p_i \in B \\ 0 & \text{diğer durumlarda} \end{cases} \quad (2.7.2)$$

Yukarıdaki değişim ile birlikte geodezik çizge kesim metodunun genel olarak, yalnız geodezik ve yalnız çizge kesim metotlarından daha iyi performans verdiği gözlenmiştir. Ancak, geodezik mesafeye bağlı  $R$  teriminin, geodezik yönteminin daha iyi çalışması beklenen (ön ve arka plan renk dağılımlarının ayırt edilebildiği) durumlarda daha fazla ağırlıklandırılması için çalışılmıştır. Bu geodezik mesafeye bağlı  $R$  terimine daha fazla ağırlık verilmesi ile  $B$  teriminin yol açtığı kısa kesim probleminin önüne geçilmeye çalışılmıştır. Bölge ve sınır terimlerini ayrı ayrı ağırlıklandırmak için tek bir  $\lambda$  etkeni yerine bölge teriminin ağırlığını kontrol eden  $\lambda_R$  ve sınır terimini kontrol eden  $\lambda_B$  faktörleri eklenmiştir.  $\lambda_B$  klasik çizge kesim yaklaşımında olduğu gibi kullanılmış ve sadece resim boyutu gözetilerek ayarlanmıştır.  $\lambda_R$  geodezik mesafe teriminin başarısı tahmin edilerek otomatik olarak ayarlanmıştır. Bunun için ön plana bağlı olan örneklerin arka plana bağlı olma olasılıklarının toplamı ön plan örnek sayısı ile normleştirilmiş, arka plana bağlı örneklerin ön plana bağlı olma olasılıklarının toplamı da arka plan örnek sayısı ile normleştirilerek bu iki değerlerin ortalaması alınmıştır. (2.7.3)

$$\varepsilon = \frac{1}{2} \left[ \frac{\sum_{p \in F} P_B(C(p))}{|\Omega_F|} + \frac{\sum_{p \in B} P_F(C(p))}{|\Omega_B|} \right] \quad (2.7.3)$$

$\varepsilon = 0$  durumunda, renk tabanlı terimlere ( $M_l$  ve  $G_l$ ) tam ağırlık verilirken, renk modelleri ayırt edilemez durumda iken ( $\varepsilon = 0.5$ ) 0 ağırlık verilmesi ön görülmüştür. Buna göre  $\lambda_R$  aşağıdaki şekilde tanımlanmıştır.

$$\lambda_R = \begin{cases} 1 - 2\varepsilon & \varepsilon < 0.5 \\ 0 & \text{diğer durumlarda} \end{cases} \quad (2.7.4)$$

Resimdeki küresel renk özelliklerinin gözetilmesi bölge ve sınır terimlerinin farklı resim türlerine göre otomatik olarak ayarlanmasına yardımcı olsa da aynı resimdeki farklı yerel bölgelerin özelliklerini dikkate almamaktadır. Bunun için geodezik ve sınır terimleri ilaveten geodezik bileşenin yerel güvenilirliği ( $u(p)$ ) gözetilerek ağırlıklandırılmıştır.

$$u(p_i) = \left| \frac{D_F(p_i) - D_B(p_i)}{D_F(p_i) + D_B(p_i)} \right|^\gamma \quad (2.7.5)$$

DeneySEL olarak  $\gamma = 2, 2.5$  in iyi çalıştığı gözlemlenmiştir. Bu ifadenin de eklenmesiyle birlikte bölge terimi şu şekilde güncellenmiştir:

$$R_l(p_i) = s_l(p_i) + M_l(p_i) + u(p_i)G_l(p_i) \quad (2.7.6)$$

Böylece bir pikselin ( $p_i$ ) açık bir şekilde ön plan içinde ( $u(p_i) \approx 0$ ) veya dışında ( $u(p_i) \approx 1$ ) bulunduğu durumlara göre geodezik terim ağırlıklandırılmıştır.

Geodezik terimin klasik çizge kesim enerji fonksiyonuna eklenmesi ve bölge ve sınır terimlerinin otomatik olarak küresel ve yerel olarak dengelenmesi bu metodun temel katkısı olarak göze çarpmaktadır. Enerji fonksiyonu daha sonra önerilen minimum kesim algoritmalarından biriyle minimize edilerek piksellerin sınıflandırılması gerçekleştirilmiştir.

### 3. SÜPER PİKSEL BÖLÜMLEMESİ

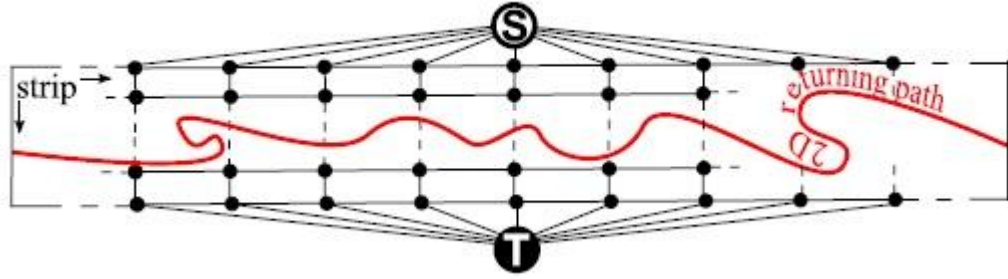
Süper piksel bölümlenme algoritmaları resmi meydana getiren pikselleri algısal olarak anlamlı bir şekilde gruplamak için geliştirilmiştir. Piksellerin konum ve renk bilgilerinin büyük ölçüde muhafaza edilerek, sayıca çok daha az indirgenmesi süper piksel algoritmalarını bilgisayarla görü alanında sıkça kullanılan bir ön işleme adımı haline getirmiştir. Süper pikselleri meydana getirmek için önerilen ve kendilerine göre avantajları ve dezavantajları bulunan birçok algoritma önerilmiştir. Örneğin, uygulamanın resimdeki sınır bölgelerine duyarlı çalışması gerekiyorsa çizge tabanlı süper piksel algoritması iyi bir seçim olabilir. Öte yandan, süper pikseller daha sonra bir çizge oluşturmak amacıyla kullanılacaksa bunun için de düzenli ızgara yapısını koruyan bir süper piksel algoritmasının tercih edilmesi daha uygun olabilir. [9] Bu nedenle bir uygulama içerisinde ön işleme amacıyla kullanılması planlanan süper piksel algoritmasının uygulamaya uygun olarak seçilmesi gerekmektedir. Ancak genel olarak bir süper piksel algoritmasının belirtilen şu özellikleri taşıması uygulamalarda etkili sonuçlar alabilmek için önem arz etmektedir: [9] Oluşturulan süper piksellerin resim içindeki sınır bölgelerini ayırıcı olmalıdır. Bunun yanı sıra, hesaplama karmaşıklığını düşürmek için ön adım olarak kullanılan süper piksel algoritması hızlı hesaplanabilir olmalıdır. Bu bölümde son yıllarda önerilen ve popüler olan süper piksel algoritmaları özetlenmiştir.

#### 3.1. Süper piksel Kafesleri

2008 yılında Moore ve diğerleri tarafından önerilmiştir. [10] Bu yöntemde süper pikseller, piksellerin düzenli ızgara topolojileri korumaktadır. Algoritmanın çalışma şekli aşağıda aktarılmıştır.

Algoritma, girdi olarak sınır haritası alır. Sınırlar  $[0,1]$  arasındaki değerler ile ifade edilmiştir. Sınır haritasında, iki piksel arasında büyük olasılıkla sınır olması 0, olmaması 1 ile değerlendirilmiştir. Amaç, resmi bu sınır değerlerinin düşük olduğu yerlerde bölümlenmektir. Bunun için minimum ağırlıklı yollar tespit edilmiştir. Süper piksel kafesleri artırımlı olarak oluşturulmuştur. İlk olarak resim yatay ve dikey

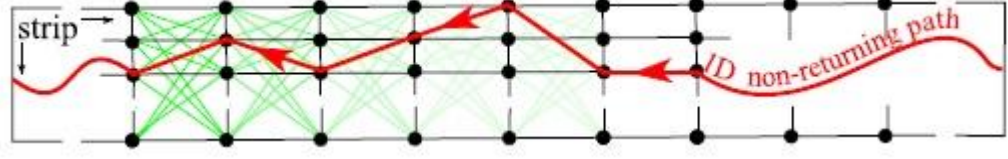
olarak ikiye ayrılmıştır. Buna göre 4 süper piksel oluşmuştur. Her tekrar eden bir alt adımda yeni yatay ve dikey yollar eklenmiştir. Düzenli topoloji her bir yatay ve dikey yolun sadece bir kez kesiştirilmesiyle garanti edilmiştir. Algoritma her adımda sınır haritasında yer alan değerler üzerinden en iyi yolları belirlemeye çalışmaktadır. Yolların rastgele dolaşımını engellemek için önceden belirlenen aralıklar içerisinde kalmaları şart koşulmuştur. Böylece yolların resim içerisinde çapraz olarak hareket edilmesinin önüne geçilmiştir. Ayrıca, bu şekilde göz önünde bulundurulması gereken yolların sayısı azaltılarak hesaplama sayısı azaltılmıştır.



Şekil 3.1.1. Minimum kesim ile en iyi yolun bulunması

Minimum ağırlıklı yolların tespit edilmesi için iki farklı yöntem önerilmiştir. İlk olarak minimum kesim ile bir çözüm aranmıştır. Dilimin bir ucundaki düğümler kaynak düğüme, diğer ucundakiler de hedef düğüme bağlanmıştır. (Şekil 3.1.1 [10]). Düğümler arasındaki kenar ağırlıkları sınır haritası ile belirlenmiştir. Ayrıca dilime dik konumdaki kenarlara sabit bir değer eklenmiştir.

Böylece yolun eğriliği, düz çizgiden sapması, kontrol edilmiştir. İkinci yöntem olarak dinamik programlama önerilmiştir. Bunun için farklı bir çizge oluşturulmuştur. (Şekil 3.1.2 [10]). Burada yolun eğriliği çapraz kenarlara eklenen ağırlıklarla kontrol edilmiştir.



