

İmge Eşiklemeye Ayrık İkili PSO Temelli Yeni Bir Yaklaşım

A new approach based on Discrete Binary PSO for image thresholding

Çağlar Kılıkçier¹, Ersen Yılmaz¹

¹Elektronik Mühendisliği Bölümü
Uludağ Üniversitesi
caglar@uludag.edu.tr, ersen@uludag.edu.tr

Özet

Parçacık sürüsü optimizasyonu (PSO) tabanlı metotlarla çözülen imge eşikleme uygulamalarının çözüm uzayları ayrık ve sınırlıdır. PSO'nun ayrık çözüm uzayları için kullanılan çeşidi olan Ayrık İkili PSO'da (DB-PSO) parçacıklar bu uzaylarda yer alan ayrık noktalar arasında geçiş yapabilmektedirler. Ayrık çözüm uzayına sahip problemlerde parçacıklar yerel optimumlara takılabilirler. Bu çalışmada parçacık değerlerinin yerel optimumlardan çıkabildiği DB-PSO temelli bir yaklaşım önerilmiştir. Önerilen yaklaşımın başarısını BSDS500 veri kümesinden alınan imgeler üzerinde incelenmiştir.

Abstract

The solution space in image thresholding applications solved by PSO-based methods is discrete and limited. In DB-PSO which is a version of PSO used for discrete solution space the particles can switch between discrete points. In problems having discrete solution space the particles may get stucked on local optimum solutions. In this work we propose an approach based on DB-PSO where the particles can get out of local optimum solutions. Performance of the proposed approach is analysed on images from BSDS500 dataset.

1. Giriş

İmge eşikleme örüntü tanımadan imge filtrelemeye birçok alanda kullanılmaktadır. Eşikleme problemi için çeşitli metotlar ortaya atılmıştır. Bu metotlardan en çok kullanılanı Otsu metodudur. Bu metot eğiticiye ihtiyaç duymaz ve parametrik değildir [1]. Otsu metodundaki ölçüt imge üzerindeki nesnelere ayırmayı garantilememektedir. Ancak bu ölçüt, imgede farklı eşik seviyelerinde kalan kısımların ayırt edilebilirliğini arttırmayı başarılı bir şekilde yapabilir. Otsu metoduyla en iyi eşik değerinin bulunabilmesi için tüm eşik seviyeleri tek tek denenmelidir. Bunun yerine PSO kullanılarak daha az deneme ile eşik seviyeleri bulunabilir. Literatürde Otsu metoduyla imge eşikleme için kullanılan çeşitli PSO uygulamaları görülebilir. PSO basitliği, hızı ve yüksek başarı performansıyla geniş bir uygulama alanına sahiptir. Literatürde PSO'nun başarımının artırılması amacıyla önerilmiş pek çok yaklaşım bulunmaktadır [2-6]. PSO'nun başarımını arttırmak için ileri sürülen bu yaklaşımlarda parçacık sürüsünün konumuna ve iterasyona

bağlı olarak adapte olabilen bireysel ve de sosyal eğitim katsayıları ya da farklı iyileştirme metotlarıyla ortaklaşa oluşturulan yapılar önerilmiştir. Oluşturulan bu çeşitli varyasyonlarda amaç daha yüksek bir performansa sahip olmaktır.

İmge eşiklemede olduğu gibi ayrık seviye veya durumlarla ifade edebilen problemler için literatürde özel olarak tasarlanmış PSO temelli yöntemler önerilmiştir [6-8]. Bu yöntemlerden bir tanesini de PSO'yu öneren Kennedy ve Russell tasarlamıştır [7]. Ayrık parçacık sürüsü optimizasyonu için ikili sistemde bit temelli olan bir PSO yaklaşımı önermişlerdir. Bu yaklaşımda parçacıklar sayısal değerlerini oluşturulduğu bitlerden almaktadır. Bu bitlerin değerleri optimize edilerek istenilen değerler ayrık olarak ifade edilebilmektedir.

