

# Ham Fotokapan Verisinde Hayvan İçeren Fotoğrafların Belirlenmesi

## Detection of Images with Animals in Raw Camera-Trap Data

Ulaş Tekeli, Yalın Baştanlar  
Bilgisayar Mühendisliği Bölümü  
İzmir Yüksek Teknoloji Üniversitesi  
Urla, İzmir, Türkiye  
tekeliulas@gmail.com, yalinbastanlar@iyte.edu.tr

**Özetçe**—Fotokapanlar doğada vahşi hayvanları gözlemlemek için kurulan hareket sensörlü kameralardır. Gelişen teknolojilerle birlikte fotokapan kullanımı ve dolayısıyla sahadan toplanan imge sayısı belirgin bir şekilde artmıştır. Araştırmacıların imgeler üzerinde gruplandırma ve etiketleme yapması için büyük bir iş gücü gerekmektedir. Fotokapanlardan gelen ham veri kümesinde hayvan içermeyen fotoğrafları eleyerek kullanıcının iş yükünü azaltan, evrişimli yapay sinir ağları ve arkaplan çıkarımı kullanılan birleşik bir yöntem öneriyoruz.

**Anahtar Kelimeler**—fotokapan, imge işleme, nesne tanıma, arkaplan çıkarımı, evrişimli yapay sinir ağları, derin öğrenme.

**Abstract**—Camera-traps are motion sensed cameras that are used to observe animals in nature. The number of images collected from camera-traps has increased significantly with developing technologies. A great workload is required for researchers to group and label these images. By employing convolutional neural networks and background subtraction, we propose a system to eliminate the images without animals from raw data sets of camera traps in order to decrease the amount of time spent by researchers.

**Keywords**—camera-trap, image processing, object detection, background subtraction, convolutional neural networks, deep learning.

### I. GİRİŞ

Fotokapanlar doğadaki hayvanların geçiş yolları üzerine yerleştirilen hareket sensörlü kameralardır. Tek bir fotokapan-dan ayda 1000 imge elde edilebilir ve bu imgeler arasında hayvanların yer almadığı, çevresel faktörlerin hareketiyle çekilmiş imgeler olabileceği gibi, bulanık, aşırı parlak ve aşırı karanlık imgeler yer alabilir. Birkaç ayda biriken yüksek sayıda imgelerde hayvan tahlili yapmak çok fazla vakit ve işgücü gerektirir. [1]'te bu probleme yönelik olarak bir veri yönetimi yazılımı önerilmiştir. Yazılım kullanıcının dostu bir arayüzle, imgeleri etiketleme ve dosyaları düzenleme işlevlerini görür. [2]'de ise benzer özelliklere sahip bir veri yönetimi yazılımının üzerine imgeleri internet ortamında depolama özelliği eklenmiştir. Ancak, bu araçlar araştırmacıların tüm imgeleri gözle inceleme

