

Süperpiksel Algoritmalarının Gürültülü İmgeler İçin Bölütleme Performansının İncelenmesi

Investigation of Segmentation Performances of Superpixel Algorithms for Noisy Images

Fadime ÖZER¹, Ufuk ÖZKAYA¹

¹Elektronik Haberleşme Mühendisliği Bölümü, Süleyman Demirel Üniversitesi, Isparta, Türkiye
fadimeozzer.1992@gmail.com, ufukozkaya@sdu.edu.tr

Özetçe—Süperpikseller, özellikle imge ve video bölütleme uygulamalarında kullanılan; bir imgeye veya videoya ait benzer renk, yoğunluk ve doku özellikleri benzerlik gösteren piksellerden oluşan görsel yapılardır. Bu çalışmada süperpiksel algoritmalarının gürültülü imgeler için bölütleme performansı incelenmiştir. Ayrıca, bölütleme uygulamalarında filtre kullanımının bölütleme performansına etkisi de incelenmiştir. Bu amaçla, en yaygın kullanılan üç süperpiksel algoritması imge bölütleme veri setinden seçilen imgelere uygulanmıştır. Gerçekleştirilen uygulamalar sonucunda, süperpiksel algoritmalarının gürültü bağısıklığı ve bölütleme performansları incelenmiştir. Ayrıca, ön işlem olarak filtre kullanımının bölütleme performansına olan etkisi incelenmiştir.

Anahtar Kelimeler—süperpiksel, imge bölütleme, gürültü.

Abstract—Superpixels, used espically in image and video segmentation applications, are the visual structures composed of pixels having same color, intensity and texture behavior. In this work, segmentation performances of superpixel algorithms for noisy images are investigated. Moreover, the effect of denoising to segmentation accuracy is also investigated. For this purpose, most common three superpixel algorithms are implemented to the images selected from image segmentation database. As a result of efforts made, noise sensitivity and segmentation performance of superpixel algorithms are investigated. Also, the effect of denoising process as a preprocessing step to the segmentation performance is also examined.

Keywords—superpixel, image segmentation, noise.

I. GİRİŞ

Süperpikseller, bir imgeye veya videoya ait renk, yoğunluk ve doku özellikleri benzerlik gösteren piksellerden oluşan görsel yapılardır. Süperpikselin sahip olduğu bu özellik bir imgenin veya videonun daha az veri ile ifade edilebilmesini mümkün kılmaktadır. İmge işleme uygulamalarında süperpiksel kullanılmasının başlıca amacı işlem yükünü ve işlem süresini oldukça azaltmaktır. Bu yüzden süperpiksel algoritmaları nesne tanıma [1], bölütleme [2], vücut modeli tahmini [3], kenar tespiti [4] gibi uygulamalarda ön işlem adımı olarak sıklıkla kullanılmaktadır.

Literatürde birçok süperpiksel algoritması bulunmaktadır. Süperpiksel algoritmalarında genellikle şekilsel ve boyutsal düzenlilik, homojenlik gibi kriterler dikkate alınmaktadır. Literatürde bulunan çalışmalarda önerilen süperpiksel algoritmalarının performansları sadece gürültü içermeyen imgeler için değerlendirilmiş ve burada genellikle düzenli ve/veya homojen yapıların elde edilebilmesi amaçlanmıştır. Fakat bilgisayarla görü, sinyal, imge ve video işleme alanında, veri toplama ve aktarımı sırasında gürültü ne yazık ki kaçınılmazdır. Ayrıca imge bölütlemenin en yoğun kullanıldığı alanlardan biri de biyomedikal imge bölütlemesidir. Biyomedikal cihazlardan alınan imgelerin gürültü içermesi nedeniyle başarılı bir bölütleme işlemi için ya ön işlem olarak filtrelerin kullanılması veya gürültü bağısıklığı yüksek olan bölütleme yöntemlerinin kullanılması gerekmektedir.

Literatürde gürültü içeren imgelerin bölütlenmesi amacıyla gerçekleştirilen çalışmalara örnek olarak gürültüye dayanıklı FCM (Fuzzy C-Means) tabanlı bir algoritma verilebilir [5]. Çalışmada imgeye ait bölütler hem benzerlik ölçüsüne hem de üyelik fonksiyonuna bağlı olarak iteratif bir şekilde oluşturulmaktadır. Bir diğer çalışmada imge bölütleme problemlerinde en çok kullanılan süperpiksel algoritmalarının gürültüsüz ve gürültülü geniş bir imge havuzu için performansları incelenmiştir [6]. Bu çalışmada algoritmaların performansları karşılaştırılırken gauss gürültüsü, tuz&biber gürültüsü ve bulanıklığın etkileri incelenmiştir. Diğer bir çalışmada ise mikroskopik görüntülerde farklı gürültü türlerinin süperpiksel algoritmalarının performansındaki değişim incelenmiştir [7]. Süperpiksel algoritmalarının gürültülü verilerde sınır bilgisini kaybetmesi problemi gürültü filtreleme işlemi ile iyileştirilmiştir. Diğer taraftan süperpiksel algoritmalarının diğer bir kullanım alanı ise gürültü tahminidir [8,9]. İmgedeki gürültü seviyesini kestirmek oldukça zor olduğundan imge daha küçük parçalara yani süperpiksellere ayrılarak her bir süperpikselin gürültü tahmini yapılarak daha doğru sonuçlar elde edilmektedir.

Bu çalışmada gürültünün süperpiksel algoritmalarının performanslarını nasıl etkilediği ve ön işlem olarak gürültü giderici filtreler kullanıldığında bölütleme performansının

hangi oranda iyileştirileceği incelenmiştir. Literatürde en yaygın olarak kullanılan ve en başarılı süperpiksel algoritmalarından olan Simple Linear Iterative Clustering (SLIC) [10], Turbopixel (TP) [11] ve Regularity Preserved Superpixels (RPS) [12] algoritmaları seçilmiştir. Bu algoritmalar sırasıyla orijinal imgelere, gürültülü imgelere ve filtrelenmiş gürültülü imgelere uygulanmıştır. Gürültülü imgeler orijinal imgelere gauss ve tuz & biber gürültüsü eklenerek elde edilmiştir. Ayrıca, imgeler üzerindeki bulanıklığın süperpiksel algoritmaları üzerindeki etkisi de incelenmiştir. Filtreleme işleminde bölütleme problemlerinde sıklıkla kullanılan iki filtre tipi, median ve wiener filtre kullanılmıştır.