Şekil 3.1.2. Dinamik programlama ile en iyi yolun bulunması

Paralel yolların birbirleriyle kesişmesini önlemek için, her bir yolun oluşturulmasından sonra sınır maliyet haritası güncellenmiştir. Ayrıca yolların birbirlerine çok yakın olmasını önlemek için de komşu şeritlerde yer alan düğümlerin kenar ağırlıkları artırılmıştır. Böylelikle çok küçük süper piksellerin oluşmasının önüne geçilmiştir.

### 3.2. Turbo Pikseller

Levinshtein ve diğerleri tarafından 2009 da önerilmiştir. [11] Algoritmanın çalışması adım adım şöyledir:

İlk olarak, elde edilmek istenen süper piksel sayısı kadar tohum, ızgara yapısındaki piksellere yerleştirilir. (Piksellerden bazıları düzenli aralıklarla işaretlenir). Bu işaretlenen pikseller yüksek değişim derecesine sahip noktalarda yer alıyor ise daha düşük değişim derecesine sahip yakın noktalara kaydırılırlar. Tüm tohum pikselleri “atanmış” olarak işaretlenir.  $\psi^n$  kümesini “atanmış” piksellere olan işaretli Öklid mesafeleri olarak tanımlarsak; 3.2.1 ile “atanmış” piksellere değer verilir.

$$\sum_{x,y}[\psi^n(x,y) \geq 0] \quad (3.2.1)$$

Sonraki adımda piksel benzerlikleri hesaplanır  $\phi(x,y)$ . “atanmış” piksellerin değer dönüşümü belirli bir değerden büyük olduğu sürece aşağıdaki adımlar tekrarlanır.

Resmin vektörel hızı  $S_I$  ve sınır vektörel hızı  $S_B$  hesaplanır.  $\psi^n$ ,  $i$ . izokonturunda  $S_I S_B$  hızında genişletilir.  $\psi^{n+1}$ ,  $\psi^n$  dar şerit içerisinde evrimleştirilerek hesaplanır. 3.2.1 e göre “atanmış” piksel değerleri güncellenir.



$\psi^n$  bulunduktan sonra, homotopik iskeleti çıkartılarak süper pikseller bulunur. Bu çalışmada süper piksellerin bulunması bir geometrik akış çatısı altında modellenmiş ve süper pikseller evrimsel olarak ortaya çıkarılmıştır. Şekil 3.2.1 [11] de algoritmanın yukarıda anlatılan aşamaları gösterilmiştir.



Şekil 3.2.1. Turbo piksel bölümlenmesinin aşamaları

### 3.3. Enerji Eniyilemesi ile Süper piksel Çıkarımı

Algoritma Veksler ve diğerleri tarafından 2010 yılında önerilmiştir. [12] Süper piksel çıkarımı, çizge kesim bölümlenmesindeki gibi bir enerji eniyilemesi çatısı altında ele alınmıştır. Bunun için 2.1.1 deki enerji formülü kullanılmıştır. Şekil 3.3.1 [12] de gösterildiği gibi birbiriyle örtüşen sabit ebatlı parçalar ile kaplanmıştır. Amaç, işlemin sonunda her bir pikseli tek bir parça ile eşleştirmektir. Bunun için süper piksellere karşılık gelen sınıflar  $L = \{1, 2, \dots, k\}$  ile numaralandırılmıştır.  $S(l)$  ile  $l \in L$  parçasına bağlı piksellerin kümesi temsil edilmektedir. Buna göre  $l$  sınıfı yalnızca  $S(l)$  içinde yer alan piksellere atanabilir. Bu nedenle 2.1.1 deki tekil terim 3.3.1 ile ifade edilmiştir.



Şekil 3.3.1. Örtüşen parçaların gösterimi

$$R(p) = \begin{cases} 1 & p \in S(l) \\ \infty & \text{diğer durumlarda} \end{cases} \quad (3.3.1)$$

İkili terim ise;

$$B(p, q) = \exp\left(-\frac{(I_p - I_q)^2}{\text{dist}(p, q) \cdot 2\sigma^2}\right) \quad (3.3.2)$$

ile ifade edilmiştir. Burada,  $I$  piksel yoğunluğunu,  $\text{dist}$  iki komşu piksel arasındaki Öklid mesafesini sembolize eder. Burada dikkat edilmesi gereken,  $R(p)$  nin tüm olası sınıflandırmalar için aynı olduğudur. Bu nedenle iki terim arasındaki ödünleşmeyi sağlayan  $\lambda$  faktörünün optimizasyona bir etkisi yoktur. O yüzden  $\lambda = 1$  seçilmiştir. Genellikle  $\lambda$  sonucu etkileyen ve doğru ayarlanması gereken bir parametredir. Bölge ve sınır terimlerini dengeleyen faktör olduğu için sınır uzunluklarına da etki eder. Ancak bu optimizasyon yapısında sınır uzunluğu,  $\lambda$  yerine parça boyutu ile ayarlanmaktadır. Büyük parça seçimi daha az sınır elde edilmesine neden olmaktadır. Enerji yapısı oluşturulduktan sonra parçalar resim üzerine karenin (parçanın) bir kenarının dörtte biri aralıklarla yerleştirilmiştir. Daha sonra minimum enerji çizge kesimi ile bulunarak süper pikseller elde edilmiştir.

### 3.4. Basit Doğrusal Yinelemeli Kümeleme (SLIC) ile Süper piksel Çıkarımı

Bu yöntem 2010 yılında Achanta ve diğerleri tarafından önerilmiştir. [13] Yaklaşım olarak, piksellerin birbirlerine renk ve resim düzlemindeki yakınlıklarına göre kümelenmesi benimsenmiştir. Bu yöntem tezde önerilen metotta ön aşama olarak kullanıldığı için daha detaylı verilmiştir.

Piksellerin kümelenmesi 5 boyutlu  $[\text{labxy}]$  uzayında gerçekleştirilmiştir. Burada lab, kullanılan CIELAB renk uzayını xy de piksel pozisyonu temsil etmektedir. CIELAB renk uzayının tercih edilmesi küçük renk mesafeleri için algısal olarak tekdüze olduğunun gözlemlendiği içindir. Renk uzayındaki (lab) maksimum renk mesafesi belirli bir değerle sınırlıyken, xy düzlemindeki uzaysal mesafe resmin boyutuna bağlıdır. Bu nedenle uzaysal mesafeleri normalleştirmeden direkt 5 boyutlu Öklid

mesafesinin kullanılması mümkün değildir. 5 boyutlu uzayda kümeleme yapabilmek adına süper piksel boyutunu da göz önünde bulunduran bir mesafe ölçümü yapılmıştır. Bu ölçüm sayesinde renk benzerliği ve konumsal yakınlığın birlikte kullanılmasına olanak sağlanmıştır.

### 3.4.1. Uzaklık Ölçüsü

Algoritma girdi olarak resmin yanında elde edilmek istenen süper piksel sayısını ( $K$ ) alır.  $N$  pikselden meydana gelen bir resim için bir süper piksel  $N/K$  pikselden oluşmaktadır. Yani; her  $S = \sqrt{N/K}$  ızgara aralığında bir süper piksel merkezi yer almaktadır. Her bir süper pikselin uzaysal boyutu yaklaşık olarak  $S^2$  olduğu için bu süper piksel sınıfına ait piksellerin, süper piksel merkezinin  $2S \times 2S$  lik alanında yer aldığı rahatlıkla varsayılabilir. Başka bir deyişle, bir süper piksele piksel ataması yapılacakken sadece bu alan içerisinde yer alan piksellerden seçim yapılabilmektedir.

Renk uzayında olabilecek maksimum mesafenin sınırlı olduğu belirtilmiştir. Uzaysal mesafelerin bu mesafeden fazla olmaları durumunda ise, konumsal yakınlığın renk benzerliğinden daha baskın hale geldiği ve sonuç olarak süper piksellerin resimdeki sınır bölgelerine riayet etmedikleri gözlemlenmiştir. Bu nedenle 5 boyutlu uzayda sıradan Öklid mesafesi kullanmak yerine 3.4.1 deki uzaklık ölçüsü önerilmiştir.

$$d_{lab} = \sqrt{(l_k - l_i)^2 + (a_k - a_i)^2 + (b_k - b_i)^2}$$

$$d_{xy} = \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2} \quad (3.4.1)$$

$$D_s = d_{lab} + \frac{m}{S} d_{xy}$$

Yukarıdaki gösterimde  $i$  ile pikseller  $k$  ile de süper pikseller ifade edilmektedir. Formülden de anlaşıldığı üzere  $D_s$  renk uzaklığı ile ortalama iki süper piksel arasındaki uzaklık ile normalleştirilmiş uzaysal mesafenin toplamından oluşmaktadır.  $m$  değişkeni süper piksellerin sıkı yapısını kontrol etmek için ileri sürülmüştür.  $m$

nin değeri artırıldıkça uzaysal yakınlık baskın hale gelmekte ve daha sıkı kümeler elde edilmektedir.

### 3.4.2. Algoritma

Düzenli  $S$  aralıklarla yerleştirilen süper piksel merkezleri, sınır bölgeleri üzerinde yer almalarını engellemek için  $3 \times 3$  komşuluktaki en az değişim derecesine sahip noktalara kaydırılmaktadır. Değişim dereceleri 3.4.2 ile hesaplanmaktadır.

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

(3.4.2)

$I(x, y)$  ile  $(x, y)$  piksel pozisyonuna karşılık gelen lab renk vektörü sembolize edilmektedir. Merkez noktaları en düşük değişim derecesine sahip piksel noktalara kaydırıldıktan sonra aşağıdaki adımlar tekrarlanır.

Her bir küme merkezi için  $2S \times 2S$  lik komşuluğunda yer alan pikseller ile arasındaki mesafe 3.4.1 ile hesaplanır. Her merkezin konumu 3.4.1 e göre kendisine en yakın piksellerin konum değerlerinin ortalaması alınarak güncellenir. Bu adım yeni merkez konumlarıyla bir önceki merkez konumları arasındaki fark önceden belirlenen bir değerden küçük oluncaya kadar tekrarlanır. Ayrıca her adımda merkezlerin renk değerleri de kendilerine bağlı piksellerin renk değerlerinin ortalaması ile güncellenir. Bu işlemin sonucunda bazı pikseller ile bağlı oldukları kümeler arasında kopukluk olabilir. Bu kopuk pikseller komşu oldukları en büyük kümeye atanarak problem çözülür.