zorunluluğunu ortadan kaldırmaz. Bizim ele aldığımız yaklaşım ise incelenecek imge sayısını azaltmaktır. Bildirimizde, hem arkaplan çıkarımı ile nesne tespiti hem de evrişimli yapay sinir ağları (EYSA) ile nesne bulma teknikleri bir arada kullanılarak hayvan içermeyen imgeleri ayıklayan bir sistem öneriyoruz. Arkaplan çıkarımı, gerçek zamanlı videolardaki hareketli nesnelerin ayırt edilmesi için yaygın bir yöntemdir. Farklı yöntemlerle arkaplan modelini oluşturan ve güncelleyen algoritmalar mevcuttur. Zivkovic [3], Gaussian Karışım Modeli kullanarak arkaplan çıkarımı önermiştir. İmgelerdeki tüm pikseller için belirli sayıda Gaussian dağılımı çıkarılır ve bu dağılımlardan arkaplan pikselleri seçilir. Her ne kadar bu tarz yöntemler video görüntüleri için geliştirilmiş ise de fotokapanların değişmeyen arkaplanı bulunan sahneden değişen zaman aralıkları ile imge topladığı göz önünde bulundurularak, fotokapanlar için de ön plan nesne tespitinde etkili olabileceğini değerlendirdik. Bu amaçla arkaplan çıkarımı öncesi imgeleri gruplayan bir algoritma tasarlayarak çalışmamızda kullandık. Evrişimli Yapay Sinir Ağları (EYSA), özellikle Alexnet [4], ILSVRC [5] yarışmasının imge sınıflandırma kategorisinde birinci gelmesinden itibaren bilgisayarla görünün pek çok alt dalında etkin bir şekilde kullanılmaktadır. Nesne tespiti için geliştirilmiş birçok EYSA yaklaşımı bulunmaktadır. Bunlardan bazıları: OverFeat [6], Faster R-CNN [7] ve YOLO [8]'dur. OverFeat modelinde, EYSA önce sınıflandırıcı olarak eğitilmiş, daha sonra eşiksiz en büyük işlev (softmax) katmanı silinip yerine bağlanım (regression) katmanı eklenerek nesnenin tam konumu kestirilmeye çalışılmıştır. Faster R-CNN [7] modelinde ise, nesne barındırabilecek potansiyel konumlar son evrişim katmanı üzerinden kayan pencere yaklaşımı ile otomatik olarak önerilmiş ve önerilen konumlar içinde nesne aranmıştır. Bu şekilde önemli bir hız artışı sağlanmıştır. YOLO ise sınıflandırıcı bazlı bir sistem kullanmamakta, böylece önerilen çok sayıda konum üzerinde sınıflandırma yapmak yerine, tüm imgeyi tek bir seferde işleme sokarak sistemi hızlandırıp, gerçek zamanlı kullanım hedefler. Çalışmamızda Faster R-CNN'i herhangi bir hayvan bulduğunda pozitif sonuç verecek şekilde eğiterek kullandık.

Bölüm II'de derin öğrenme ile hayvan tespiti, Bölüm III'te arkaplan çıkarımı ile önplan nesnelerin tespit edilmesi anlatılmıştır, Bölüm IV'te yapılan deneylerin sonuçları önce

yukarıdaki iki yöntem için ayrı ayrı verilmiş ardından yöntemler birleştirildiğindeki sonuçlar aktarılmıştır. Bölüm V'te ise vargılara yer verilmiştir.