Takip eden bölümlerde sırasıyla süperpiksel algoritmaları, uygulama örnekleri ve elde edilen sonuçlar sunulacaktır ve son kısımda ise sonuçlar tartışılacaktır.

II. SÜPERPİKSEL ALGORİTMALARI

Süperpiksel algoritmaları imge işlemede farklı kullanım alanlarına sahip olduğundan tek bir başlık altında toplamak mümkün değildir. Bu yüzden kullanılan alana bağlı olarak çeşitli yöntemler geliştirilmiştir. Kullanılan metotlar imgedeki piksel gruplarını oluşturmak için kümeleme tabanlı, graf tabanlı, kontur tabanlı ve yol tabanlı olarak ana başlıklar altında toplanabilmektedir. Kümeleme tabanlı algoritmalar derinlik, renk bilgileri kullanılmaktadır ve merkez noktasından başlayarak k -ortalama algoritması gibi kümeleme algoritmasından esinlenilmiştir. Sezgisel olarak oluşturulan süperpiksellerin sayısı ve kompaktlığı kontrol edilebilmektedir. Kümeleme tabanlı algoritmalar iteratif olarak çalışmasına rağmen bağlılığı arttırmak için son işleme ihtiyaç duyulmaktadır. Graf tabanlı algoritmalar öncelikle imgeyi yönsüz graf teorisi yardımıyla pikseller düğüme ve bu düğümleri birbirine bağlayan kenarlara dönüştürülmektedir. Kenar ağırlıkları renk benzerliği veya farklılığının hesaplanması ile oluşturulmaktadır. Kontur tabanlı algoritmalar ise imgede önceden belirlenen sayı ile küme merkezleri oluşturulmaktadır. Belirlenen merkez noktalarından başlayarak gelişen konturlar ile süperpikseller meydana gelmektedir. Kontur tabanlı algoritmalar etiketlenmemiş pikseller kaldığı için son işleme gerek duymaktadır. Bir diğer yöntem ise yol tabanlı algoritmalar. Yol tabanlı algoritmalar merkez noktalarından belirli bir kritere göre pikseller boyunca yol bağlantıları ile imgeyi bölerek süperpikseller oluşturulmaktadır. Bu tip algoritmalar süperpiksel sayısı kolaylıkla kontrol edilebilirken, kompaktlık genellikle kontrol edilemez. Bu çalışmada kümeleme tabanlı algoritmalar Simple Linear Iterative Clustering (SLIC), yol tabanlı algoritmalar Regularity Preserved Superpixels (RPS) ve kontur tabanlı algoritmalar Turbopixel (TP) incelenmiştir.

A. Simple Linear Iterative Clustering (SLIC)

SLIC, kümeleme tabanlı süperpiksel algoritmaları arasında literatürde en çok kullanılan algoritmaların

başında gelmektedir. Bu algoritma başlangıç kümelerinin oluşturulması ve gradyan bilgisine bağlı olarak bu kümelerin yinelemeli güncellenmesine mantığına dayanmaktadır. İlk olarak oluşturulan başlangıç kümeleri için merkez noktaları 3×3 komşuluğundaki noktalar arasındaki en düşük gradyan bilgisine göre belirlenmektedir. Daha sonra imgeye ait tüm piksellerin hangi süperpiksele ait olduğu atanır. Bu atama işlemi temel olarak k -ortalama algoritmasının mantığına dayanmaktadır. Burada her bir süperpikselin merkez noktası ile imgeye ait piksellerin arasındaki mesafeye bakılarak o pikselin en yakın olduğu süperpiksel belirlenir. Burada mesafe değeri için konum değerlerinden ve bu algoritmada kullanılan $\{L, a, b\}$ renk uzayındaki renk bilgisinden yararlanılmaktadır ve (1) nolu denklem kullanılarak hesaplanmaktadır. Bir pikselin etiketine mesafe değeri en küçük olan süperpikselin etiket değeri atanır. Atama gerçekleştirildikten sonra her bir süperpiksel merkezi, süperpiksel içerisindeki piksellerin ortalaması alınarak güncellenir ve bu güncelleme işlemi artık hata değerinin belirlenen eşik değerinden küçük olma şartını sağlayana kadar yinelenir. Artık hata, yeni merkezlerle eski merkezler arasında öklit uzaklığının hesaplanması ile elde edilir. Ancak tüm piksellerin tek tek kontrol edilmesi işlem yükünü arttırdığından SLIC algoritmasında işlem yükünün azaltılması için k -ortalama algoritmasında yeniden düzenlenmiştir ve imgenin tüm uzayının kontrol edilmesi yerine $2S \times 2S$ arama uzayı kullanılmaktadır. Burada başlangıç adımında belirlenen süperpikselin kenar uzunluğudur ve $S = \sqrt{N/k}$ formülü ile hesaplanmaktadır. N imgenin tüm piksel sayısı, k ise belirlenen süperpiksel sayısına karşılık gelmektedir.

$$\begin{aligned} d_c &= \sqrt{(l_j - l_i)^2 + (a_j - a_i)^2 + (b_j - b_i)^2} \\ d_s &= \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \\ D &= \sqrt{d_c^2 + m^2(d_s / S)^2} \end{aligned} \quad (1)$$

Denklem (1)'de d_c ve d_s oluşturulan süperpikseller ile imgeye ait piksellerin sırasıyla renk ve konum değerleri arasındaki mesafeyi göstermektedir. D ise etiket atama işleminde kullanılan mesafe ölçüsünü göstermektedir. Burada m , renk ve konum değerleri arasındaki ağırlık ölçüsüdür ve $[1, 40]$ aralığındadır (Şekil 1(a)).

