

## Mean Shift Ve Gaussian Filtre İle Gölge Tespiti Shadow Detection With Mean Shift And Gaussian Filter

Yunus SANTUR<sup>1</sup>, Haluk DİLMEN<sup>1</sup>, Semiha MAKİNİST<sup>2</sup>, M. Fatih TALU<sup>1</sup>

<sup>1</sup> Bilgisayar Bölümü Mühendislik Fakültesi  
İnönü Üniversitesi

ysantur@gmail.com, hdilmen2@gmail.com, fatihtalu@gmail.com

<sup>2</sup> Bilgisayar Bölümü Mühendislik Fakültesi  
Fırat Üniversitesi

semihamakiniist@gmail.com

### Özet

Görüntü işleme problemlerinden biri olan gölge özellikle nesne takibinde veya bir cismin renk değerini belirlerken bazı problemlere neden olmaktadır. Son zamanlarda bununla ilgili bazı çalışmalar yapılmış ve farklı yöntemler geliştirilmiştir. Örneğin bunlardan biri MoG Algoritmasıdır. Bu algoritma videolar için kullanılmakta, hareketli nesnelere ön plan olarak algılamakta ve duran sabit cisimleri ise arka plan olarak algılamaktadır. Fakat bu yöntem sabit cisimlerin oluşturduğu gölgeyi tespit edemediği için uygun bir yöntem değildir. Bu makalede ise mean shift ve gaussian filtreler kullanılarak hem video hem de sabit resimlerdeki gölgelerin tespiti için bir çalışma yapılmıştır. Bu çalışma python platformunda opencv ve pymeanshift kütüphaneleri kullanılarak geliştirilmiştir.

### Abstract

Shade is one of the image processing couses problems especially when tracking abject and determining color of object. Recently, some studies we conducted and different methods has been developed. For example, MOG algorithm is one of them. This algorithm is used for video images, It detects moving objects in the foreground and background is perceived as standing still objects. This method is not suitable because It does not detect shadow formed by fixed objects. I this article a study was conducted to identify shadow of the both video and fixed images by using mean shift and gaussian filters. This study was developed in python platform using opencv and pymeanshift libraries.

### 1. Giriş

Bazı görüntü işleme çalışmalarında gölge büyük bir problem olmaktadır. Özellikle nesne takip uygulamalarında cisim ve cismin gölgesinin ayırt edilememesi hatalı takip yapılmasına neden olabilmektedir. Bu tip problemlerin giderilmesi için resimdeki gölgenin algılanması ve/veya resimden çıkarılması gerekmektedir. Literatürde bununla ilgili bazı çalışmalar

yapılmıştır. Örneğin Jiang ve arkadaşı yaptığı çalışmada video üzerinde cisim ve gölgesini ayırt ederek sadece cismin takip edilmesi için bir çalışma yapmışlardır [1]. Sanin ve arkadaşları ise bir videodaki kişileri ve gölgeleri ayırt etmek için bir yöntem önermiştir [2]. Resimlerde parlak noktaların ve gölgelerin renk varyasyon değerleri hesaplanarak renkli resimin algılanması için bir yöntem önermişlerdir [3]. Swain ve arkadaşları resimdeki renkleri indeksleyerek ön ve arka plan resimlerini çıkarmıştır [4]. Avina-Cervantes ise yaptığı çalışmada gölgenin parlaklığını ayarlayarak görüntüden kaldırmıştır [5].