### 3.4.3. Zaman Karmaşıklığı

Algoritma  $N$  piksel noktasının  $K$  kümeye olan uzaklığını  $I$  yinelemede hesapladığı için karmaşıklığı  $O(NKI)$  dir. Ancak mesafeler sadece merkezlerin  $2S \times 2S$  komşuluklarındaki piksellerle karşılaştırılmaktadır. Bir başka deyişle bir pikselin en fazla kendisine en yakın 8 merkezle arasındaki mesafe hesaplanmaktadır. Bu

durumda  $K=8$ , yani sabittir. Öte yandan yinelemeli adım hızlı bir şekilde yakınsamaktadır ve en fazla  $I=10$  olarak seçilmiştir. Özetle,  $N$  pikselden meydana gelen bir resim sabit sayıda (maks. 80) işlemden geçirilmektedir. Bu durumda algoritmanın karmaşıklığı  $O(N)$  dir.

### 3.5. Diğer Yöntemler

Bir önceki alt başlıklarda bahsedilen yöntemlerin dışında da resim fazla bölümlenmesi için birçok yöntem önerilmiştir. Önerilen yöntemler genel olarak iki başlık altında toplanmıştır: (1)Çizge tabanlı ve (2)yinelemeli yöntemler. Aşağıda bu yöntemler hakkında kısa bilgiler verilmektedir. [9]

**Watershed metodu(2):** 1991 yılında önerilmiştir. [14] Meydana getirdiği süper pikseller oldukça düzensiz ve sınır bölgelerini iyi ayırt edemeyen yapıdadır. Çalışma zamanı  $O(N\log N)$  dir.

**Normalleştirilmiş kesimler(1):** 2000 de önerilmiştir. [15] Düzenli süper pikseller oluştursa da sınır duyarlılığı zayıftır. Ayrıca çalışma zamanı oldukça fazladır. ( $O(N^{3/2})$ ).

**Ortalama kayma algoritması(2):** 2002 de önerilmiştir. [16] Şekil ve boyut açısından düzensiz süper pikseller oluşturur.  $O(N^2)$  çalışma zamanıyla oldukça yavaştır.

**Felzenszwalb & Huttenlocher(1):** 2004 yılında önerilmiştir. [17] Sınır bölgelerine duyarlı süper pikseller oluştursa da meydana gelen süper pikseller şekil ve boyut açısından son derece düzensizdir. Çalışma zamanı  $O(N\log N)$  dir.

**Hızlı kayma algoritması(2):** 2008 de önerilmiştir. [18] Ortalama kayma yöntemine benzerdir. Sınır duyarlılığı iyi olmasına karşın çalışma zamanı fazladır.  $O(kN^2)$ .

3.1 de önerilen yöntem çizge tabanlıdır. Çalışma zamanı  $O(N^{3/2}\log N)$  dir. 3.2 deki yöntem yinelemeli bir yöntemdir ve teorik olarak  $O(N)$  çalışma zamanında olsa da

pratikte yavaş çalışmaktadır. Ayrıca sınır bağılılığı iyi değildir. 3.3 deki yöntem çizge tabanlıdır.  $O(N^2)$  çalışma zamanındadır.

[9] da bu bölümde bahsedilen algoritmaların ayrıntılı karşılaştırılması yapılmıştır ve SLIC algoritmasının gerek hız gerekse de doğruluk olarak diğer algoritmalara göre üstün olduğu açık seçik ifade edilmiştir. Ayrıca bu algoritmanın paralel olarak uygulanmasının olanaklı olması ve birbirine çok yakın boyutlarda süper pikseller üretmesi önerilen sistemde ön aşama olarak kullanılması adına tercih sebebi olmuştur.

## **4. GEODEZİK BÖLÜMLEME**

Bu bölümde, ilgili çalışmalarda (bölüm 2.4) değinilen Geodezik bölümlleme metodu detaylı bir şekilde anlatılmaktadır. Geodezik bölümlleme metodu, etkileşimli bölümlleme için önerilmiş olup tez çalışmasında önerilen metoda dayanak oluşturmaktadır. Geodezik bölümlleme metodu temel olarak iki ana aşamadan meydana gelmektedir. Buna göre öncelikle piksellerin konumsal bilgileri gözetilmeksizin işaretli (sınıflandırılmış) piksellere olan yakınlığı sadece renk bilgisi kullanılarak tahmin edilmiştir. Daha sonra piksellerin sınıflandırılmış piksellere olan geodezik mesafeleri hesaplanarak sınıflandırmaları yapılmıştır. Geodezik mesafe ile konum bilgisi doğal olarak devreye sokulmaktadır.

### **4.1. Olasılık Yoğunluk Tahmini**

Bu alt bölümde anlatılan adımda, sınıflandırılmamış piksellerin bireysel olarak işaretlenmiş piksel sınıflarına (ön ya da arka plan) ait olma olasılıkları tahmin edilmeye çalışılmıştır. Bunun için yalnızca lab (süper piksel çıkarımında da kullanılan) renk uzayındaki renk bilgisi kullanılmıştır. Ancak, renk özellikleri dışında uygulamaya bağlı olarak doku, yoğunluk bilgisi ya da bunların çeşitli birleşimleri de kullanılabilir. Sınıflandırılmış piksellerin bu renk bilgisi kullanılarak olasılık yoğunluk tahmini yapılmıştır. Olasılık yoğunluk tahmini, parametre dışı çekirdek yoğunluk tahmini ile yapılmıştır. Diğer yoğunluk tahmini yöntemleri için [19] incelenebilir. Zaman karmaşıklığını azaltmak için hızlı Gauss dönüşümü kullanılmıştır. [20]

#### **4.1.1. Çekirdek Yoğunluk Tahmini**

Çekirdek yoğunluk tahmini parametre dışı bir yoğunluk tahmin yöntemidir. Parametre dışı yöntemlerin avantajı, yoğunluğun yapısı hakkında bir varsayımda bulunmadan keyfi seçilmiş dağılımları modelleyebilmesidir. Çekirdek yoğunluk tahmini tekniğinde olasılık yoğunluk fonksiyonu 4.1.1 ile tahmin edilir.

$$\hat{f}(x) = \sum_i w_i K\left(\frac{x-x_i}{\sigma}\right) \quad (4.1.1)$$

Burada,  $K$  ile çekirdek fonksiyonu sembolize edilmektedir.  $x_i, i = 1, \dots, n$  örnek (işaretlenmiş, sınıflandırılmış) veri noktalarını,  $w_i$  da bu noktaların ağırlık katsayılarını belirtmektedir. Genel olarak aynı ( $w_i = \frac{1}{n}$ ) ağırlıklar kullanılmaktadır.

Çeşitli çekirdek fonksiyonları mevcuttur ancak, temel alınan ve yapılan çalışmada olduğu gibi en çok Gaussian fonksiyonu tercih edilmektedir.  $\sigma$  ile çekirdek fonksiyonunun bant genişliği sembolize edilmiştir. Bant genişliğinin sonuçlar üzerinde çok fazla etkisi bulunmaktadır. Bu nedenle bu değerin iyi seçilmesi gerekmektedir. [21] de en iyi bant genişliği değerinin nasıl seçileceğine dair geliştiren yöntemlerin incelemesi yapılmıştır. Ek olarak, sabit bir bant genişliği değeri yerine örnek noktaların konumuna göre değişen bant genişlikleri de kullanılabilir.

Ancak baz alınan ve önerilen yöntemde bu değer sabit kullanılmıştır. Çekirdek yoğunluk tahmin yönteminin kullanımı her yeni gelen nokta  $x_j$  için  $\hat{f}(x_j)$  nin hesaplanmasını gerektirir. Resim verisi üzerinden ele alırsak;  $M$  adet kullanıcı tarafından işaretlenmiş piksel olduğunu varsayalım. Buna göre  $N$  pikselden meydana gelen bir resim için bu işlemin karmaşıklığı  $O(MN)$  dir. Başka bir deyişle bu hesaplama resmin  $M$  defa işlenmesini gerektirir. Çabuk işleminin önemli olduğu birçok uygulama için bu karmaşıklık oldukça yüksektir.

Bu işlemin karmaşıklığını azaltmak için hızlı Gauss dönüşümü önerilmiştir. Bu yöntem sayesinde karmaşıklık  $O(M + N)$  ye düşürülmektedir. Bir sonraki başlıkta bu yöntem aktarılmıştır.

#### **4.1.1.1. Renk modellemesi**

Lab renk uzayında, örnek noktalara göre olasılık yoğunluk tahmini aşağıdaki ifadeyle hesaplanmaktadır.

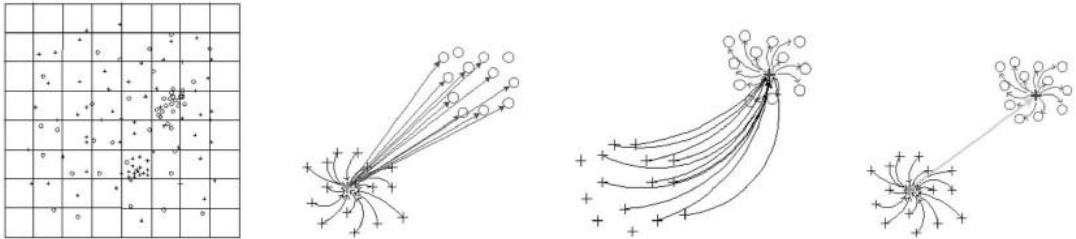


$$\hat{f}(l, a, b) = \frac{1}{M\sigma_l\sigma_a\sigma_b} \sum_{i=1}^M K\left(\frac{l-l_i}{\sigma_l}\right) K\left(\frac{a-a_i}{\sigma_a}\right) K\left(\frac{b-b_i}{\sigma_b}\right) \quad (4.1.2)$$

Renk uzayının her boyutunda, aynı çekirdek fonksiyonu ( $K$ ) farklı bant genişlikleri ( $\sigma_j$ ) ile kullanılmıştır. Her renk boyutunda değişkenlik birbirinden farklı olduğu için, renk boyutuna özel bant genişliği kullanmak makbuldür. Örneğin; genellikle parlaklık boyutu, kromatik boyutlarına göre daha fazla değişkenlik göstermektedir. [22]

#### 4.1.2. Hızlı Gauss Dönüşümü

Bu işlemin temel mantığı kaynak ( $M$ ) ve hedef ( $N$ ) noktalarının kümelenerek, toplamların belirli bir kesinlik derecesinde aynı olan, daha küçük toplamlarla değiştirilmesidir. Algoritma, her Gaussian ı, her hedef noktasında hesaplamak yerine ( $O(MN)$ ), önce her Gaussian ı,  $d$  kaynak küme merkezi Gaussian üzerinde hesaplar ( $O(dN)$ ). Daha sonra, bu merkez Gaussian lar, hedef noktaları üzerinde hesaplanır ( $O(dM)$ ).  $d$  merkez sayısı elde edilmek istenen doğruluk derecesine göre ayarlanabilir. Bu dönüşüm sayesinde  $O(MN)$  karmaşıklığı  $O(M+N)$  ye düşürülmüş olur ( $d$  sabit). Şekil 4.1.1 [22] de bu süreç görselleştirilmiştir.