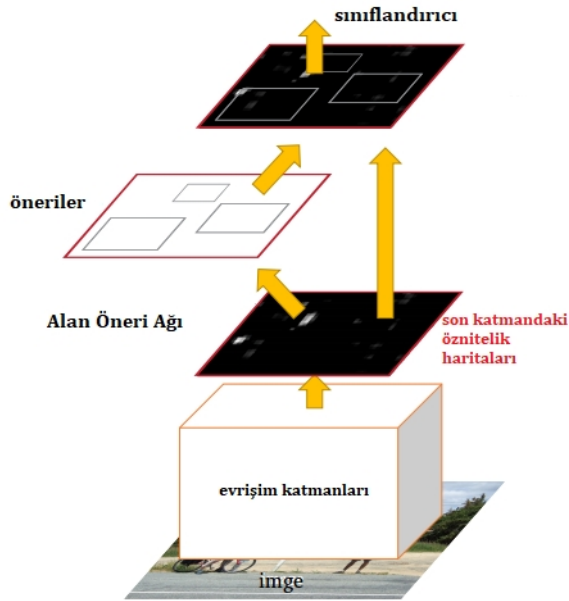
## II. DERİN ÖĞRENME İLE HAYVAN TESPİTİ

Derin ağlar ile nesne tespiti çalışmaları için Faster R-CNN yöntemi seçilmiştir. Faster R-CNN, evrişimsel katmanların üzerine eklenen ve birisi alan önerilerinde bulunan, diğeri ise bu öneriler üzerinde sınıflandırma yapan iki başlıktan oluşur (Şekil 1). Bu iki başlık son evrişimsel katmanı ortak olarak kullanır. Alan öneri ağı, paylaşılan son evrişimsel katman üzerinde kayan pencere yaklaşımı uygulayarak, her imge için farklı boyutlarda ve farklı ölçeklerde yaklaşık 20000 alan önerisinde bulunur ve bu alanlara nesnelik skoru atar. Daha sonra 20000 sayısını düşürmek için bu alanlar üzerinde örtüşen alanlardan düşük skorluların elenmesi (non-maximum supresion) ve skor eşik değerleri uygular. 300'e kadar indirilen öneri sayısı, sınıflandırıcı başlığa gönderilir. Sınıflandırıcı başlık bu öneriler üzerinde sınıflandırma ve kapsayan kutu bağlanımı (bounding-box regression) yaparak en son konum tahmininde bulunur.

Faster R-CNN'i kullanma nedenlerimiz, kodlarının daha ulaşılabilir ve anlaşılabilir oluşu ve YOLO gibi daha güncel yaklaşımların gerçek zamanlı tespitte izin verse de başarı düzeyi olarak Faster R-CNN'in çok üzerine çıkamamasıdır. Belirli bir hayvan türü hedeflenmediğinden Faster R-CNN iki sınıflı (hayvan, arkaplan) olarak eğitilmiştir. Eğitim setindeki bütün hayvanlar pozitif sınıf örneklerini oluşturmuştur.

## III. ÖNPLAN NESNE İÇEREN İMGELERİN ARKAPLAN ÇIKARIMI İLE TESPİTİ

Arkaplan çıkarımı için [10]'da karşılaştırılan yöntemlerden biri olan Gaussian Karışım Modeli [3] seçilmiştir. Her piksel koordinatı için 'arkaplan' Gauss dağılımları olarak modellenir, bu dağılımlara yeterince dahil olmayan piksel değerleri önplan



Şekil 1: Faster R-CNN yapısı [7].

nesnesi olarak kabul edilir. İki-modlu arkaplanlarla uyumlu çalışabilmesi bu seçimde etkili olmuştur. Arkaplan çıkarımı sonrası elde edilen imgeler bir dizi morfolojik işlemden geçirilir. Morfolojik işleme maruz kalmış imgeler bağlantılı bileşen algoritmasıyla önplanı ifade eden alan bileşenlere ayrılır ve bileşenlerin alanları hesaplanır. Eşik değerin üzerinde alana sahip bileşenler önplan nesnelere olarak tanınır. Şekil 2a'da başarılı şekilde nesne niteliği kazanan bir bileşen görülmektedir.

Arkaplan çıkarma algoritmaları genellikle saniyede yüksek sayıda imge içeren videolar üzerinde çalışır, yani değişim miktarı bir önceki kareye göre oldukça az olan imgeler için tasarlanmıştır. Sistemimizdeki hedef, aynı fotokapandan değişen zamanlarda alınan imgeler üzerinde arkaplan çıkarımı yapmaktır. İki imge arasında süre az olduğu halde ışık değişiminin çok olduğu ve süre çok olduğu halde ışık şartlarının benzer olduğu (ayrı günlerin aynı saatinde alınmış fotoğraflar gibi) örnekler mevcuttur. İmgeleri bir video dizisi gibi görüp, aralarındaki farklılık en az iken işleme sokmak gereklidir. Bu amaçla tasarladığımız algoritma aşağıda açıklanmıştır.