Resimden gölge kaldırma işleminin doğru ve etkin yapılabilmesi için ilk adım olarak gölgenin iyi bir şekilde tespit edilmesi gerekmektedir. Bu amaçla literatürde yapılan çalışmalardan biri OpenCv kütüphanesinde de kullanılan MoG Algoritmasıdır [6]. Bu algoritma videolarda hareketli cisimlerin gölgesini tespit için kullanılır. MoG Algoritması temelinde Gaussian filtre kullanılmaktadır ve bir Gauss karışım yöntemidir. Bu yöntemin basit olarak çalışma mantığı her bir çerçevedeki piksellerin ağırlık değerlerinin hesaplanması ve oluşan bu yeni görüntü üzerinde piksel komşu uzaklıklarına bakılarak gölge eşik değerine göre (varsayılan eşik değeri “127”) gölge tespiti yapılarak arka plan ve ön plan ayrımı yapılmasıdır. Bu yöntem hareketli cisimlerin gölge tespitinde etkin olmasına rağmen sabit cisimlerin gölge tespitinde kullanılamamaktadır. Bir diğer yöntemde ise arka plan görüntüsünden resim önsel bilgilerinden faydalanarak HSV değerlerine göre gölge tespiti yapılmaktadır [7-9]. Bu yöntem ile resimde ışığı yansıtmayan görüntüler aranmaktadır, dış mekan çekimlerinde kullanışlı iken iç mekan çekimlerinde pek kullanışlı değildir. Bunu iyileştirmek için Finlayson yaptığı çalışmada iç mekanda ışıklanmayı flash kullanarak elde ettiği yüzey bilgisi ile gölge tespit etmiştir [8-10].

Bu çalışma Tübitak Teydeb 1512 programı tarafından 2130276 proje numarası ile desteklenen “otonom boyacı robot” projesinde kullanılmıştır. Sözü edilen ar-ge çalışması multikopter diye tabir edilen insansız bir robot aracılığı ile

bina dış cephelerinin görüntü işleme tekniklerinden faydalanarak otonom boyanmasıdır. Bu amaçla robot boya yapılacak bina dış cephe yüzeyindeki gezinimi sırasında yüzeylerin püskürtme tabancası ile boyanıp/boyanmamasına önsel bilgilerden ürettiği eşik değerine göre karar verecektir. Ancak deneysel çalışmalar sırasında gölge içeren görüntülerin eşik değerini bozması nedeniyle hatalı sonuçlar elde edilmiş bunun sonucunda görüntülerde ki gölgenin hızlı ve etkin bir şekilde tespit edilmesi ihtiyacı doğmuştur.

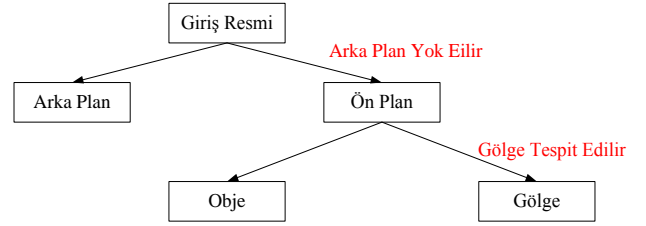
Bu çalışmada mean shift ve gaussian filtreler kullanılarak hem sabit ve hem de hareketli görüntülerde gölge tespiti için bir yöntem önerilmiştir. Yapılan çalışma Python dili kullanılarak OpenCV ortamında geliştirilmiş ve google code gurubunun geliştirdiği pymeanshift kütüphaneleri kullanılmıştır. Bu yöntem literatürdeki diğer yöntemlere göre daha basit ama doğru bir şekilde gölge tespit yapmaktadır.

Bu makaledeki ikinci bölümde önerilen yöntem hakkında bilgi verilmekte, üçüncü bölümde deneysel sonuçlar verilmekte ve dördüncü bölümde ise sonuç bölümü yer almaktadır.

## 2. Önerilen Yöntem

Gölge, saydam olmayan bir cismin üzerine düşen ışığı geçirmemesi nedeniyle oluşmaktadır. Yani kısacası bir ışık olayıdır. Görüntü işleme konusunda bu olay bazen (kullanım yerine göre) problemlere neden olmaktadır. Problemin giderilmesi için öncelikle resimdeki gölge tespit edilmelidir. Bu işlemin en temel çalışma adımları Şekil 1’de verilmiştir.