Şekil 3.3.2. HGD ile kaynak ve hedeflerin kümelenerek eşleştirilmesi

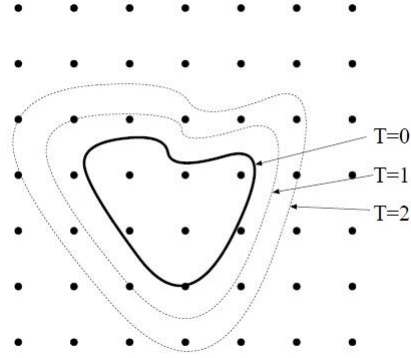
#### 4.2. Geodezik Mesafe

Bu alt bölümde sınıflandırılmış (işaretlenmiş) piksellerle, sınıflandırılmamış pikseller arasındaki geodezik mesafelerin nasıl hesaplandığı anlatılmaktadır. Geodezik mesafenin tanımı bölüm 2.4 de ayrıntılı olarak yapılmıştır. Bu mesafenin

hesaplanabilmesi için Hızlı Yürüyüş Metodu [23] önerilmiştir. Bir sonraki alt bölümde bu metodun detayları yer almaktadır.

#### 4.2.1. Hızlı Yürüyüş Metodu

Hızlı Yürüyüş Metodu, yüzeylerin yayılışını hesaplayan yöntemlerden biridir. HYM de Şekil 4.2.1 [24] de gösterildiği gibi yüzeyin dışa doğru normal yönde zamanla yayıldığı varsayılmaktadır. Hızın her yerde aynı olması şart değildir ancak negatif olmamalıdır. Verilen bir noktada, yüzeyin hareketi Eikonal denklemi olarak bilinen denklem ile verilir:



Şekil 4.2.1. Yüzeyin zaman içerisinde dışarıya doğru yayılması

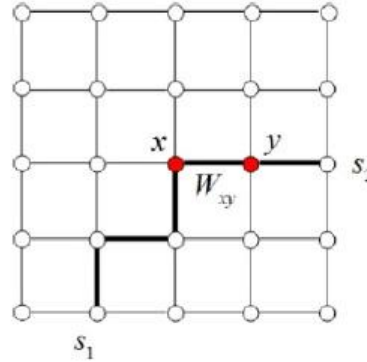
$$|\nabla T(x)|F(x) = 1 \quad (4.2.1)$$

$T$  yüzeyin  $x$  noktasına varış zamanı iken,  $F \geq 0$  yüzeyin  $x$  noktasındaki hızıdır. Hız fonksiyonu belirlidir ve yüzeyin  $x$  noktasıyla ne zaman çakışacağı denklemin çözülmesiyle bulunur. HYM, kafes (ızgara) yapısı üzerinde çalışmaktadır. Bu algoritmaya göre başlangıçta yüzeyi meydana getiren noktalar “değişmez” dir ve bu noktaların komşularına olan mesafeler hesaplanır. Uzaklıkları hesaplanan ve henüz “değişmez” durumda olmayan noktaların “dar bant” içerisinde yer aldığı söylenir. Algoritmanın, her yinelemesinde dar bantta yer alan en düşük uzaklığa sahip nokta “değişmez” duruma getirilir ve bu noktanın komşuları için mesafeler hesaplanır. “değişmez” durumdaki noktalar diğer noktaların değerlerini hesaplamak için kullanılır fakat kendi değerleri tekrar hesaplanmaz. Böylelikle, metotta dar bantta yer

alan noktaların yayılmakta olan yüzeyi oluşturduğunu söyleyebiliriz. Metodun uygulama detayları ile ilgili bilgiye [24] de yer verilmiştir.

### 4.3. Metot

2.4.3 deki ifadeyle, ön plan olasılık yoğunluk fonksiyonunun, ön ve arka plan olasılık yoğunluk fonksiyonlarının toplamına bölümü ile her bir pikselin ön plana bağlı olma olasılığı hesaplanmıştır. Olasılık yoğunluk fonksiyonlarının nasıl tahmin edildiği Bölüm 4.1 de anlatılmıştır. Daha sonra, 2.4.1 deki ifadeyle gösterildiği ve Bölüm 4.3 de de anlatıldığı gibi her bir piksel için ön ve arka plan piksellerine olan geodezik mesafeler hesaplanmıştır. Geodezik mesafelerin hesaplanmasında ayrık formda 4-komşuluk kullanılmıştır. (Şekil 4.3.1 [25]). Bu nedenle 2.4.2 deki sürekli ifade yerine 4.3.1 deki ayrık ifade temsil için daha doğrudur.



Şekil 4.3.1. İki piksel arasındaki 4-komşuluk üzerinden en kısa ağırlıklı yol

$$d(s_1, s_2) := \min_{C_{s_1, s_2}} \sum_{x, y} W_{xy} \quad (4.3.1)$$

$W_{xy}$  kenar ağırlıkları iki komşu pikselin ön plana bağlı olma olasılıklarının mutlak farkı alınarak hesaplanmıştır (4.3.2).

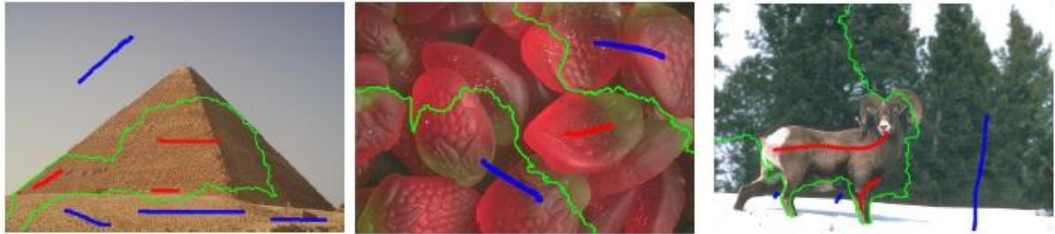
$$W_{xy} = |P_F(x) - P_F(y)|, \quad x, y \in C_{s_1, s_2} \quad (4.3.2)$$

$$\Omega_F = \{x | D_F(x) \leq D_B(x)\} \quad (4.3.3)$$

Son olarak piksellerin ön plan olarak sınıflandırılması 4.3.3 ile yapılmıştır. Yani, bir pikselin ön plan piksellerine olan geodezik mesafesi arka plan piksellerine olan geodezik mesafesinden daha kısa ise ön plan olarak sınıflandırılmıştır.

## 5. GEODEZİK MESAFE TABANLI, İKİLİ BÖLÜMLEME ALGORİTMASI

Geodezik bölümlenme metodunun basit bir renk modellemesi ve geodezik mesafe hesaplamasıyla etkileşimli resim bölümlenmesinde iyi sonuçlar verdiği gözlemlenmiştir. Ancak, geodezik bölümlenme metodunda geodezik mesafelerin sınıflandırmada ayırıcı rol olması için pikseller arasındaki ağırlık değerlerinin önemi çok fazladır. Bu ağırlık değerlerinin renk modellemesiyle hesaplandığını göz önünde bulundurursak geodezik bölümlenme yönteminin başarısının temel olarak renk dağılımlarının iyi modellenmesinden geçtiği söylenebilir. Ne var ki, bu yöntemin kolayca ayırt edilebilen renk dağılımlarında iyi, aksi durumda kötü çalıştığı gözlemlenmiştir. (Şekil 5.1 [8]). Önerilen yöntemde renk dağılımlarını daha iyi modelleyebilen uzaysal yakınlığı da göz önünde bulunduran yoğunluk tahmin metodu devreye sokulmuştur. Geodezik bölümlenme metodunun diğer bir eksikliği de Bölüm 2.7 de belirtildiği üzere işaretlenmiş piksel yerlerine bağlı sonuçlar vermesidir. Bu problem Bölüm 2.7 de önerilen yöntemle giderilmeye çalışılmıştır.