Ayrık seviye veya durumlarla çalışan PSO veya DB-PSO uygulamalarında parçalar yerel optimumlarda takılabilmekte veya daha önceden denenmiş değerleri tekrar tekrar alabilmektedir. Yerel optimumlarda tekrarların azaltılması için çeşitli yollar önerilmekle beraber bunu tamamen engelleyen bir yaklaşım önerilmemiştir. Bu yüzden parçalar ayrık PSO algoritmaları tarafından daha önce denenip denenmediğine bakılmaksızın tekrar kullanılabilir. Tekrar eden her parçacık çözüm uzayını tarama kapasitesini azalmakta ve işlem yükünün verimsiz olarak kullanılmasına yol açmaktadır. Bu çalışmada yukarıda bahsedilen problemin çözümü için DB PSO temelli bir yaklaşım önerilmiştir. Önerilen Parçacık Tekrarsız DB-PSO ile daha önceden denenmiş parçacık değerlerinin tekrarından kaçınılarak yerel optimumlardan çıkılması, çözüm uzayının daha az iterasyon ile daha yüksek oranda taranması hedeflenmiştir.

2. Parçacık Sürüsü Optimizasyonu

Parçacık sürüsü algoritması Eberhart ve Kennedy tarafından gerçek sayılar uzayında bir optimizasyon tekniği olarak tanıtılmıştır [7]. Başlangıçta çözüm uzayında rasgele dağılımlı olarak bulunan parçacıklar her adımda bireysel ve sosyal tecrübelerine dayanarak optimize edilmek istenen fonksiyonda en iyi sonucu veren parçacığa doğru ilerlemektedir. PSO algoritmasında parçacıkların bu ilerleme hareketi iki denklem yardımıyla tanımlanmıştır. Parçacıkların hızını veren (1) denklemi ve bu hıza göre yer değiştirmeyi sağlayan (2) denklemi.

Parçacığın hızı hesaplanırken (1) numaralı denklemde kullanılan c_1 katsayısı yerel veya bireysel olarak adlandırılan öğrenme katsayısıdır. Bu katsayı p_i parçacığının daha önceki kendi en iyi konumundan elde edeceği tecrübelerinden ne ölçüde faydalanacağını belirler. Global optimum olarak adlandırılan p_g ise optimize edilmek istenen fonksiyonda en iyi sonucu veren konum bilgisidir. Bu konumun parçacıkların hızına katkısı c_2 katsayısı ile ayarlanır. Herhangi bir parçacığın yeni hızı hesaplanırken o parçacığın eski hızı da hesaba katılarak (1) numaralı denklemde kullanılır. Parçacıkların hızında hangi bilginin hangi ağırlıkta kullanılacağı ise c_1 katsayısının c_2 katsayısına oranına göre belirlenir. c_1 'den daha büyük c_2 katsayısı hareketin sürüdeki en iyi parçacığa doğru olmasına neden olurken tersi durum parçacığın kendi eski en iyi konumuna doğru gitmesine neden olur. Denklemdeki *rand* isimli 0, 1 aralığındaki rasgele değişkenler kullanılarak da parçacıkların konumlarındaki değişimlerde en iyi sonuca doğru rasgele bir hızla ilerlemeleri sağlanmış olur. Parçacığın hızı v_i^t hesaplandıktan sonra denklem (2)'de parçacığın eski konumu olan x_i^{t-1} ile toplanarak parçacığın yeni konumu hesaplanır.

$$v_i^t = v_i^{t-1} + c_1 * rand * (p_i - x_i) + c_2 * rand * (p_g - x_i) \quad (1)$$

$$x_i^t = x_i^{t-1} + v_i^t \quad (2)$$

Bu işlemler belli bir döngü boyunca veya optimize edilmek istenen fonksiyonda parçacıkların iyileşmesi son bulana kadar tekrarlanır. İşlemler sonlanınca p_g bize aranan optimum değeri verir.