Bu yöntemde giriş verisine önce “Gauss Filtre” (komşuluk matrisi 5x5’lidir) uygulanır. Böylece resimdeki gürültü giderilir. Daha sonra elde edilen resme “Mean Shift Filtresi”



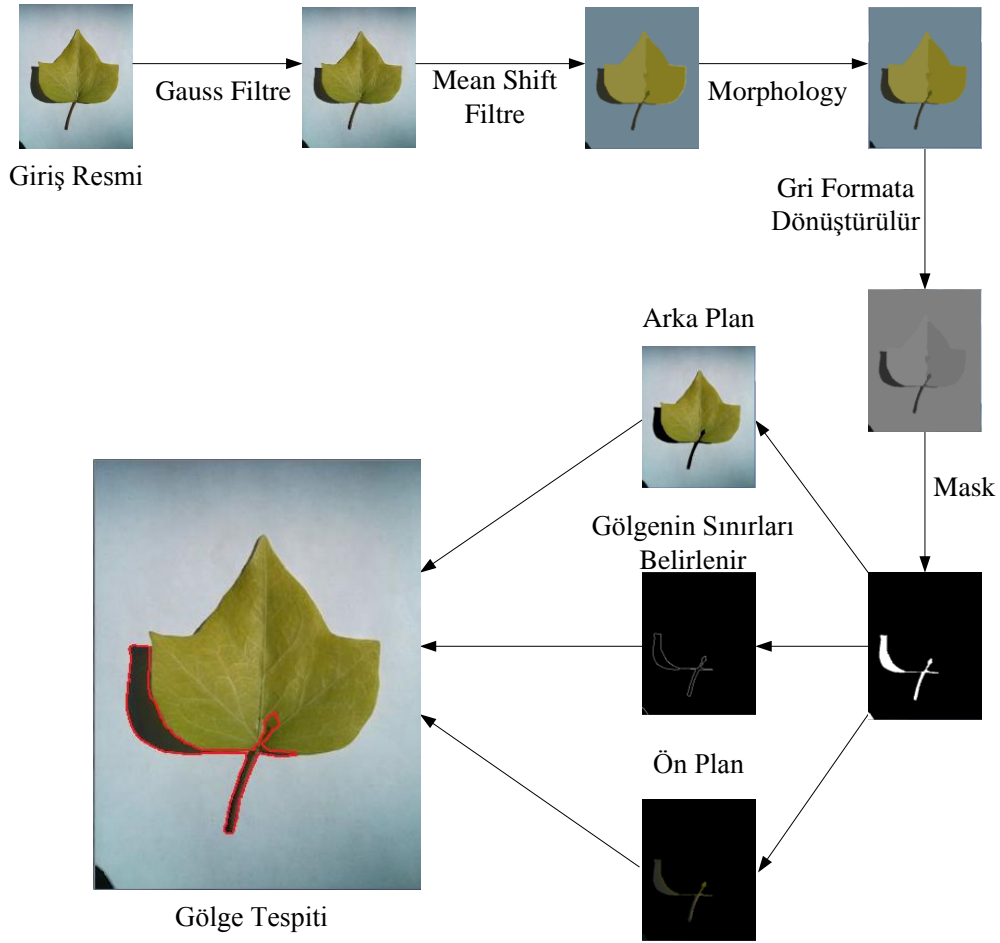
Şekil 1: Gölge tespitin basit çalışma prensibi

uygulanarak resmin netliği silinir ve en son resme “Morphology” işlemi uygulanarak resimdeki küçük bozukluklar silinir. Elimizde gerçek bir resim netliği ve gürültülerden silinmiş bir resim kalmaktadır. Bu resim önce gri resim formatına dönüştürülür bu işlem sonucunda oluşan resim gölge tespitinde kullanılacak olan maske resmidir. Denklem 1’de bir resmin gri formata dönüşüm formülü verilmiştir.

$$gri = \sum_{1}^n \left( \frac{0.29 \cdot B_n + 0.59 \cdot G_n + 0.11 \cdot R_n}{255} \right) \quad (1)$$

Bu mask kullanılarak önce gölgenin sınırları tespit edilir daha sonra ön ve arka plan resimler tespit edilir. Son adımda bu üç resim birleştirilerek Şekil 2’de gösterildiği gibi kırmızı çizgi ile gölgenin yeri tespit edilmiş olunur.

Bu makalede gölge tespiti için kullanılan yöntemin çalışma prensibi ve aşamaları Şekil 2’de verilmiştir. (Şekil 2’de verilen örnekte kullanılan pymeanshift parametreleri Spatial RADIUS=6, Range RADIUS=4.5, Min Density=150’dir.)