Şekil 5.1.0. Geodezik bölümlenme metodunun iyi çalışmadığı örnekler

Tez çalışmasında önerilen yöntemde, arka plan işaretlemesinin çıkartılmak istenen nesnenin etrafını saracak şekilde yapılması ön görüldüğünden problemin bu kısıtla otomatik olarak aşılması planlanmıştır. Ayrıca, orijinal yöntemde yoğunluk tahmininin masraflı olması nedeniyle bu aşamayı hızlandırıcı bir yöntem (hızlı Gauss dönüşümü) kullanılmıştır. Önerilen yöntemde ise yoğunluk tahmini ve geodezik mesafe hesaplama aşamalarının hızlandırılması için piksel sayısını büyük oranda azaltan süper piksel gruplama yöntemi ön aşama olarak kullanılmıştır. Özetle, önerilen yöntem aşağıdaki adımlardan oluşmaktadır:

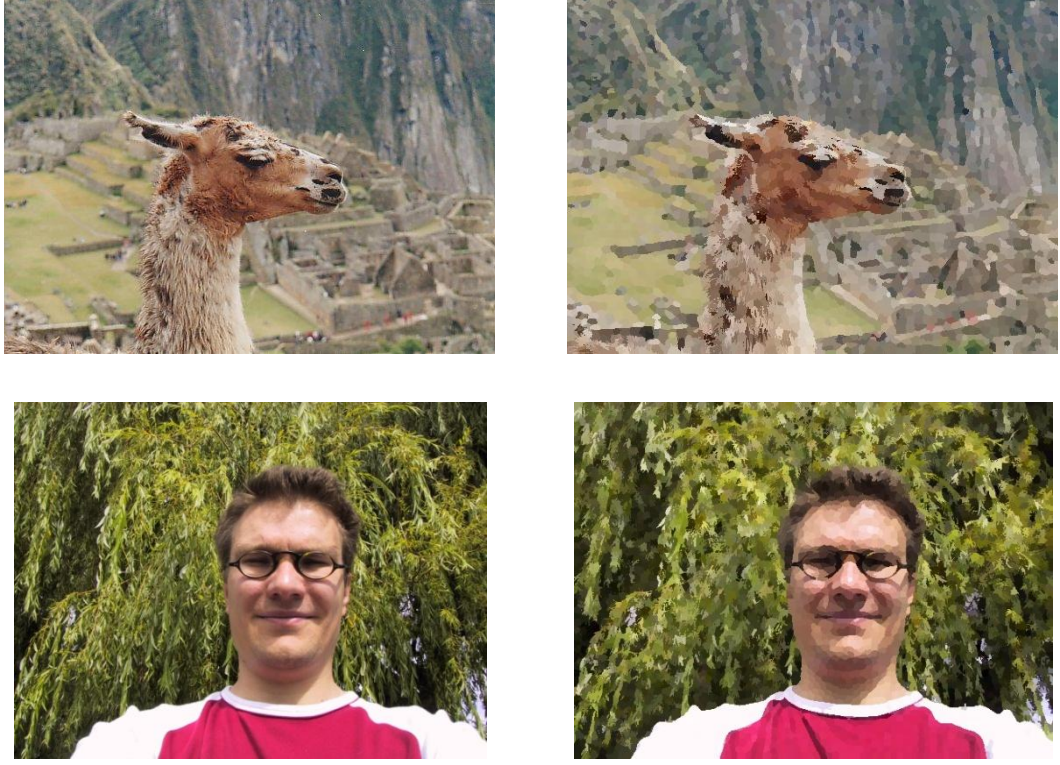
- 1) SLIC yöntemiyle resmin süper piksellere bölünmesi
- 2) Uzaysal olarak değişken renk dağılımları ile ön ve arka planların modellenerek olasılıkların hesaplanması
- 3) Geodezik mesafelerin hesaplanması
- 4) Büyük oranda ön ve arka plana ait oldukları kesinleşen süper piksellerin sınıflandırılması, aitliği kesin olmayan piksellerin, yerel olarak renk dağılımlarının incelenerek sınıflandırılmaları

Bir sonraki alt bölümde metodun detayları incelenmiştir.

### 5.1. Yöntem

**Süper piksel bölümlenmesi:** Bu adımda Bölüm 3.4 de anlatılan yöntem kullanılmıştır. Yapılan çalışmada, iki süper piksel merkezi arasındaki uzaklık  $S=8$  olarak alınmıştır. Konumsal uzaklık, renksel uzaklığa göre 5 e 1 oranında alınmıştır. Ayrıca yineleme sayısı, süper piksel merkez değişim oranı süper piksel sayısının 32 de 1 inden az oluncaya kadar şeklinde belirlenmiştir. Buna göre yineleme ortalama 5-6 adım sürmektedir. Şekil 5.1.1 de SLIC süper piksel bölümlenmesi ile elde edilen örnek resimlere yer verilmiştir.



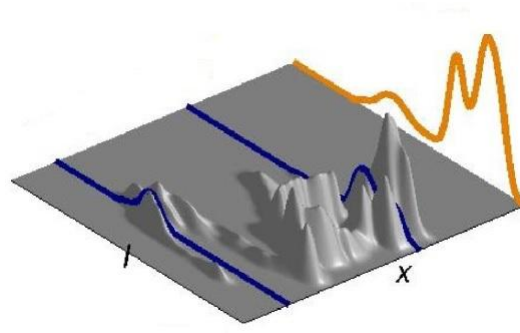


Şekil 5.1.1. Süper piksel bölümlenmesi örnekleri

**Konumsal olarak değişken yoğunluk tahmini:** [26] de çok sınıflı bölümlenme için önerilen konumsal olarak değişken yoğunluk tahmini ile olasılık yoğunluk fonksiyonlarının piksellerin konum bilgisi de hesaba katılarak hesaplanması sonucu, sınıfları daha ayırıcı bilgiler elde edildiği örneklerle gösterilmiştir. Burada da geodezik bölümlenme metodunda olduğu gibi çekirdek yoğunluk tahmin metodu kullanılmış, ancak 4.1.1 deki gibi sadece renk dağılımları için değil, 5.1.1 ile de ifade edildiği gibi piksel konumları için de ayrı bir çekirdek fonksiyonu kullanılarak, oyf tahmini yapılmıştır.

$$\hat{f}(l, a, b, x, y) = \frac{1}{M\sigma_l\sigma_a\sigma_b\sigma_x\sigma_y} \sum_{i=1}^M K\left(\frac{l-l_i}{\sigma_l}\right) K\left(\frac{a-a_i}{\sigma_a}\right) K\left(\frac{b-b_i}{\sigma_b}\right) K\left(\frac{x-x_i}{\sigma_x}\right) K\left(\frac{y-y_i}{\sigma_y}\right) \quad (5.1.1)$$

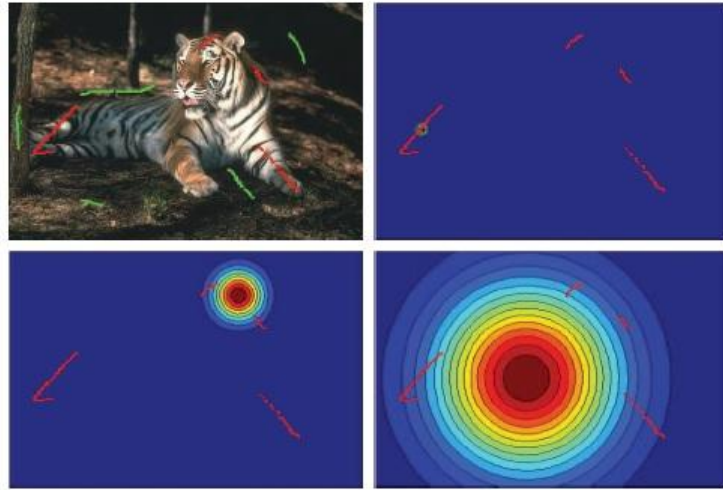
Bu hesaplama göre aynı renk değerine sahip pikseller dâhil, tüm pikseller için farklı yoğunluk fonksiyonları elde edilmiştir. (Şekil 5.1.2 [26]).



Şekil 5.1.2. Pikel konumlarına göre farklı olasılık yoğunluk fonksiyonlarının elde edilmesi

Ancak uzaysal modelleme için, renk modellemesinde olduğu gibi sabit değerli  $\sigma_x = \sigma_y$  bant genişliği seçmek mümkün değildir. Çünkü renk örnekleri birbirinden bağımsız olmasına karşın, konumsal örnekler değildir. Bu düzensizliği hesaba katmak için uyarlanabilir bant genişliği kullanılması önerilmiştir. Buna göre, bir pikselin, her bir sınıf için ayrı, en yakın olduğu örnek noktası belirlenmiş ve bu örnek noktasına olan mesafeye göre bant genişliği seçilmiştir (5.1.2).

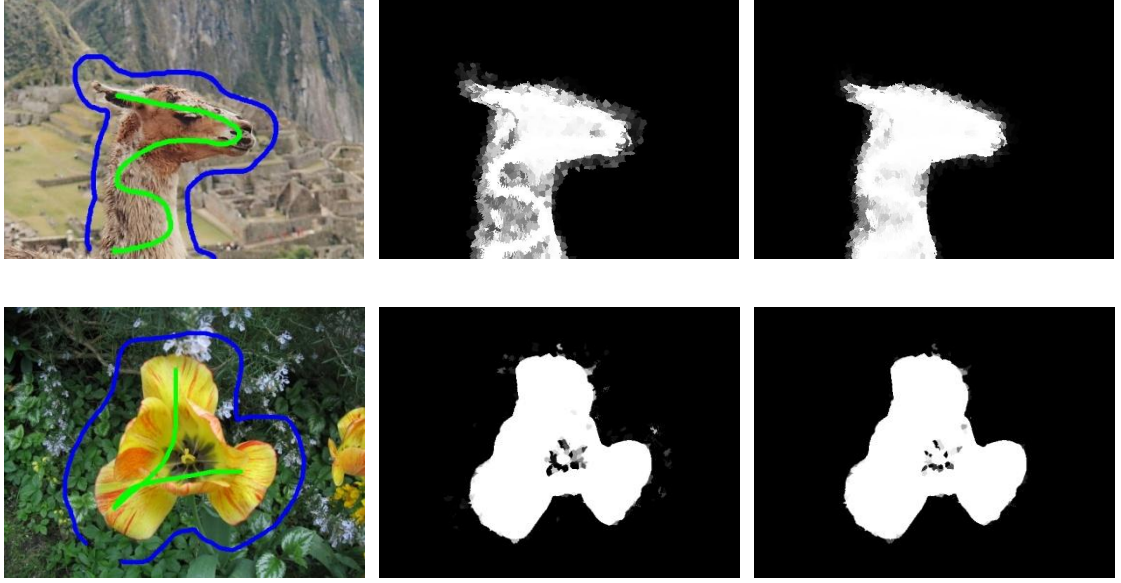
$$\sigma_{s_i}(p) = \alpha |p - p_i|_2 \quad (5.1.2)$$



Şekil 5.1.3. Uzaysal değişken bant genişliği kullanmanın etkisi

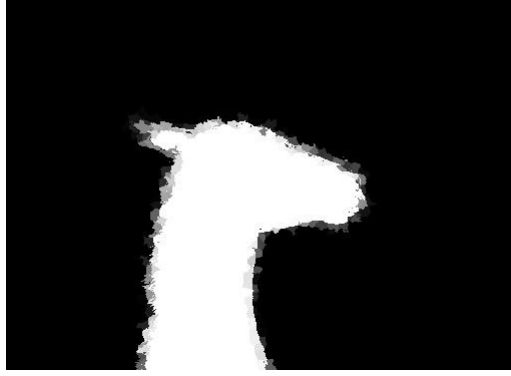


Piksel konumuna göre deęişken bant genişlięi kullanılmasının etkisi Şekil 5.1.3 [26] de gösterilmiştir. Konumsal olarak deęişken oyf tahmininin, sabit oyf tahminini ile karşılaştırılması, piksellerin ön plana baęlı olma olasılıklarının 2.4.3 ile hesaplanması sonucu elde deęerlerin  $[0,255]$  arası normalleştirilmesiyle Şekil 5.1.4 de gösterilmiştir. Buradan uzaysal olarak deęişken oyf tahmininin etkisi görülebilir.