B. Regularity Preserved Superpixels (RPS)

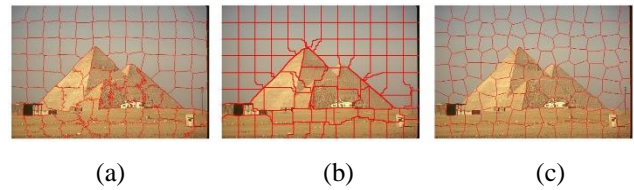
RPS algoritması, süperpiksellerin oluşumu sırasında süperpiksel yapılarının düzenli olmasını amaçlamaktadır. Düzenli süperpiksel kavramını değerlendirmek için iki kritere bakılmaktadır. Kriterlerden ilki, süperpikseller piksellerin bilgisini temsil etmektedirler ve piksellerden oluşan bölgelerin birbirleriyle uyumlu olması gerekmektedir. İkinci kriter ise piksellerin yapı, homojenlik, mekânsal topolojisi gibi kendine ait özellikleri koruması gerekmektedir. Algoritma genel anlamda üç aşamadan oluşmaktadır. İlk aşama bağlantı noktalarının

bulunması işlemidir. Giriş imgesi için kenar genlik haritasına ihtiyaç duyulmaktadır. Çalışmada gPb haritası kullanılmaktadır, çünkü nesne sınırlarında arka plan bölgelerine göre daha yüksek değer almaktadır. Sınır haritasının değer aralığı $(0,1]$ 'dir. Bağlantı noktalarının yoğunluğu süperpiksel sayısına bağlıdır. Süperpiksel sayısı, algoritmanın giriş parametresidir. Algoritmanın ikinci aşaması, bağlantı noktalarının yer değiştirmesi işlemidir. Bağlantı noktaları sınır haritasının maksimal genliğine göre optimal pozisyona hareket ettirilir. Komşu bağlantı noktalarıyla çarpışmayı sınırlandırmak için bölgesel arama uzayı belirlenir. Her bir bağlantı noktasının optimal pozisyonda yer değiştirmesi için iki kısıt vardır. Kısıtlardan biri süperpikselerin bağlantı noktaları nesne sınırları üzerinde olmasıdır ve diğer kısıt ise süperpikselerin konumsal yapılarını koruması gerekmektedir yani başlangıçtaki ızgara yapıya yakın olması gerekmektedir. Algoritmanın son aşaması ise bağlantı noktalarının birleştirilmesi işlemidir. Burada komşu bağlantı noktaları dikey ve yatay olarak bağlanmaktadır. Bağlama problemi en kısa yol probleminden yararlanılarak çözülmektedir. En kısa yol problemi kullanılırken yönsüz graf teorisinden yararlanılmaktadır. Graf teorisinde piksellere düğüm ve bu düğümleri birbirine bağlantı kuran hatlara kenar adı verilmektedir. Kenarların ağırlığı, komşu düğümlerin ilişkisine bağlı olarak değişmektedir. Sonuç olarak oluşan süperpikseller ızgara benzeri yapılarda oluşmaktadır ve sınır uyumu en kısa yol yöntemi ile korunmaktadır. ızgara yapılarının kısıtlı olması ve sınırlı arama alanı sayesinde homojen süperpikseller oluşturulmaktadır (Şekil 1(b)).

C. Turbopixel (TP)

TP süperpiksel algoritmasında, süperpiksel olarak adlandırılan bir imgenin yoğun bir yansımaları hesaplamak için geometrik akış temelli bir algoritma tanımlanmaktadır. Algoritma bir taraftan lokal imge sınırlarını korurken bir taraftan da kompaktlık kısıtı ile az bölütlemeyi sınırlandırmaktadır. Çalışmanın temel fikri; süperpiksel hesaplamasını etkili bir şekilde çözülebilir bir geometrik akış problemine indirgemektedir. Algoritma beş ana prensibe dayanmaktadır. Bunlardan ilki düzgün boyut ve kapsama alanıdır. Süperpiksel bölütleme bir imgeyi bölütlerken oluşan bölütlerin boyut ve şekil bakımından hemen hemen eşit alanlarda oluşumunun incelenmesi durumudur. Çalışmada her bir tohum süperpiksele karşılık gelmektedir ve düzgün dağıtılan tohumların başlangıç kümeleri genişleyerek bir geometrik akış tasarlanarak gerçekleşmektedir. İkincisi ise bağlılıktır ve her bir süperpiksel piksel kümelerinin birlikteliği ile gösterilmektedir. Bu çalışmada genişleme tabanlı akış, seviye kümesi (level set) uygulaması ile birleştirilmiştir. Bir diğer prensip kompaktlıktır. Bölgesel kenar bilgisi bilinmediğinde süperpikseller kompakt yapısından çıkarım yapılmalıdır. Merkez noktasından dairesel olarak genişleme başlar ve süperpiksel sınırlarının yerleri hakkında öncü bilginin olmadığı varsayılmaktadır. Kompaktlığı maksimum yapmak için merkez noktasından

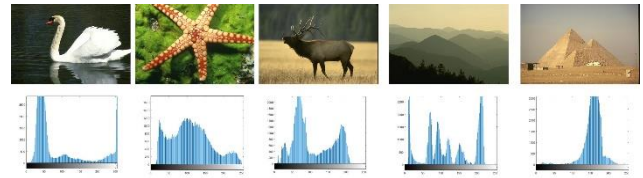
başlayarak dışa doğru genişleyen ve sabit yoğunlukta bir bölge oluşumuna kadar devam etmektedir. Dördüncü prensip ise pürüzsüzlüktür. Büyüme durduğunda süperpiksel sınırları imgenin kenarları ile kesişmesi gerekmektedir. Geometrik akış formülasyonu için üç özellik gerekmektedir. Kenarlara yaklaştıkça sınır büyümeleri yavaşlamalı, kenarlar belirgin olmalı ve sınırlar pürüzsüz olmalıdır. Sonuncu ise çakışma olmama durumudur. Her bir piksele atanan etiket tek olmalıdır. Algoritma adımları sırasıyla başlangıç merkez noktalarının yerlerinin belirlenmesi, T süre boyunca sınırlarının genişlemesi, atanmayan piksel bölgelerinin iskelet yapısının oluşturulması ve bu iskelet yapısının güncellenmesi, son olarak hız güncellenmesi ve süperpiksel çıkarımı gerçekleştirilmesi ile sağlanmaktadır (Şekil 1(c)).