Şekil 2: Önerilen yöntem ile gölge tespiti

Önerilen yöntemin iyi sonuç verebilmesi için google code grubunun geliştirdiği pymeanshift kütüphanesinin kullandığı parametre değerlerini iyi belirlemek gerekmektedir. Bu hem işlem performansını hem de elde edilen çıkış değerlerini etkilemektedir. Bu nedenle optimum parametre değerlerinin tespit edilmesi çok önemlidir [13].

### 3. Bulgular

Yapılan uygulamalarda Paython 2.7 dili ve OpenCV görüntü işleme kütüphanesi kullanılmış ve tüm deneyler aynı

bilgisayar üzerinde yapılmıştır. Çalışmalar intel core i3(3M cache ve 1.4 Hz) 4 GB RAM ve 320 GB HDD ve 12.04 ubuntu ortamında ortamında geliştirilmiştir.

Gölge tespitinde, pymeanshift kütüphanesinden yararlanılarak ve ortalama kayma algoritması kullanarak resim segmentelenmektedir. Böylece resimdeki netlik yok edilmiş olunur. Buda gölgenin daha iyi bir şekilde tespiti için yardımcı olunur. Çizelge 1' de farklı pymeanshift parametre değerleri için sistem performansları verilmiştir.

Çizelge 1: Farklı pymeanshift değerlerine göre sistem performansı

|    | Mean Shift Filtre Parametreleri                            | Sistem Performansı(sn) |
|----|--|------------------------|
| 1  | Spatial Redius=6,<br>Range Redius=5.5,<br>Min Density=0    | 0.969000101089         |
| 2  | Spatial Redius=6,<br>Range Redius=4.5,<br>Min Density=150  | 2.10351395607          |
| 3  | Spatial Redius=6,<br>Range Redius=4.5,<br>Min Density=50   | 2.06179904938          |
| 4  | Spatial Redius=1,<br>Range Redius=1,<br>Min Density=300    | 0.342250823975         |
| 5  | Spatial Redius=2,<br>Range Redius=2,<br>Min Density=20     | 0.367501974106         |
| 6  | Spatial Redius=2,<br>Range Redius=2,<br>Min Density=300    | 0.272608041763         |
| 7  | Spatial Redius=3,<br>Range Redius=4.5,<br>Min Density=300  | 0.542633056641         |
| 8  | Spatial Redius=6,<br>Range Redius=1.5,<br>Min Density=300  | 3.91345214844          |
| 9  | Spatial Redius=6,<br>Range Redius=8,<br>Min Density=300    | 1.67206311226          |
| 10 | Spatial Redius=10,<br>Range Redius=10,<br>Min Density=0    | 5.9845700264           |
| 11 | Spatial Redius=10,<br>Range Redius=10,<br>Min Density=300  | 5.97231006622          |
| 12 | Spatial Redius=15,<br>Range Redius=4.5,<br>Min Density=300 | 38.514770031           |


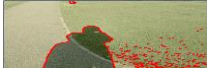






Elde edilen çıktılar gölgenin etkin bir şekilde elde edilmesi ve işlem süresi kriterlerine göre kıyaslanmıştır. Çizelge 1’ de 1 numaralı satırda elde edilen sonuç en etkin gölge tespitidir, 5 numaralı satırda elde edilen sonuçta hem gölgenin etkin belirlenmesi hemde çalışma süresi açısından en optimum sonuç elde edilmiştir. Satır 12’ de elde edilen sonuçta hem gölge tespiti hemde çalışma süresi açısından en kötü çıkış elde edilmiştir. Sistem performansı göz önüne alındığında, aralık yarı çapı azaldıkça daha hassas sistem çıktıları elde edilmektedir. Spatial Redius=2, Range Redius=2 ve Min Density=20 seçildiğinde elde edilen sonuç Şekil 3’de verilmiştir.



Şekil 3. RADIUS=2, RANGE RADIUS=2 ve MIN DENSITY=20 için sistem çıkışı

Bu işlem farklı test resimleri üzerinde denenmiş ve iyi sonuçlar elde edilmiştir. Çizelge 2’ de farklı resimler için elde edilen sonuçlar verilmiştir. Bu tabloda elde edilen değerler için parametre değerlerimiz Spatial Redius=6, Range Redius=5.5 ve Min Density=0’ dır.