Şekil 5.1.4. Konumsal olarak sabit (orta) ve deęişken oyf tahmini sonuçları

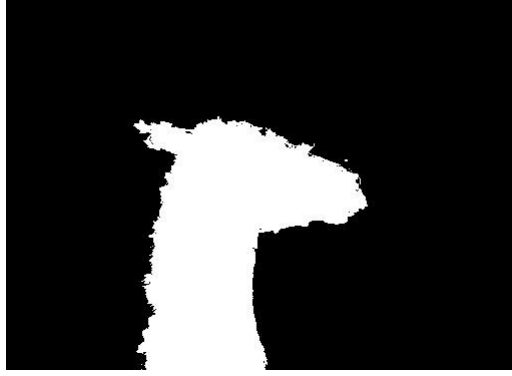
**Geodezik mesafe hesaplaması:** Geodezik mesafeler Bölüm 4.2 de anlatılan yöntemle bulunmuştur. Mesafelerin hesaplanması için gerekli olan pikseller (süper) arasındaki kenar ağırlıkları, piksellerin ön plana ait olma olasılıklarının bir önceki alt başlıkta anlatıldığı gibi hesaplanmasıyla bulunmuştur. Bir süper pikselin ön plan sınıfına olan geodezik mesafesinin, ön ve arka plan sınıflarına olan mesafelerinin toplamına bölünmesi sonucu elde edilen ön plan sınıfına ait olasılık deęerleri ( $[0,255]$  arası normalleştirilmesiyle) Şekil 5.1.5 de gösterilmiştir.



Şekil 5.1.5. Geodezik mesafe hesaplaması sonucu elde edilen ön plan olasılık değerleri

$$P_F(x) = \frac{D_F(x)}{D_F(x)+D_B(x)} \quad (5.1.3)$$

**Piksel düzeyinde sınıflandırma:** Yukarıdaki adımların uygulanması neticesinde 4.3.3 de de belirtildiği gibi ön plan sınıfına arka plan sınıfından, geodezik mesafe bakımından, daha yakın olan piksellerin ön plan olarak sınıflandırılması sonucu, özellikle nesnelerin sınır bölgelerinde tutarsızlıklar olduğu gözlemlenmiştir. (Ön ve arka plan sınıflandırmalarının sınır bölgelerinde genellikle yanlış olduğu gözlenmiş). (Şekil 5.1.6). Ayrıca burada süper pikseller ile sadece merkezde yer alan piksellerin renk değerleri kullanıldığı için bu piksellerin yanlış sınıflandırılmasının bu piksellere bağlı diğer piksellerin de yanlış sınıflandırılmasına neden olduğu saptanmıştır. Ek olarak, doğru sınıflandırılan piksellerin çok büyük olasılıklarla atandıkları sınıflara bağlı olduğu, yanlış sınıflandırılanların ise bu derece yüksek olasılıklar taşımadıkları gözlemlenmiştir. Bu nedenlerle öncelikle belirli bir güvenilirlik (olasılık) değerinin üstünde/altında kalan pikseller için sınıf atamaları gerçekleştirilmiştir.



Şekil 5.1.6. Süper piksel düzeyinde geodezik mesafe hesaplaması sonucu elde edilen bölümlenme

$$\Omega_F = \{x|P_F(x) > T_F\} \text{ ve } \Omega_B = \{x|P_F(x) < T_B\} \quad (5.1.4)$$

$T_F = 0.9$  ve  $T_B = 0.01$  in iyi çalıştığı gözlemlenmiştir. Buna göre ön plana ait olma olasılığı %90 dan daha yüksek olan pikseller ön plan olarak, %0.01 den de daha az olanlar arka plan olarak sınıflandırılmıştır. Geri kalan piksellerin sınıflandırılması için aşağıda anlatılan işlem önerilmiştir.



Şekil 5.1.7. Sınıf değerleri kesin olmayan (kırmızı) piksellerin gösterimi

Şekil 5.1.7 de atanan ve atanamayan pikseller gösterilmektedir. Atanamayan piksellerin, önceden sınıfı belli olan (kullanıcı tarafından verilen) ve daha sonra sınıf atamaları gerçekleştirilen piksellerin yardımıyla atamaları yapılmıştır. Bunun için, atanamayan her piksele konum olarak en yakın olan ön ve arka plan piksel mesafeleri hesaplanmıştır. Bu piksele, bu mesafelerin belirli bir katı ( $\alpha = 5.4$ ) kadar yakın olan (5.1.5) sınıflandırılmış pikseller örnek olarak alınmış ve renksel oy tahminleri

yapılmıştır. Hesaplanan oyf lere göre geri kalan pikseller de ön ya da arka plan olarak sınıflandırılmıştır. Son adımda pikseller bireysel olarak renk özelliklerine göre sınıflandırıldığından konumsal bütünlük ortadan kalkmaktadır. Ancak, son adım birbirlerine yakın pikseller üzerinde çalıştığı için bu bütünlük basit bir şekilde Gaussian filtrelemesi [27] (başıboş pikselleri temizlemek için) ve ardından kapama işlemi [27] (boşlukları doldurmak için) uygulanmasıyla sağlanmıştır. (Şekil 5.1.8).



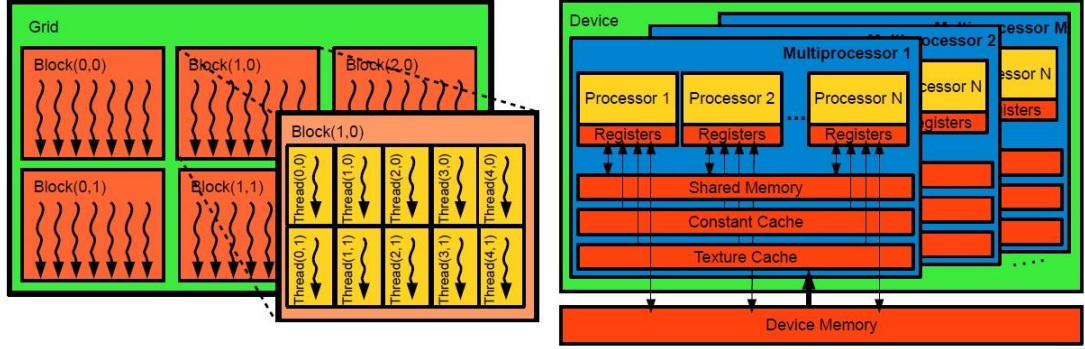
Şekil 5.1.8. Piksel düzeyinde bölümlenme ve morfolojik iyileştirme

$$\alpha|p - p_i|_2 \quad (5.1.5)$$

## 5.2. Uygulama ve Çalışma Zamanları

### 5.2.1. Uygulama

Uygulama C, C++ ve CUDA C programlama dillerinde, Visual Studio 2010 ortamında Windows 8 platformu üzerinde geliştirilmiştir. Tüm sonuçlar CUDA işlem kapasitesi 3.0 olan ve i7 2.3 GHz işlemcisine sahip bilgisayar üzerinde yapılmıştır. Süper piksel bölümlenmesi, [28] den yararlanılarak CUDA C dilinde GPU üzerinde paralel hesaplama yapılmasıyla gerçekleştirilmiştir. CUDA C dili ve GPU üzerinde paralel programlama ile ilgili detaylı bilgiye [29] den erişilebilir. Şekil 5.2.1 [28] de CUDA nın iş parçacığı ve hafıza modeline yer verilmiştir. Olasılık yoğunluk tahmini, geodezik mesafe hesaplaması ve diğer hesaplamalar CPU üzerinde C ve C++ programlama dilleri kullanılarak yapılmıştır.



Şekil 5.2.1. CUDA iş parçacığı ve hafıza modeli

### 5.2.2. Çalışma Zamanları

Yapılan benzer çalışmaların hepsinde çalışma sürelerine yer verilmediği için, geliştirilen uygulamaya ait çalışma zamanlarının deneyler bölümü yerine bu bölümde yer almasının daha uygun olacağına karar verilmiştir. Ayrıca elde edilen çalışma sürelerinin üzerinde hesaplama yaptırılan makineye göre değişiklik arz ettiğini göz önünde bulundurursak makalelerde belirtilen çalışma süreleri üzerinden karşılaştırma yapmak adil olmayacaktır. Önerilen yöntemin işlem adımlarına göre çalışma süreleri Tablo 1 de yer almaktadır.

Tablo 5.1 Uygulamanın ortalama çalışma süreleri

İşlem	Uygulama	Süre
Süper piksel çıkarımı	GPU/CUDA C	22,84ms.
Yoğunluk tahmini	CPU/C,C++	32,88ms.
Geodezik mesafe	CPU/C,C++	18,08ms.
Piksel düzeyinde iyileştirme	CPU/C,C++	363,68ms.

## 6. DENEYLER

Etkileşimli bölümlene yöntemlerinin başarısının karşılaştırılabilmesi için 50 resimden oluşan Grabcut [30] veri seti kullanılmaktadır. (Şekil 6.1). Karşılaştırmalar hata oranları üzerinden yapılmaktadır. Hata oranı aşağıdaki formülle hesaplanmaktadır.