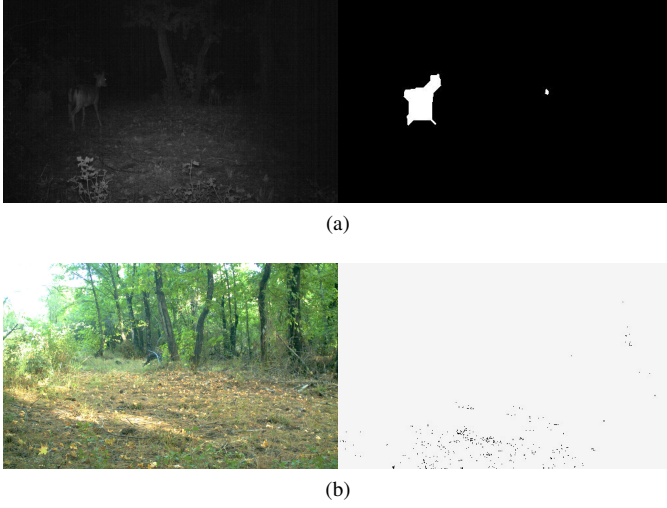
Öncelikle, piksel değerlerine bakılarak imgeler arasında benzerlik metriği oluşturulur. İki imge arasında benzerlik ölçülürken piksel piksel karşılaştırma yapılır. Eğer değer farkları, deneyler sonucu belirlenen eşik değerinin altında ise fark yok, üzerindeyse fark var kabul edilir. Fark yok kabul edilen piksel sayısının toplam piksel sayısına oranı benzerlik metriğini ifade eder. İlk imge için en benzer imge bulunduktan sonra, ikinci sıraya konur. Daha sonra bu işlem her yeni gelen imge için tekrarlanır. Bu şekilde sıralanan imgeler arkaplan çıkarımında daha yüksek performans göstermektedir. İmgeler sıralanmaz ise yaşanan bir problem Şekil 2b'de gözlenebilir.

Sıralanan imgeler, ayrıca ışık şartlarının çok değiştiği (benzerliğin düştüğü) yerlerden ayrılarak kümelendir. Gece çekilen imgeler genellikle tek bir küme altında toplanırken, gündüz çekilen imgelerin birkaç farklı küme altında toplandığı gözlemlenmiştir (Şekil 3). Sonrasında ise her küme kendi içlerinde arkaplan çıkarma yöntemi uygulanır. Diğer bir deyişle, öğrenilen arkaplan modeli küme değiştiğinde unutulur. Arkaplan çıkarımına dair akış şeması Şekil 4'te verilmiştir.

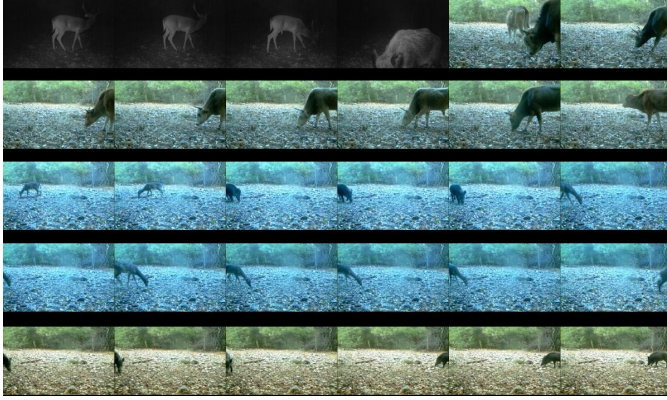
## IV. DENEYLER

T.C. Orman ve Su İşleri Bakanlığı'ndan elde edilen yaklaşık 40000 imge taranarak, müdahale edilmemiş (ham) klasörlerden (fotokaparlardan) veriseti oluşturuldu. Hayvan içeren imgelerde, LabelImg [9] adlı etiketleme aracı kullanılarak hayvanlar kutu içerisine alındı ve Pascal VOC formatında etiketler elde edildi. Eğitim setinin, mümkün olduğunca çok sayıda hayvan, çok sayıda ışıklandırma biçimi ve çok sayıda poz içermesine, kısmen imge içerisine giren hayvanlara yer verilmesine ve farklı arkaplana sahip sahnelerden oluşmasına dikkat edildi.