Şekil 1. Algoritmaların 100 süperpiksel gösterimi (a)SLIC, (b)RPS, (c)TP

III. UYGULAMA VE SONUÇLAR

Süperpiksel algoritmalarının performanslarını ölçebilmek amacıyla, literatürdeki diğer bölütleme ve süperpiksel çalışmalarında da kullanılan Berkeley Bölütleme veri kümesi kullanılmıştır (<https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/>). Bu veri kümesinden farklı kontrast, farklı doku özelliklerine sahip 5 adet imge kullanılmıştır (Şekil 2). Kullanılan imgeler 321x481 boyutlarındadır.

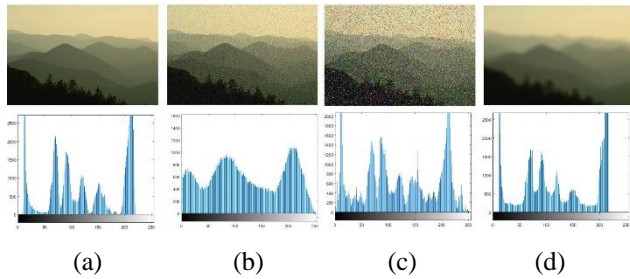


Şekil 2. Orijinal imge gösterimi ve histogram dağılımları

A. Gürültü Performansının İncelenmesi

Bu çalışmada, süperpiksel algoritmalarının gürültüye karşı bağımsızlığı incelenmektedir. Gürültü performansının incelenmesi için orijinal imgelere gürültü ekleyerek yapay imgeler oluşturulmuştur. Bu yapay imgeler üç farklı gürültü tipi ve her bir gürültü tipi için farklı seviyeleri oluşturulmuştur. Süperpikselerin bölütleme performansı 5 farklı gauss gürültüsü seviyesi ($\sigma = 0.002, 0.004, \dots, 0.01$), 4 farklı tuz&biber gürültü yoğunluğu (% 4, %8, %12, %16) ve 4 farklı bulanıklık komşuluğu ($k = 5, 9, 13, 17$) için toplam 14 farklı seviyede incelenmiştir (Şekil 3). Haberleşme kanalından imge gürültülü olarak geldiğinde veya

biyomedikal bir imgenin cihaz kaynaklı gürültülü olduğunda bir ön işleme tabi tutularak süperpiksel performansı incelenerek istenilen sonuca yakınlığı değerlendirilmiştir. Gürültülü imge median filtresi ve wiener filtresinden geçirilerek oluşturulan yeni yapay imgeler ile orijinal imgedeki süperpiksel performansı ile karşılaştırılmıştır. Median filtre belirli bir pencere aralığındaki sayıların ortancasını alarak aşırı büyük atlamaları kaldırmak için kullanılmaktadır. Wiener filtresi ise frekans bölgesinde çalışan bir filtredir ve doğrusal olarak çalışmaktadır.



Şekil 3. (a) Orijinal imge ve histogram dağılımı (b) Gauss gürültülü imge ve histogram dağılımı (0.01) (c) Tuz & biber gürültülü imge ve histogram dağılımı (% 16) (d) Bulanıklaştırılmış imge ve histogram dağılımı (17)

PSNR (Peak Signal-Noise Ratio) tepe sinyal gürültü oranı olarak adlandırılmaktadır ve imgenin kalitesinin değerlendirilmesinde kullanılmaktadır. İki imgenin benzerliği gösteren PSNR değeri, sonsuz değere sahip ise aynı iki imge olduğunu göstermektedir. Tablo I'de farklı doku ve kontrast özelliklerine sahip 5 imgenin orijinal, gürültülü ve filtrelenmiş versiyonları için elde edilen ortalama PSNR değerleri sunulmuştur. Gauss ve tuz & biber gürültülü imgelere uygulanan filtreler imgenin kalitesini oldukça yükseltmektedir. Fakat bulanıklık işlemi sonucunda çok küçük değişimler meydana gelmiştir. Bunun nedeni ise bulanıklık işlemi ile imgenin komşu piksel değerlerinin birbirine yaklaşmasıdır.

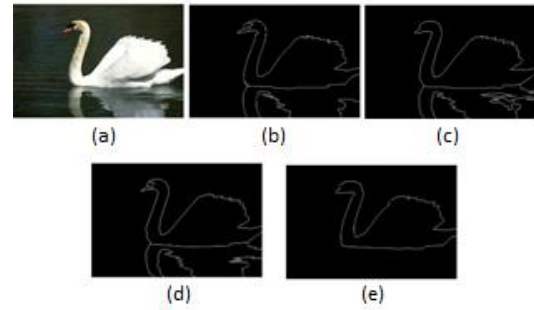
Tablo I. Gürültülü ve filtrelendi imgelerin PSNR değerlerinin karşılaştırılması

	Gürültülü İmge	Median Filtrelendi İmge	Wiener Filtrelendi İmge
Gauss gürültüsü (0.01)	20,36596	26,33111	27,03686
Tuz & biber gürültüsü (% 16)	12,86845	29,64084	17,60111
Bulanıklık (17)	25,14131	25,12298	25,13261

B. Performans Metrikleri

Süperpiksel algoritmalarının başarılarını ölçebilmek için literatürde çeşitli performans metrikleri önerilmiştir. Süperpiksel algoritmalarının performanslarını ölçmek amacıyla, kullanılan imgelerin gerçek sınır bilgilerini içeren kesin referans verisi kullanılmıştır. Bu veri yine aynı

veri kümesinde her bir imge için kullanıcılara sunulmaktadır. Her bir imge için farklı segment sayılarını içeren kesin referans verisi bulunmaktadır (Şekil 4). Süperpiksel algoritmaların elde edilen süperpiksellerin sahip olduğu içerik ve sınır bilgilerinin kesin referans verisindeki içerik ve sınır bilgisiyle olan yakınlık ve benzerlik ilişkileri analiz edilerek süperpiksellerin başarıları değerlendirilmektedir.