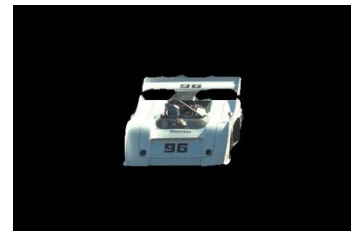
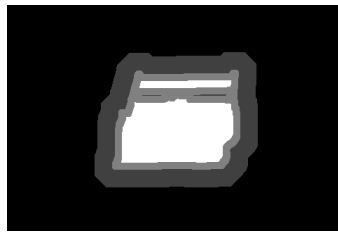
Şekil 6.1.0. Grabcut veri kümesi

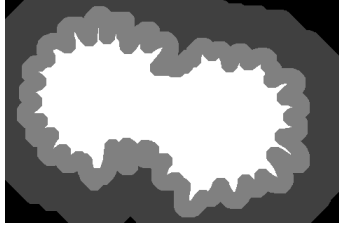
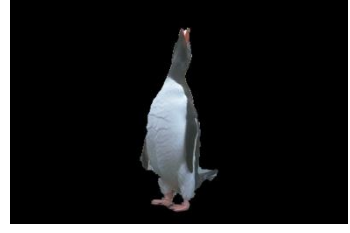
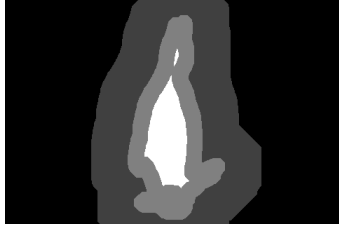
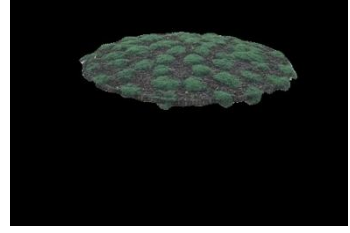
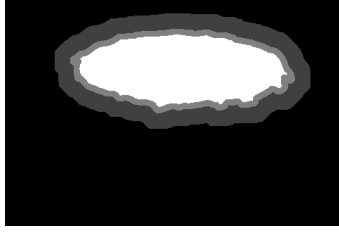
$$\varepsilon = \frac{|yanlış sınıflandırılmış pikseller|}{|sınıflandırılmamış pikseller|} \quad (6.1)$$

Bölüm 2 de anlatılan ilgili çalışmaların ve önerilen yöntemin bu hesaplamayla elde edilen hata oranlarına Tablo 2 de yer verilmiştir. UE ile uyarlanabilir eşikleme ifade edilmektedir. Uyarlanabilir eşikleme [31] de önerilen bir yöntemdir. Buna göre ön plana ait olma olasılıkları ( $p$ ) hesaplanan piksellerden kesin olarak ön veya arka plana ait olmayanlarının ( $0 < p < 1$ ) etrafında, en az bir ön ( $p=1$ ) ve bir arka ( $p=0$ ) plana ait olduğu belirlenen piksel yer alacak şekilde minimum alana sahip dikdörtgensel bir pencere oturtulur. Bu pencere içinde yer alan ve  $0 < p < 1$  olan piksellerin olasılıklarının ortalaması alınır. Eğer pikselin olasılık değeri bu ortalama olasılık değerinden yüksek veya eşit ise ön, değilse arka plan olarak sınıflandırılır. Ancak, bu yöntemin sınıflandırılmamış piksel bandının ortasına doğru sınırları yerleştirme eğiliminde olduğu ve “lasso” işaretlemesinde sanal bir avantaj sağladığı [8] de belirtilmiştir.

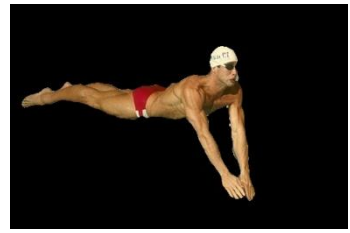
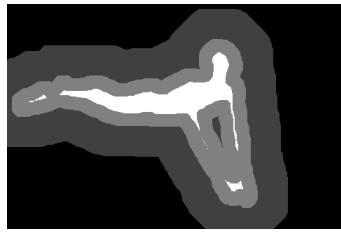
Tablo 6.1. Bölümleme hata oranları

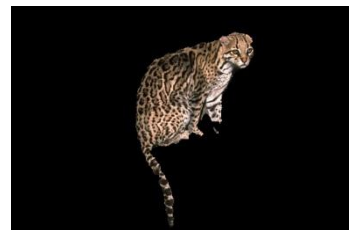
Yöntem	Hata Oranı
Çizge Kesim(2001) [1]	%6.7
GMMRF(2004) [3]	%7.9
Rastgele Yürüyüşler(2006) [4]	%5.4
RY + UE(2006) [4]	<b>%3.3</b>
Geodezik(2007) [5]	%6.8
İletimsel Bölümleme(2008) [6]	%5.4
İB + UE(2008) [6]	<b>%3.3</b>
Kısıtlı Rastgele Yürüyüşler(2010) [7]	<b>%4.1</b>
Geodezik Çizge Kesim(2010) [8]	%4.8
Önerilen Metot	<b>%4.1</b>

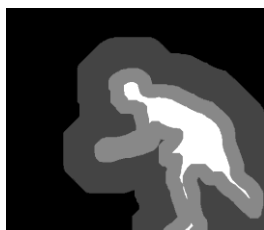
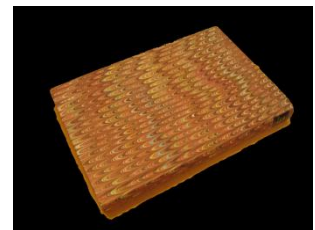
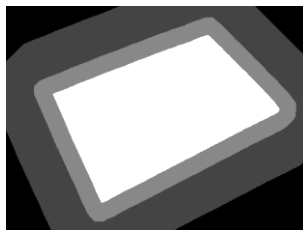


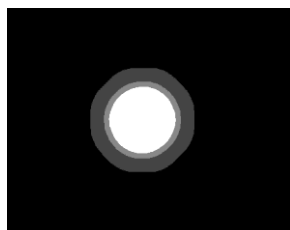
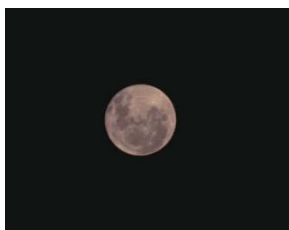




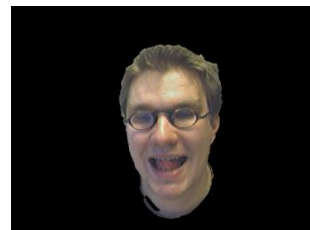


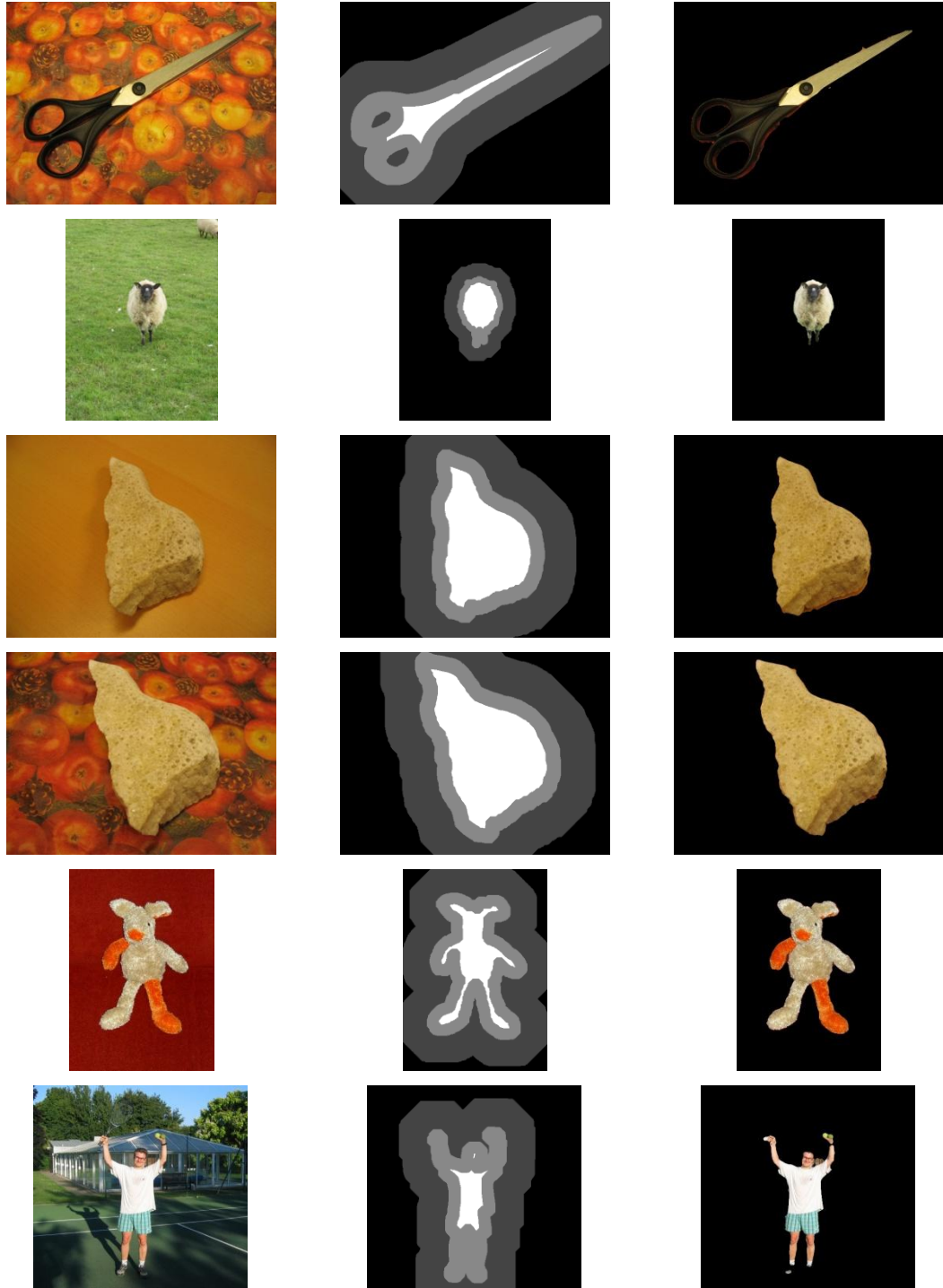






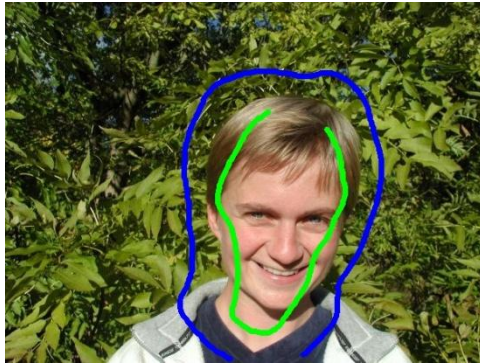






Şekil 6.2.0. Grabcut veri kümesi üzerinde alınan sonuçlar

Şekil 6.2 de “lasso” işaretlemesi kullanılarak elde edilen ve veri kümesindeki tüm resimleri içeren bölümlene sonuçları gösterilmektedir. Şekil 6.3 de diğer bir işaretleme türüne ait birkaç sonuca yer verilmiştir.