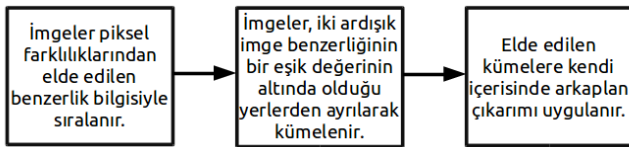
Çalışmamızda hangi hayvan içerdiğine bakmaksızın hayvanlı imgelerin tutulması diğer imgelerin ise elenmesi hedeflendiğinden EYSA eğitimi sırasında verisetindeki tüm hayvanlar tek bir pozitif sınıf altında etiketlenmiştir. Bazı hayvanların çok az yer alması ve eğitim setinde bulunan bir hayvan bilgisinin bulunmayan başka bir hayvanı bulmak için de işe yarayabilecek olması açısından da tek sınıf etiketleme



Şekil 2: Ham verisetinde gruplama yapılmadan ardarda arkaplan çıkarma işlemine giren iki imge (her satırın solunda). Sağdaki imgeler önplan imgesini göstermektedir. (b) imgesi (a) imgesinden hemen sonra işleme girdiği için arkaplan öğrenme işlemi gerçekleşmemiş, imgede nesne yer almamasına rağmen (a)'ya göre neredeyse tüm piksellerde değişim olduğundan hatalı sonuç alınmıştır. Önerdiğimiz imge gruplama algoritması ile bu hata giderilmektedir.



Şekil 3: Sıralanan imgelerin arkaplan sahnesine göre oluşturdukları kümeler. Sol üstten başlayarak 1., 5., 13. ve 25. imgelerde yeni kümeler başlamıştır.



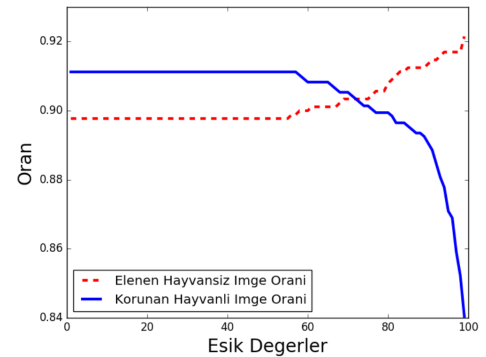
Şekil 4: Arkaplan çıkarma akış şeması

uygundur. Negatif örnekler (arkaplan) ise aynı imgelerin hayvan içermeyen bölümlerinden otomatik olarak elde edilmiştir. Toplam 930 eğitim imgesi kullanılmıştır.

Test için yine mümkün olduğunca çeşitli hayvan ve ışıklandırma biçimi barındırmaya dikkat ederek yaklaşık 2000 imge içeren bir veriseti oluşturuldu. Derin öğrenme algoritması fotokapanın gördüğü arkaplanı da öğrendiğinden test setinde eğitim setinde kullanılan bir fotokaptan hiç fotoğraf kullanılmadı. Test seti, yüksek oranda hayvan içeren test grubu ve düşük oranda hayvan içeren test grubu olarak ikiye ayrıldı. Bu ayrımın sebebi, her bir fotokapan verisinin bu iki karakterden birine uyduğunun gözlenmesidir. Deneilerde yer alan Veriseti 1 düşük sayıda hayvan içeren imge barındırırken, Veriseti 2 yüksek sayıda hayvan içeren imge barındırmaktadır. Başarı, hayvan içeren imgelerin korunma ve hayvan içermeyen imgelerin elenme oranı ile ölçülmüştür. Sırasıyla, derin öğrenme ile, arkaplan çıkarımı ile ve birleşik yöntem ile deney sonuçları aşağıda aktarılmıştır.