2.1. Ayrık İkili Parçacık Sürüsü

Çözüm uzayının ayrık olduğu durumlarda normal PSO algoritmasına benzeyen DB-PSO önerilmiştir[7]. Yapı olarak normal PSO ile benzer hız denklemine (3) sahiptir. Ancak hesaplama şekli değişiklik göstermektedir. Parçacık yer yenileme denklemi (4) ise tamamen farklıdır. Parçacığın hızı hesaplanırken her bir parçacık ayrı ayrı bitlerin toplamı olarak ifade edilip (3) numaralı denklemde teker teker hesaplanmaktadır. Bulunan hızlar bir sigmoid fonksiyonundan geçirilerek aldığı hız değerlerin 0, 1 aralığında kalması sağlanır. Bu sayede sigmoid fonksiyonundan (5) geçirilen v_{id} 0, 1 aralığında olan rasgele bir *rand* değişkeniyle karşılaştırılarak bitlerin 1 veya 0 değerini alması sağlanır. Daha sonra bu bitlerin birleştirilmesiyle parçacığın gittiği yeni yer hesaplanır.

$$v_{id}^t = v_{id}^{t-1} + c_1 * rand * (p_{id} - x_{id}) + c_2 * rand * (p_{gd} - x_{id}) \quad (3)$$

$$x_{id} = \begin{cases} rand < s(v_{id}) & x_{id} = 1 \\ diger & x_{id} = 0 \end{cases} \quad (4)$$

$$s(v_{id}) = \frac{1}{1 + e^{-v_{id}}} \quad (5)$$

2.1.1. Parçacık Tekrarsız DB-PSO

Sürekli çözüm uzayları için PSO algoritmasında parçacıkların daha önceden aldıkları değerleri tekrarlayıp tekrarlamadıklarının bakılmasına gerek yoktur. Parçacıklar her adımda yeni bir sayısal değer almaktadır. Fakat aynı durum ayrık çözüm uzayları için geçerli değildir. Ayrık seviye veya noktaların bulunduğu durumlarda parçacıkların hızı hareket etmeleri için yetersiz kalabilmekte veya daha önceden her hangi bir parçacık tarafından denenmiş olan bir duruma tekrar gidebilmektedir. Bu durum işlem yükünü gereksiz arttırır ve çözüm uzayının taranma kapasitesinin azalmasına sebep olur.

Bu çalışmada, belirtilen durumu engellemek için parçacıkların tekrarlanmadığı yeni bir yaklaşım ortaya konmuştur. Önerilen Parçacık Tekrarsız DB-PSO yaklaşımı ile çözüm uzayının daha geniş taranarak daha güvenilir çözümlere ulaşılması sağlanmıştır.

3. Parçacık Tekrarsız DB-PSO ile imgelerde eşik seviyesi bulma

Eşik seviyelerinin belirlenmesinde Otsu metodu kullanılmıştır. Otsu metodunun en büyük avantajı eğiticieye ihtiyaç duymaması ve parametrik olmamasıdır. Nesne tanımda ön işlemlerde ve imge filtreleme işlemlerinde geniş bir kullanım alanı vardır. Otsu metodu için imgenin histogramının ortalaması, olasılıkları ve varyansları denklem (6) gibi hesaplanır. Denklem (6)'da k eşik değeri, L en büyük gri ton seviyesi, p_i bir pikselin i seviyesinde olma olasılığı ve ω_i i grupta olma olasılığıdır. Denklem (7) kullanılarak da PSO için kullandığımız uygunluk değeri ölçütü olan λ bulunur[1].

$$\begin{aligned} \mu_0 &= \sum_{i=1}^k i p_i / \omega_0 \\ \mu_1 &= \sum_{i=k+1}^L i p_i / \omega_1 \\ \mu_T &= \sum_{i=1}^L i p_i \\ \sigma_0^2 &= \sum_{i=1}^k (i - \mu_0)^2 p_i / \omega_0 \\ \sigma_1^2 &= \sum_{i=k+1}^L (i - \mu_1)^2 p_i / \omega_1 \end{aligned} \quad (6)$$

$$\begin{aligned}\sigma_w^2 &= \omega_0 \sigma_0^2 + \omega_1 \sigma_1^2 \\ \sigma_B^2 &= \omega_0 (\mu_0 - \mu_T)^2 + \omega_1 (\mu_1 - \mu_T)^2 \quad (7) \\ \lambda &= \sigma_w^2 / \sigma_B^2\end{aligned}$$

İmgeler için en uygun eşik değeri, Otsu metodunda kullanılan λ uygunluk fonksiyonunun aldığı değeri maksimum yapan p_g vermektedir.