Şekil 4. Orijinal imge ve kesin referans gösterimi (a) Orijinal imge (b) 10 bölümlü kesin referans (c) 9 bölümlü kesin referans (d) 8 bölümlü kesin referans (e) 2 bölümlü kesin referans

1) Erişilebilir Bölütleme Doğruluğu (EBD)

Her bir süperpikselin etiketi ile kesin referansın etiketinin alanine hesaplar. Süperpiksel $S = \{S_1, S_2, \dots, S_j\}$, kesin referans $G = \{G_1, G_2, \dots, G_i\}$ ile gösterilmektedir. N toplam piksel sayısıdır. EBD değerinin yüksekliği ile süperpiksel algoritmasının performansı doğru orantılıdır.

$$EBD(S, G) = \frac{1}{N} \sum_{S_j \in S} \max_{G_i} \{|S_j \cap G_i|\} \quad (2)$$

$$S_j \cap G_i = \{x \in S_j \mid x \in G_i\}$$

2) Sınır Örtüşmesi (SÖ)

Süperpiksel sınırları ile kesin referansın oluşturduğu sınırların birbirine yakınlık derecesinin ifadesidir. En ideal durum her iki bölütleme sonucunda sınırlarının örtüşmesidir. p kesin referansın oluşturduğu bölütlerin sınır piksellerinin, q ise süperpiksellerin oluşturduğu sınır pikselleridir. $dist_{xy}$, sınırlar arasında öklit mesafesi hesaplanmaktadır ve ε değeri 2 olarak seçilmiştir. SÖ değeri $[0,1]$ arasında olup 1'e yakınlığı süperpiksel algoritmasının başarısını göstermektedir.

$$SÖ_G(S) = \frac{\sum_{p \in \delta G} \phi(\min_{q \in \delta S} dist_{xy}(p, q) < \varepsilon)}{\delta G} \quad (3)$$

3) Açıklık Değişimi (AD)

Süperpiksellerden kaynaklanan renk bozukluk seviyesini ölçmektedir yani süperpikseller ile orijinal

pikseller arasındaki tüm farklılıkları değerlendirmektedir. Burada, N süperpiksel sayısını göstermekte ve μ_k her bir süperpikselin piksel yoğunluğunun değerlerinin ortalamasını ifade etmektedir. μ değeri ise tüm imgenin ortalama değeridir. x_i imgedeki her bir pikselin yoğunluğunu ifade etmektedir. AD değerinin yüksek olması süperpiksellerin oluşturduğu piksel gruplarının homojenliğini göstermektedir.

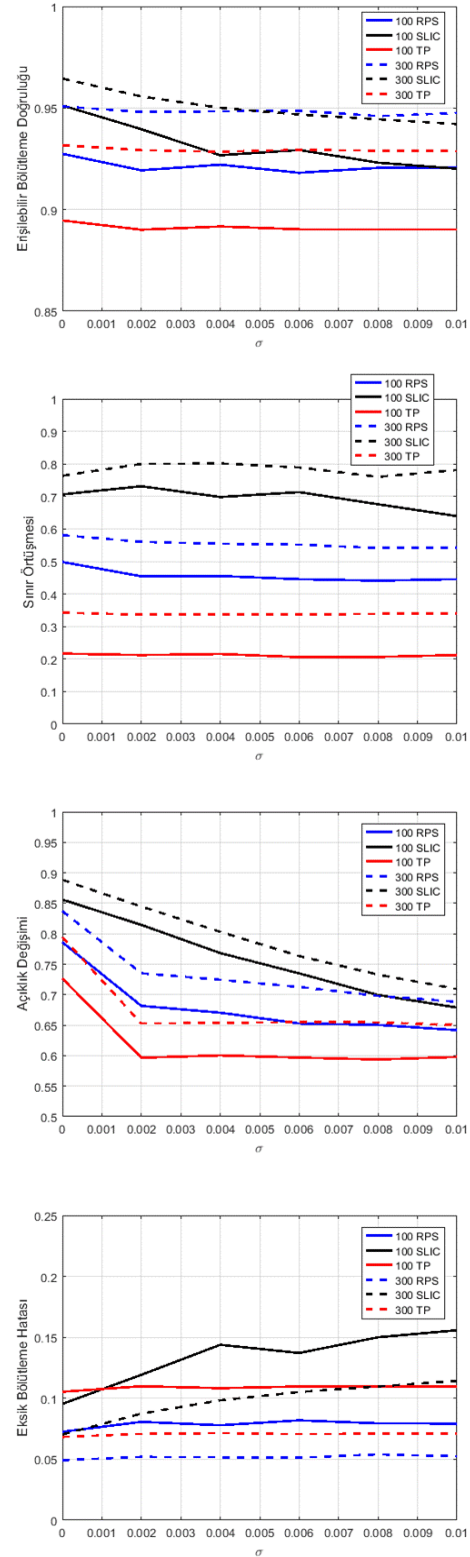
$$AD = \frac{\sum_{k=1}^N (\mu_k - \mu)^2}{\sum_{i \in I} (x_i - \mu)^2} \quad (4)$$

4) Eksik Bölütleme Hatası (EBH)