#### A. Derin Öğrenme ile Hayvanlı İmge Tespiti Deneyleri

Eğitilen EYSA'nın test seti üzerindeki sonuçları Şekil 5'te gösterildiği üzere farklı eşik değerleri ile grafiğe dökülmüştür. Nesnelik skoru eşik değerinin altında olan kutular elenecek şekilde hayvansız imgelerde 'elenme oranı' ve hayvanlı imgelerde 'korunma oranı' görülmektedir. Eşik değer 0.6 olana kadar grafikte bir değişiklik gözlenmemesi, sistemin ürettiği nesne kutularına 0.6'dan düşük skor vermediğini gösterir. Buradan hareketle, eşik değeri 0.5 seçilerek tüm test seti için derin öğrenme deney sonuçları Tablo 1'e aktarılmıştır. Sonuçların tablo ile verilmesi aynı zamanda arkaplan çıkarımı yöntemi ile karşılaştırmayı sağlamaktadır. Derin öğrenme % 90 dolayında elenme ve korunma oranı gibi iyi sonuçlar verse de deneyler sırasında farkedilen bir eksiklik, kayalık arazide bulunan taşlar için yanlış alarmlar vermesi, yahut ağaçlık ve taşlık dokuya benzer dokuda hayvanları arkaplan nesnesi olarak nitelendirmesidir. Bu yanlışlar arkaplan çıkarımı tabanlı yöntem ile giderilebilir. Arkaplan çıkarımındaki değişim ölçen nitelik, arkaplanda yer almayan ve ağaçsı dokuya sahip bir hayvanı yine de ayırt edecektir. Aynı şekilde arkaplanda yer alan kayalıkları, sahnede değişim olmadığından hayvan olarak nitelenecektir. Şekil 6'da bu durumların birer örneği verilmiştir.



Şekil 5: Farklı eşik değerleri için derin öğrenme deney sonuçları





(a)



(b)

Şekil 6: Derin öğrenme yaklaşımı ile doğru tespit edilen hayvanlar (a)'da yer alırken, (b)'de soldaki imgede su birikintisi önünde bulunan iki domuz arkaplan dokusuna benzedikleri için ayırt edilememiş, (b)'de sağdaki imgede ise hayvan dokusuna benzer dokudaki taşlar ve ağaçlar etiketlenmiştir.

#### B. Arkaplan Çıkarımı ile Hayvanlı İmge Tespiti Deneyleri

Bölüm III'te anlatıldığı üzere önerdiğimiz ışık değişimlerine göre imge gruplama yöntemi ve arkasından Gaussian Karışım Modeli [3] ile arkaplan modellemesi sonucu hayvan olduğu tespit edilen imgelerle Tablo II'deki sonuçlar alınmıştır. Korunan hayvanlı imge sayısı derin öğrenme yöntemine yakın çıkarken elenen hayvansız imge oranı gerilemiştir.

#### C. Birleşik Yöntem ile Hayvanlı İmge Tespiti Deneyleri

Arkaplan çıkarımı yönteminin sonuçları, derin öğrenme deney sonuçlarına göre daha başarısız görünmekle beraber, yapılan hatalar farklı imgelerde olabileceğinden her iki yöntemin sonuçlarını karar aşamasında birleştirdiğimiz üçüncü bir deney gerçekleştirdik. Her iki yöntemin de eksi sonuç verdiği imgeler elenirken, herhangi bir yöntemin hayvan var demesi imgenin korunması için yeterli kılındı. Böylece iki yöntemin kendi içerisinde kaçırdığı hayvanlı imgeler kaçırılmamış oldu. Bu durum elenen hayvansız imge sayısında düşmeye sebep

Tablo I: Derin Öğrenme ile Elenen ve Korunan İmge Oranları

Verisetleri	Hayvan İçeren İmge Sayısı	Hayvan İçermeyen İmge Sayısı	Elenen Hayvansız İmge Yüzdesi	Korunan Hayvanlı İmge Yüzdesi
Veriseti1	76	631	90.8	51.3
Veriseti2	941	307	86.9	94.1
TOPLAM	1015	938	89.5	91.1

Tablo II: Arkaplan Çıkarımı Yöntemi ile Elenen ve Korunan İmge Oranları

Verisetleri	Hayvan İçeren İmge Sayısı	Hayvan İçermeyen İmge Sayısı	Elenen Hayvansız İmge Yüzdesi	Korunan Hayvanlı İmge Yüzdesi
Veriseti1	76	631	60.6	75
Veriseti2	941	307	46.9	91.9
TOPLAM	1015	938	56.1	90.8