Parçacık Tekrarsız DB-PSO denendiği BSDS 500 veri kümesinden seçilmiş olan 3 farklı imge şekil 1,2 ve 3 te gösterilmektedir [9].



Şekil 1: BSDS veri kümesinden kullanılan imge(12003).



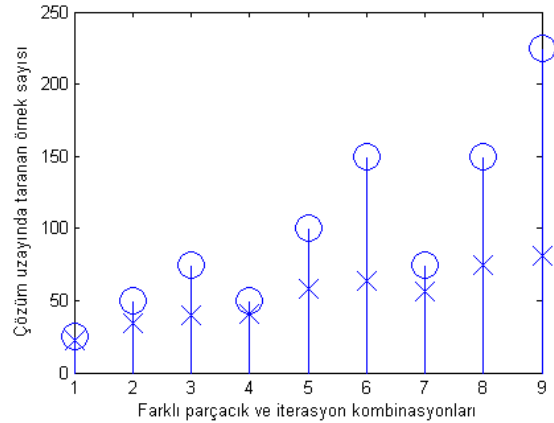
Şekil 2: BSDS veri kümesinden kullanılan imge(86016).

Parçacık Tekrarsız DB-PSO bu imgelerde kullanılarak işlem yükü artırılmadan çözüm uzayının daha geniş bir kısmının taranmasını sağladığı gözlemlenmiştir. Şekil 4'de imge 106005 için bu sayısal olarak gösterilmektedir.



Şekil 3: BSDS veri kümesinden kullanılan imge(106005).

Şekil 4'teki çarpı işaretliler normal DB-PSO yuvarlak içindikiler ise parçacık tekrarsız DB-PSO'yu temsil etmektedir. Şekil 4'te verilen parçacık ve iterasyon kombinasyonları sırasıyla 5 parçacık için 5, 10 ve 15 iterasyonu, 10 parçacık için 5, 10 ve 15 iterasyonu, 15 parçacık için 5, 10 ve 15 iterasyonu ifade etmektedir.



Şekil 4: Farklı kombinasyonlar için çözüm uzayında denenen örnek sayısı.

Çözüm uzayının taranma oranlarının ortalamaları ve standart sapmaları yüzdeler halinde çizelge 1, 2 ve 3'te verilmiştir. Verilen oranlar PSO algoritmalarının bin kez çalıştırılarak bulunan ortalamalarından oluşturulmuştur. Bu sayede karşılaştırmalar yapılırken algoritmadaki rasgele değişkenlerin kullanımından kaynaklanabilecek performans farklılıkları ihmal edilebilir hale getirilmiştir. Çizelgelerde DB PSO ve Parçacık Tekrarsız DB-PSO yaklaşımın farklı parçacık ve iterasyon sayıları için çözüm uzayını tarama oranları verilmiştir.

4. Sonuçlar

Çizelge 1: Denizyıldızı için çözüm uzayı taranma oranı

İmge (12003)	5 İterasyon için	10İterasyon için	15İterasyon için
DB PSO 5 parçacık	%8.75±0.7 0	%13.67±2. 16	%16.29±3. 05
DB PSO 10 parçacık	%15.99±1. 37	%22.54±3. 51	%25.07±4. 81
DB PSO 15 parçacık	%21.95±2. 01	%29.07±4. 61	%31.85±5. 52
Tekrarsız DB PSO 5 parçacık	%9.8	%19.61	%29.41
Tekrarsız DB PSO 10 parçacık	%19.61	%39.22	%58.82
Tekrarsız DB PSO 15 parçacık	%29.41	%58.82	%88.24

Çizelge 2: Bahçe için çözüm uzayı taranma oranı

İmge (86016)	5 İterasyon için	10İterasyon için	15İterasyon için
DB PSO 5 parçacık	%8.79±0.6 8	%13.72±2. 13	%15.93±3. 26
DB PSO 10 parçacık	%16.01±1. 40	%22.54±3. 66	%25.27±4. 92
DB PSO 15 parçacık	%22.11±2. 03	%29.07±4. 75	%31.78±5. 97
Tekrarsız DB PSO 5 parçacık	%9.8	%19.61	%29.41
Tekrarsız DB PSO 10 parçacık	%19.61	%39.22	%58.82
Tekrarsız DB PSO 15 parçacık	%29.41	%58.82	%88.24