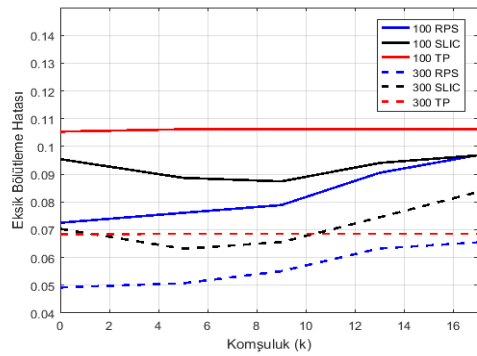
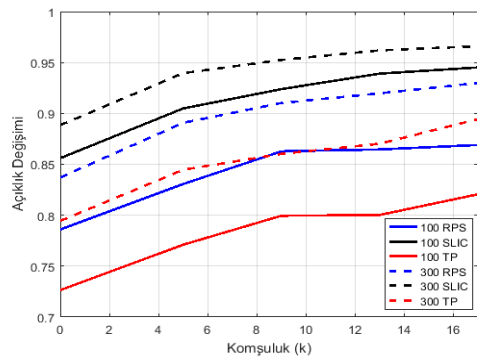
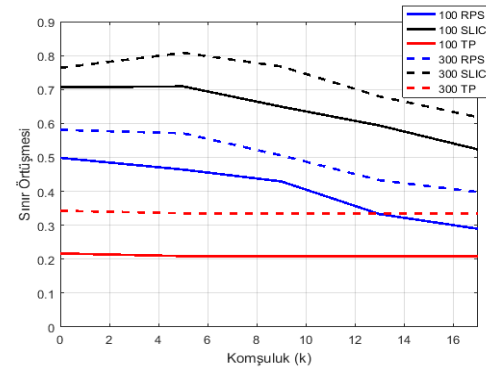
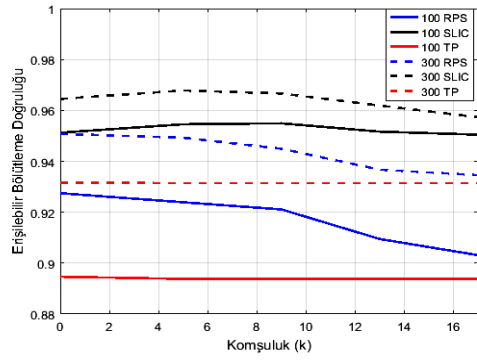
Kesin referans bölgeleri ile süperpiksel bölgeleri arasında sızıntı bölgeleri ölçer yani örtüşmeyen bölgeleri hesaplar. Süperpiksel $S = \{S_1, S_2, \dots, S_j\}$, kesin referans $G = \{G_1, G_2, \dots, G_i\}$ ile gösterilmektedir. N imgedeki toplam piksel sayısıdır. Eksik Bölütleme hatası değeri $[0,1]$ aralığındadır ve hata değerinin 0'a yakın olması istenilmektedir.

$$EBH(S, G) = \frac{1}{N} \sum_{G_i \in G} \sum_{S_j \cap G_i \neq \emptyset} \min\{|S_j \cap G_i|, |S_j - G_i|\} \quad (5)$$

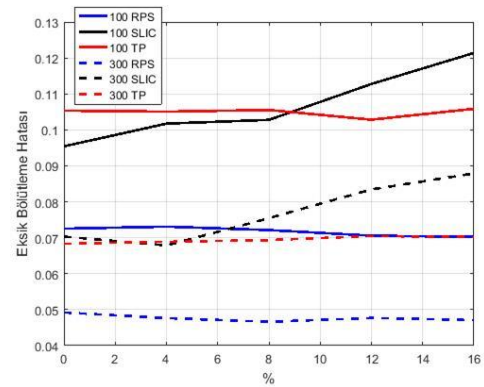
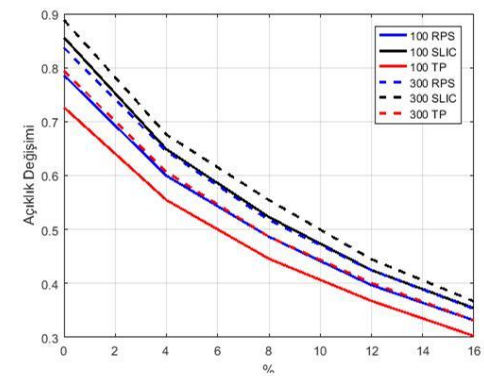
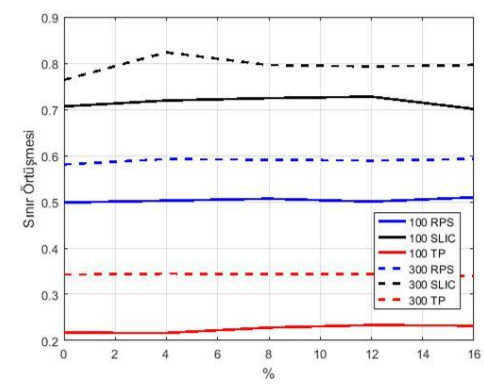
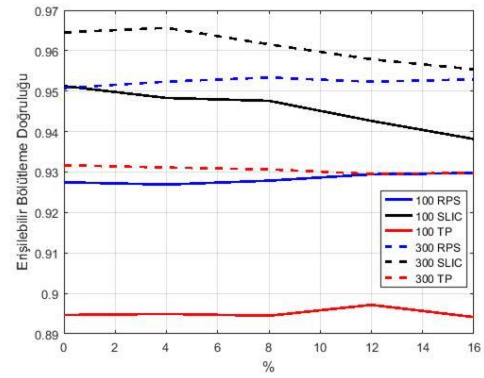
Şekil 5, Şekil 6 ve Şekil 7 orijinal imge ve farklı gürültü tiplerinin farklı gürültü seviyelerindeki performans metriklerinin değişimi gösterilmektedir. Tüm sonuçlar çalışmada kullanılan 5 imgenin farklı kesin referans bölütlerinin ve kullanılan imgelerin ortalamasını alınmasıyla elde edilmiştir. Şekil 5'te farklı seviyelerdeki gauss gürültüsünün 100 ve 300 süperpiksel için süperpiksel algoritmalarının etkisi incelenmiştir. Süperpiksel sayısının artması ile süperpiksel algoritmalarının performansı iyileşmektedir. Fakat gürültü seviyesinin artmasıyla imgenin yapısında meydana gelen bozulmalardan dolayı performansı düşmektedir. EBD değeri için gürültü bağımsızlığı yüksek olan algoritma TP'dir. Diğer algoritmalarda original değerleri ile gürültü seviyesinin artmasıyla oluşan değerlerde farklar ortaya çıktığı görülmektedir. Fakat sayısal değer olarak SLIC algoritmasının performansı daha iyidir. SÖ ve AD parametreleri gürültü seviyesine bağlı olarak SLIC algoritmasında düşme meydana gelirken TP ve RPS algoritmalarında kısmen performansını korumuştur. Yine aynı şekilde sayısal değer olarak SLIC daha iyi performans göstermektedir. EBH parametresinde ise RPS algoritması hem performans olarak hem de gürültü bağımsızlığı olarak TP ve SLIC algoritmasına kıyasla daha iyi performans sergilemektedir. Şekil 6'da bulanıklık işleminin performans metrikleri üzerindeki etkisi gösterilmektedir.