Şekil 6.3.0. Farklı işaretleme türüyle elde edilen sonuçlar



## 6.1. Değerlendirme

Önerilen yöntemin ve diğer ilgili yöntemlerin hata oranlarına baktığımız zaman, önerilen yöntemin özellikle son yıllarda önerilen diğer yöntemler kadar iyi çalıştığı gözlemlenmiştir. Önerilen yöntem ile piksellerin konumsal özelliklerinin de devreye sokularak geodezik bölümlene metodunun sınıflandırma başarısı büyük oranda geliştirilmiştir. Ayrıca, algoritmaların bir kısmının paralel olarak uygulanması ve ön aşama adımı (piksellerin gruplandırılması) sonucunda, sistemin toplam çalışma zamanının da anlık etkileşime olanak tanıyacak seviyede olduğu söylenebilir. Bunun yanı sıra önerilen yöntemin çizgisel işaretleme yönteminde, işaretlemenin konumuna ve sayısına bağlı sonuçlar verdiğini söyleyebiliriz. Bu problem diğer yöntemlerde de göze çarpmakta ve arka planı karmaşık olan resimlerin sonuçlarda yer verilmediğini görülmektedir, örneğin “lama” resmi gibi. Son olarak, önerilen yöntemde morfolojik işlemlerin yapılması neticesinde sınır bölgelerinde bazı taşmalar meydana geldiği görülmüştür.

## 7. SONUÇ

Etkileşimli resim bölümlenmesinin özellikle medikal görüntü işlemede ve resim düzenleme araçlarında kullanımının büyük önemi vardır. Bu nedenle özellikle 2000 lerin başından itibaren dijital araçların yaygınlaşmasını da takip eden süreçten itibaren etkileşimli resim bölümleme konusu üzerinde önemli çalışmalar yapılmıştır. Yapılan bu çalışmalar neticesinde geodezik bölümleme ve çizge kesim metodlarının bu konu üzerinde etkili yöntemler oldukları açığa çıkmıştır. Geodezik bölümleme metodunun “işaretleme konumu” ve çizge kesim metodunun “erken kesim” problemlerine çözüm amacıyla bir başka çalışmada geodezik çizge kesim metodu önerilmiştir. Bu tez çalışmasında ise geodezik bölümleme metodunun arka planı karmaşık resimlerde de iyi çalışmasını sağlayabilmek için bir yöntem önerilmiştir. Önerilen yöntemin orijinal geodezik bölümleme metodunun sonuçlarını ne kadar geliştirdiği hata oranı ölçütü ile deneyler bölümünde verilmiştir. Bu ölçüte göre önerilen yöntem diğer yöntemlere yakın ve iyi sonuçlar vermiştir. Ayrıca önerilen yöntemin çalışma zamanı da paralel işleme ve ön aşama adımları neticesinde anlık etkileşimli bölümlemeye olanak tanımaktadır.

Yapılan bu çalışmanın ileride doğruluk ve çalışma zamanı açısından geliştirilmesi hedeflenmektedir. Çalışma zamanının daha da iyileştirilmesi için CPU üzerinde sırasal uygulanan adımların da GPU üzerinde paralel olarak uygulanması hedeflenmektedir. Ayrıca morfolojik işlemler sonucu oluşan hataları ortadan kaldırmak için konumsal bütünlüğün otomatik olarak sağlandığı bir piksel düzeyinde sınıflandırma aşaması üzerinde çalışılması planlanmaktadır. Son olarak çizgisel işaretlemeyle ilgili bölümlenmenin de nasıl geliştirilebileceği üzerinde çalışılması düşünülmektedir.

## KAYNAKLAR

- [1] Boykov, Yuri Y., and M-P. Jolly. "Interactive graph cuts for optimal boundary & region segmentation of objects in ND images." *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*. Vol. 1. IEEE, 2001.
- [2] Boykov, Yuri, and Vladimir Kolmogorov. "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 26.9 (2004): 1124-1137.
- [3] Blake, Andrew, et al. "Interactive image segmentation using an adaptive GMMRF model." *Computer Vision-ECCV 2004*. Springer Berlin Heidelberg, 2004. 428-441.
- [4] Grady, Leo. "Random walks for image segmentation." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 28.11 (2006): 1768-1783.
- [5] Bai, Xue, and Guillermo Sapiro. "A geodesic framework for fast interactive image and video segmentation and matting." *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. IEEE, 2007.
- [6] Duchenne, Olivier, et al. "Segmentation by transduction." *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008.
- [7] Yang, Wenxian, et al. "User-friendly interactive image segmentation through unified combinatorial user inputs." *Image Processing, IEEE Transactions on* 19.9 (2010): 2470-2479.
- [8] Price, Brian L., Bryan Morse, and Scott Cohen. "Geodesic graph cut for interactive image segmentation." *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010.
- [9] Achanta, Radhakrishna, et al. "SLIC superpixels compared to state-of-the-art superpixel methods." (2012): 1-1.
- [10] Moore, Alastair P., et al. "Superpixel lattices." *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008.
- [11] Levinshtein, Alex, et al. "Turbopixels: Fast superpixels using geometric flows." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 31.12 (2009): 2290-2297.

- [12] Veksler, Olga, Yuri Boykov, and Paria Mehrani. "Superpixels and supervoxels in an energy optimization framework." *Computer Vision–ECCV 2010*. Springer Berlin Heidelberg, 2010. 211-224.
- [13] Achanta, Radhakrishna, et al. "Slic superpixels." *École Polytechnique Fédéral de Laussanne (EPFL), Tech. Rep 149300* (2010).
- [14] Vincent, Luc, and Pierre Soille. "Watersheds in digital spaces: an efficient algorithm based on immersion simulations." *IEEE transactions on pattern analysis and machine intelligence* 13.6 (1991): 583-598.
- [15] Shi, Jianbo, and Jitendra Malik. "Normalized cuts and image segmentation." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 22.8 (2000): 888-905.
- [16] Comaniciu, Dorin, and Peter Meer. "Mean shift: A robust approach toward feature space analysis." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 24.5 (2002): 603-619.
- [17] Felzenszwalb, Pedro F., and Daniel P. Huttenlocher. "Efficient graph-based image segmentation." *International Journal of Computer Vision* 59.2 (2004): 167-181.
- [18] Vedaldi, Andrea, and Stefano Soatto. "Quick shift and kernel methods for mode seeking." *Computer Vision–ECCV 2008*. Springer Berlin Heidelberg, 2008. 705-718.
- [19] Webb, Andrew R. *Statistical pattern recognition*. Wiley. com, 2003.
- [20] Greengard, Leslie, and John Strain. "The fast Gauss transform." *SIAM Journal on Scientific and Statistical Computing* 12.1 (1991): 79-94.
- [21] Jones, M. Chris, James S. Marron, and Simon J. Sheather. "A brief survey of bandwidth selection for density estimation." *Journal of the American Statistical Association* 91.433 (1996): 401-407.
- [22] Elgammal, Ahmed, Ramani Duraiswami, and Larry S. Davis. "Efficient kernel density estimation using the fast gauss transform with applications to color modeling and tracking." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 25.11 (2003): 1499-1504.
- [23] Sethian, J. A. "Evolution, implementation, and application of level set and fast marching methods for advancing fronts." *Journal of Computational Physics* 169.2 (2001): 503-555.
- [24] Bærentzen, Jakob Andreas. *On the implementation of fast marching methods for 3D lattices*. 2001.

- [25] Bai, Xue, and Guillermo Sapiro. "Geodesic matting: A framework for fast interactive image and video segmentation and matting." *International Journal of Computer Vision* 82.2 (2009): 113-132.
- [26] Nieuwenhuis, Claudia, and Daniel Cremers. "Spatially Varying Color Distributions for Interactive Multilabel Segmentation." (2012): 1-1.
- [27] Dougherty, Edward R., and Roberto A. Lotufo. "Hands-on morphological image processing." Bellingham: Spie, 2003.
- [28] Ren, Carl Yuheng, and Ian Reid. "gSLIC: a real-time implementation of SLIC superpixel segmentation." *University of Oxford, Department of Engineering, Technical Report* (2011).
- [29] Nvidia, C. U. D. A. "NVIDIA CUDA programming guide." (2011).
- [30] Grabcut dataset: <http://tinyurl.com/grabcut>
- [31] Guan, Jian, and Guoping Qiu. "Interactive image segmentation using optimization with statistical priors." *International Workshop on The Representation and Use of Prior Knowledge in Vision, In conjunction with ECCV*. 2006.

## ÖZGEÇMİŞ

### Kişisel Bilgiler

Soyadı, Adı : ÖZTÜRK, Orkun  
Uyruğu : T.C.  
Doğum tarihi ve yeri : 13.10.1988 Isparta  
Medeni hali : Bekâr  
Telefon : 0 (505) 317 85 07  
E-posta : oozturk@etu.edu.tr

### Eğitim

Derece	Eğitim Yeri	Mezuniyet Tarihi
Lisans	TOBB ETÜ Bilgisayar Mühendisliği	2011

### İş Deneyimi

İş Deneyimi	Görev	
2010-2011	TOBB ETÜ	Stajyer
2009-2010	Akgün Yazılım	Stajyer
2008-2009	TOBB ETÜ	Stajyer

### Yabancı Dil

İngilizce