Çizelge 3: Penguen için çözüm uzayı taranma oranı

İmge (12003)	5 İterasyon için	10İterasyon için	15İterasyon için
DB PSO 5 parçacık	%8.77±0.6 8	%13.50±2. 17	%15.63±3. 27
DB PSO 10 parçacık	%15.95±1. 44	%22.73±3. 56	%25.03±4. 87
DB PSO 15 parçacık	%22.00±2. 05	%29.13±4. 82	%31.78±5. 96
Tekrarsız DB PSO 5 parçacık	%9.8	%19.61	%29.41
Tekrarsız DB PSO 10 parçacık	%19.61	%39.22	%58.82
Tekrarsız DB PSO 15 parçacık	%29.41	%58.82	%88.24

İmge eşikleme uygulamalarında karşılaşılan yerel optimumlara takılma ve dolayısıyla çözüm uzayının taranma oranının düşük olması probleminin çözümü için Parçacık Tekrarsız DB-PSO yaklaşımı önerilmiştir. Parçacık Tekrarsız DB-PSO'nun başarısı BSDS500 data setinden alınan imgeler üzerinde DB-PSO sonuçları ile karşılaştırmalı olarak incelenmiştir. Elde edilen sonuçlar önerilen yaklaşımın DB-PSO'ya göre çözüm uzayını daha yüksek oranla taradığını göstermektedir. Tek eşik seviyesi için önerilen bu yaklaşım çok seviyeli eşikleme durumuna ve diğer PSO temelli yaklaşımlara da kolayca uyarlanabilir.

5. Kaynaklar

- [1] Otsu, N., "A Threshold Selection Method from Gray-Level Histograms", *IEEE TRANSACTION ON SYSTEMS, MAN, AND CYBERNETICS*, VOL.9., 62-66, 1979.
- [2] Maitra, M. ve Chatterjee, A., "A hybrid cooperative-comprehensive learning based PSO algorithm for image segmentation using multilevel thresholding", *Expert Systems with Applications*, 34, 1341-1350, 2008.
- [3] Chander, A., Chatterjee, A. ve Siarry, P., "A new social and momentum component adaptive PSO algorithm for image segmentation", *Expert Systems with Applications*, 38, 4998-5004, 2011.
- [4] Gao, H., Xu, W., Sun, J. ve Tang, Y., "Multilevel Thresholding for Image Segmentation Through an Improved Quantum-Behaved Particle Swarm Algorithm", *IEEE TRANSACTION ON INSTRUMENTATION AND MEASUREMENT*, VOL.59., 934-946, 2010.
- [5] Nabizadeh, S., Faez, K., Tavassoli, S. ve Rezvanian A., "A Novel Method for Multi-Level Image Thresholding Using Particle Swarm Optimization Algorithms", *2nd International Conference on Computer Engineering and Technology (IC CET)*, 2010, Vol.4 271-275.
- [6] Djerou, L., Dehimi, H., Khelil, N. ve Batouche, M., "Using the BPSO Algorithm in Image Segmentation for Dynamic Thresholding", *Fourth International Conference on Bio-Inspired Computing (BIC-TA '09)*, 2009, 402-407.
- [7] Kennedy, J. ve Eberhart, R.C., "A discrete binary version of the particle swarm algorithm", *Systems, Man, and Cybernetics*, 1997, vol.5, 4104-4108.
- [8] Tasgetiren, M. A. Suganthan, P. N. ve Pan, Q.K., "A Discrete Particle Swarm Optimization Algorithm for the Generalized Traveling Salesman Problem", *GECCO '07 Proceedings of the 9th annual conference on Genetic and evolutionary computation*, 2007, 158-167.
- [9] Martin, D., Fowlkes, C., Tal, D. ve Malik, J., "A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics", *Proc. 8th Int'l Conf. Computer Vision*, 2001, 416-423.