Şekil 5. Gauss gürültüsünün performans parametrelerine etkisi



Şekil 6. Bulanıklığın performans parametrelerine etkisi



Şekil 7. Tuz&biber gürültüsünün performans parametrelerine etkisi

EBD ve SÖ metriklerinde performans değeri olarak incelendiğinde SLIC algoritmasının daha başarılı olduğu gözlemlenmektedir. Gürültü bağışıklığı incelendiğinde TP algoritması daha başarılıdır. Bir imgeye gürültü eklendiğinde performans değerlerinin düşmesi beklenmektedir fakat bulanıklık komşuluğu arttıkça imgedeki piksel değerleri birbirine yaklaşmaktadır. Bu yüzden AD değeri bu gürültü türünde gürültü seviyesinin artmasıyla performansının daha da iyileştiği gözlemlenmektedir. Ayrıca sayısal değer olarak incelendiğinde SLIC algoritması daha başarılıdır. EBH metriğinde ise gürültü bağışıklığı dikkate alındığında TP algoritması, değer bazında incelendiğinde RPS algoritmasının daha başarılı olduğu gösterilmektedir. Şekil 7'de tuz&biber gürültüsünün etkileri sunulmuştur. EBD ve SÖ metriklerinde algoritmaların gürültüye bağışıklıkları oldukça yüksektir. SLIC algoritmasının bu iki metrikte de daha başarılı olduğu gözlemlenmektedir. AD metriğinde ise gürültü bağışıklığından söz etmek oldukça zordur fakat sayısal değer olarak incelendiğinde yine SLIC algoritmasının başarısı aşikârdır. EBH metriği ise hem gürültü bağışıklığı hem de sayısal değer bakımından incelendiğinde RPS algoritması daha başarılıdır.

Tablo II. Bölütleme algoritmalarının filtrelenmiş imgeler için erişilebilir bölütleme doğruluğu (EBD) performansı

Gürültü Tipi	EBD			
	Filtre	RPS	SLIC	TP
Gauss	Median	0.9366	0.9507	0.9120
	Wiener	0.9371	0.9538	0.9126
Tuz & Biber	Median	0.9386	0.9596	0.9127
	Wiener	0.9370	0.9469	0.9126
Bulanıklık	Median	0.9274	0.9583	0.9126
	Wiener	0.9269	0.9584	0.9126
Orijinal		0.9391	0.9579	0.9132

Tablo III. Bölütleme algoritmalarının filtrelenmiş imgeler için sınır örtüşmesi (SÖ) performansı

Gürültü Tipi	SÖ			
	Filtre	RPS	SLIC	TP
Gauss	Median	0.5155	0.7248	0.2724
	Wiener	0.5219	0.7356	0.2722
Tuz & Biber	Median	0.5323	0.7513	0.2736
	Wiener	0.5093	0.7164	0.2722
Bulanıklık	Median	0.4246	0.6649	0.2722
	Wiener	0.4198	0.6606	0.2722
Orijinal		0.5398	0.7352	0.2797

Tablo IV. Bölütleme algoritmalarının filtrelenmiş imgeler için açıklık değişimi (AD) performansı

Gürültü Tipi	AD			
	Filtre	RPS	SLIC	TP
Gauss	Median	0.8166	0.8648	0.7618
	Wiener	0.8286	0.8815	0.7751
Tuz & Biber	Median	0.8302	0.8885	0.7743
	Wiener	0.7327	0.7714	0.6845
Bulanıklık	Median	0.8843	0.9424	0.8464
	Wiener	0.8869	0.9449	0.8473
Orijinal		0.8115	0.8722	0.7604

Tablo V. Bölütleme algoritmalarının filtrelenmiş imgeler için eksik bölütleme hatası (EBH) performansı

Gürültü Tipi	EBH			
	Filtre	RPS	SLIC	TP
Gauss	Median	0.0634	0.0971	0.0880
	Wiener	0.0629	0.0908	0.0874
Tuz & Biber	Median	0.0614	0.0795	0.0873
	Wiener	0.0630	0.1043	0.0874
Bulanıklık	Median	0.0726	0.0813	0.0874
	Wiener	0.0731	0.0814	0.0874
Orijinal		0.0609	0.0829	0.0868

Tablo II-V'te filtrelenmiş gürültülü imgeler için bölütleme performans metrikleri sunulmuştur. Tablolarda verilen değerler farklı imge, kesin referans ve süperpiksel sayıları için elde edilen ortalama değeri ifade etmektedir. Tablo II'de görüldüğü üzere SLIC algoritması hem orijinal imge hem de filtrelenmiş gürültülü imgeler için en iyi erişilebilir bölütleme doğruluğu değerine sahiptir. Tablo III'te görüldüğü üzere SLIC algoritması hem orijinal imge hem de filtrelenmiş gürültülü imgeler için en iyi sınır örtüşmesi değerine sahiptir. Burada dikkat çeken husus SLIC algoritmasının bulanıklaştırılmış imge için performansının düşmesidir. Bulanıklık işlemi ile imgedeki iki farklı imgeye ait renk ve doku özellikleri birbirine yaklaşmakta ve sınır bilgisi daha az görünür olmaktadır. Dolayısıyla, bu değişim sınır örtüşmesi değerinde gözle görülür bir düşüşe neden olmaktadır. Benzer şekilde SLIC algoritması hem orijinal imge hem de filtrelenmiş gürültülü imgeler için açıklık değişimi performansına sahip algoritmadır (Tablo IV). Bu noktada dikkat çeken önemli iki husus bulunmaktadır. Bulanıklık için elde edilen değerlere bakıldığında tüm süperpiksel algoritmalarının AD değerlerinin arttığı görülmektedir. Bu artış AD formülasyonunda payda kısmında bulunan resmin genel piksel ortalamasının piksel değerlerine çok yaklaşmış olmasından ileri gelmektedir. Son olarak tablo V'te görüldüğü üzere RPS algoritması hem orijinal imge hem

de filtrelenmiş gürültülü imgeler için en düşük eksik bölütleme hatası değerine sahiptir.