Çizelge 2: Farklı görüntüler için sistem çıktıları

| Gerçek Görüntü   | Sistem Çıkışı   |
|--|---|
|  |  |
|  |  |
|  |  |
|  |  |

Önerilen bu yöntemin en büyük dezavantajı ağaç gövdeleri ve çiçek saplarını da gölge olarak algılamasıdır. Bunun nedeni bu görüntü değerlerinin ışığı yansıtmayacak şekilde koyu olması ile alakalıdır.

#### 4. Sonuçlar

Literatürde bir çok farklı gölge tespit yöntemi yer almaktadır. Çoğunlukla bu yöntemler karmaşık bir yapıya sahiptir. Bu makale ile amaç hem karmaşık işlemlerden kurtulmak hem de iyi bir performansla en iyi şekilde gölgelerin yerini tespit edebilmektir.

Bu çalışma Tübitak Teydeb 1512 programı tarafından 2130276 proje numarası ile desteklenen “otonom boyacı robot” projesinde kullanılmış ve deneysel çalışmalar esnasında Tübitak 1512 desteği ile alınan donanımlar kullanılmıştır.

Yapılan çalışmalar farklı test resimleri üzerinde denenmiştir. Yapılan gözlemler sonucunda hem performans hem de gölge tespitinde iyi sonuçlar alınmıştır. Bu çalışmada önerilen yöntem dış mekanda oluşan gölgelerin tespitinde daha iyi sonuçlar vermiştir. Çalışmanın dezavantajı ise ışığı yansıtmayan cisimler üzerinde görülmektedir.

#### 5. Kaynaklar

- [1] Jiang, H. ve Drew, M., “*Tracking objects with shadows*”, Image and Video Communications and Processing 2003. Proceedings of the SPIE, 5022:512–521, 2003.
- [2] Sanin, A. , Sanderson, C. ve Lovell, B.C., “*Improved shadow removal for robust person tracking in surveillance scenarios*”, International Conference on Pattern Recognition (ICPR), Turkey, 2010.
- [3] Klinker, G.J., Shafer, S.A. ve Kanade, T., “*The measurement of highlights in color images*” Int. J. Comput. Vision, 2:7–32, 1988.
- [4] Swain, M.J. ve Ballard, D.H. “*Color indexing*”, Int. J. Comput. Vision, 7:11–32, 1991.
- [5] Avina-Cervantes, J.G., Martínez-Jiménez, L., Devy, M., Hernández-Gutierrez, A., Almanza, D.L. ve Ibarra, M. A., “*Shadows attenuation for robust object recognition*” Proceedings of the Artificial Intelligence 6th Mexican International Conference on Advances in Artificial Intelligence, 7:650–659, 2007.
- [6] Babu, R.V., “*Background Subtraction - IP*”, <http://www.serc.iisc.ernet.in/~venky/SE263/slides/BGM2.pdf>
- [7] Prati, A., Mikic, I., Trivedi, M.M. ve Cucchiara, R., “*Detecting moving shadows: Algorithms and evaluation*”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(7), 2003.
- [8] Finlayson, G.D., Drew, M., Lu, C. ve Hordley, S.D., “*On the removal of shadows from images*”, IEEE Trans. Pattern Anal. Mach. Intell., 28(1):59–68, 2006.
- [9] Finlayson, G.D., Drew, M. ve Lu, C., “*Entropy minimization for shadow removal*”, International Journal of Computer Vision, 85, 2009.
- [10] Finlayson, G.D., Drew, M. ve Lu, C., “*Shadow removal via flash/noflash illumination*”, Multimedia Signal Processing, 2006 IEEE 8th Workshop, 2006.

- [11] Fredembach, C. ve Finlayson, G., “*Simple Shadow Removal*”, <http://ivrgwww.epfl.ch/alumni/fredemba/papers/FFICPR06.pdf>
- [12] Figov, Z., Tal, Y. ve Koppel, M., “*Detecting and Removing Shadows*”, <http://u.cs.biu.ac.il/~koppel/papers/Removing-Shadows.pdf>
- [13] <https://code.google.com/p/pymeanshift/>