Tablo III: Birleşik Yöntem ile Elenen ve Korunan İmge Oranları

Verisetleri	Hayvan İçeren İmge Sayısı	Hayvan İçermeyen İmge Sayısı	Elenen Hayvansız İmge Yüzdesi	Korunan Hayvanlı İmge Yüzdesi
Veriseti1	76	631	60.0	89.4
Veriseti2	941	307	43.3	99.9
TOPLAM	1015	938	54.5	99.1

oldu, ancak elenen imge oranı %54.5 iken korunan hayvanlı imge oranı olarak %99.1'e ulaşıldı (Tablo III).

## V. VARGILAR

Çalışmamızda derin öğrenme ve arkaplan çıkarımı tabanlı yöntemlerle ham fotokapan verisinde hayvan içeren tüm imgeleri korurken araştırmacılarının bakacakları hayvansız imgeleri mümkün olduğunca elemeye çalıştık. Birleşik yöntemimiz ile 1015 hayvanlı imge %99.1 oranında doğru etiketlenirken, hayvansız 938 imgede %54.5 oranında elenme sağlanmıştır. Hayvan içeren verisinde kaçırılan %1'lik dilimi incelediğimizde kaçırılan hayvanların komşu fotoğraflarda (fotokapan hareket devam ettiği sürece fotoğraf çektiğinden) görüldüğünü fark ettik. Bu durumda deney sonucu olarak hiçbir hayvan bireyi kaçırılmadan 500 dolayında hayvansız fotoğrafın elendiği söylenebilir. Bu sonuç, araştırmacıların gözle inceleme zorunluluğunu olan fotoğraf sayısında önemli bir düşüş sağlanabileceğini göstermektedir.

## TEŞEKKÜR

Bu bildiriye katkıları için TÜBİTAK ARDEB 115E918 no'lu proje kapsamında desteklenmiştir. Yazarlar, T.C. Orman ve Su İşleri Bakanlığı Doğa Koruma ve Milli Parklar Genel Müdürlüğü'ne paylaşılan fotokapan verisi için teşekkür eder.

## KAYNAKÇA

- [1] Krishnappa Y. S. and Turner W. C., "Software for minimalistic data management in large camera trap studies", in *Ecologic Informatics*, 24:11-16, 2014.
- [2] Fegraus E.H., Lin K., Ahumada J.A., Baru C., Chandara S., Youn C., "Data acquisition and management software for camera trap data: A case study from the TEAM network", in *Ecologic Informatics*, 6(6), 2011.
- [3] Zivkovic Z., "Improved adaptive Gaussian mixture model for background subtraction", in *International Conference on Pattern Recognition*, 2004.
- [4] Krizhevsky, A., Sutskever, I. and Hinton, G.E., "Imagenet classification with deep convolutional neural networks". In *Advances in neural information processing systems* (pp. 1097-1105), 2012.
- [5] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M. and Berg, A.C., "Imagenet large scale visual recognition challenge". *International Journal of Computer Vision*, 115(3), pp.211-252, 2015.
- [6] Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R. and LeCun, Y., "Overfeat: Integrated recognition, localization and detection using convolutional networks". arXiv preprint arXiv:1312.6229, 2013.
- [7] Ren, S., He, K., Girshick, R. and Sun, J., "Faster r-cnn: Towards realtime object detection with region proposal networks". In *Advances in neural information processing systems* (pp. 91-99), 2015.
- [8] Redmon, J., Divvala, S., Girshick, R., Farhadi, A., "You Only Look Once: Unified, Real-Time Object Detection". In *IEEE Conf. on Computer Vision and Pattern Recognition(CVPR)*, 2016.
- [9] "LabelImg" : <https://github.com/tzutalin/labelImg>.
- [10] Sobral A. and Vacavant A., "A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos", in *Computer Vision and Image Understanding*, 122:4-21, 2014.