IV. TARTIŞMA

Bu çalışmada gürültünün süperpiksel algoritmalarının performanslarını nasıl etkilediği ve ön işlem olarak gürültü giderici filtreler kullanıldığında bölütleme performansının hangi oranda iyileştirileceği incelenmiştir. Literatürde en çok kullanılan SLIC, RPS ve TP süperpiksel algoritmalarına uygulanmıştır. Çalışmada iki farklı durum incelenmiştir. İlk olarak bölütleme algoritmalarının performanslarının gürültü çeşidi ve seviyesine göre değişiminin incelenmesidir. Bu amaçla gauss gürültüsü, tuz&biber gürültüsü ve bulanıklık gibi bozulmalar orijinal imgeye belli oranda eklenerek sonuçları incelenmiştir. İkinci olarak ise ön işlem olarak gürültü giderici filtre kullanımının farklı gürültü tipi ve seviyelerinde elde edilen gürültülü imgelerin bölütlenme başarısına etkisi incelenmiştir. Elde edilen sonuçlar ışığında farklı gürültü tipleri ve seviyelerine göre en iyi bölütleme performansını SLIC algoritması sağlamaktadır. Ancak SLIC algoritmasının gürültü bağışıklığı düşüktür. Ayrıca, SLIC algoritması filtre uygulanması durumunda da erişilebilir bölütleme doğruluğu, sınır örtüşmesi ve açıklık değişimi metrikleri bakımından en iyi performansı sergilemektedir. Diğer taraftan, kullanılan median ve wiener filtrelerin gauss ve tuz & biber gürültüsü içeren imgeler için bölütleme performansının korunmasını sağladıkları görülmektedir. Ancak bu durum bulanıklık için geçerli değildir.

Bu çalışma özellikle biyomedikal imgeler gibi gürültü içeren imgeleri etkili bir şekilde bölütleyebilen süperpiksel tabanlı gürbüz bir yöntemin geliştirilmesine yönelik gerçekleştirilen çalışmanın ilk sonuçlarını içermektedir. Gelecek çalışmalarda gürültü bağışıklığı yüksek ve farklı gürültü şartlarında bölütleme performansı yüksek süperpiksel algoritmasının geliştirilmesi hedeflenmektedir.

TEŞEKKÜR

Bu çalışma SDÜ-BAP 4692-YL2-16 projesi kapsamında desteklenmektedir.

KAYNAKÇA

- [1] Z. Li, X.-M. Wu, and S.-F. Chang, "Segmentation using superpixels: A bipartite graph partitioning approach," in *Computer Vision and Pattern Recognition (CVPR)*, 2012 IEEE Conference on. IEEE, 2012, pp. 789–796.
- [2] L. Zhang, Y. Gao, Y. Xia, K. Lu, J. Shen, and R. Ji, "Representative discovery of structure cues for weakly supervised image segmentation," *IEEE Trans. Multimedia*, vol. 16, no. 2, pp. 470–479, 2014.
- [3] G. Mori, "Guiding model search using segmentation," in *Int. Conf. Comp. Vision (ICCV)*, 2005.
- [4] X. Sun, K. Shang, D. Ming, J. Tian, J. Ma, A biologically-inspired framework for contour detection using superpixel-based candidates and hierarchical visual cues, *Sensors* 15 (10) (2015) 26654–26674.
- [5] Despotović, Ivana, et al. "Noise-robust method for image segmentation." *Advanced Concepts for Intelligent Vision Systems*. Springer Berlin/Heidelberg, 2010.
- [6] Computer Vision and Image Understanding, In press, accepted manuscript, Available online 2 April 2017 David Stutz, Alexander Hermans, Bastian Leibe
- [7] Roels J., De Vylder J., Aelterman J., Lippens S., Saeys Y., Philips W. (2016) Superpixel Quality in Microscopy Images: The Impact of Noise & Denoising. In: Kyriacou E., Christofides S., Pattichis C. (eds) XIV Mediterranean Conference on Medical and Biological Engineering and Computing 2016. IFMBE Proceedings, vol 57. Springer, Cham
- [8] Wu, Cheng-Ho, and Heng-Hua Chang. "Superpixel-based image noise variance estimation with local statistical assessment." *EURASIP Journal on Image and Video Processing* 2015.1 (2015): 38.
- [9] Fu, Peng, et al. "Image noise level estimation based on a new adaptive superpixel classification." *Image Processing (ICIP)*, 2014 IEEE International Conference on. IEEE, 2014.
- [10] Achanta, Radhakrishna, et al. "SLIC superpixels compared to state-of-the-art superpixel methods." *IEEE transactions on pattern analysis and machine intelligence* 34.11 (2012): 2274–2282.
- [11] Levinshtein, A., Stere, A., Kutulakos, K. N., Fleet, D. J., Dickinson, S. J., & Siddiqi, K. (2009). Turbopixels: Fast superpixels using geometric flows. *IEEE transactions on pattern analysis and machine intelligence*, 31(12), 2290–2297.
- [12] Fu, Huazhu, et al. "Regularity preserved superpixels and supervoxels." *IEEE Transactions on Multimedia* 16.4 (2014): 1165–1175.