

Deep Learning Based Human Detection in Camera-trap Images

Emrah ŞİMŞEK^{*1} Barış ÖZYER², Gülşah Tümüklü ÖZYER²

¹Erzurum Teknik Üniversitesi, Bilgi İşlem Daire Başkanlığı, Erzurum, Türkiye

²Bilgisayar Mühendisliği, Atatürk Üniversitesi, Erzurum Türkiye

Keywords:

Object Detection
Deep Learning
YOLO
Camera-Trap

Abstract

Camera-traps are embedded systems used to obtain images of things living in natural environment. These images, which the camera-traps recorded at certain frequencies with the motion trigger, include significant difficulties such as complexity, background motion, light intensity change, conclusion, affine and 3D transformation, scaling change, fragmented object finding. Because of the limited durability of traditional pattern recognition methods against these difficulties for identifying and detecting automatically living things in images, for this purpose leading to use deep learning-based methods in which the deep features of images can be extracted. YOLO architecture is a deep learning method with high detection and speed performance used in real-time object detection. In this study, the success of the YOLO architecture, which was trained with the PASCAL VOC data set, in the camera-trap data set with many difficulties was evaluated. In the camera-trap data set, high detection success was achieved.

Foto-Kapan Görüntülerinde Derin Öğrenme Tabanlı İnsan Tespiti

Anahtar Kelimeler:

Nesne Tespiti
Derin Öğrenme
YOLO
Foto-Kapan

Özet

Foto-kapanlar doğal ortamda yaşayan canlılara ait görüntülerin elde edilmesi amacıyla kullanılan gömülü sistemlerdir. Foto-kapanların hareket ile tetiklendikten sonra belirli frekanslarda kaydettiği bu görüntüler, karmaşıklık, arka plan hareketi, ışık şiddeti yetersizliği, kapanma, afin ve 3D dönüşümü, ölçekleme değişimi, parçalı nesne bulunması gibi önemli zorluklar içermektedir. Görüntülerdeki zorlukların giderilerek canlıların otomatik olarak tespit edilmesi için kullanılan geleneksel örüntü tanıma yöntemlerinin bu zorluklara karşı sınırlı dayanıklılığa sahip olmaları nedeniyle, görüntülere ait derin özniteliklerin çıkarılabildiği derin öğrenme tabanlı yöntemlerin tercih edilmesine neden olmaktadır. YOLO mimarisi, gerçek zamanlı nesne tespitinde kullanılan yüksek tespit ve hız performansına sahip bir derin öğrenme yöntemidir. Bu çalışmada PASCAL VOC veri seti ile eğitilmiş YOLO mimarisinin çok sayıda zorluk içeren foto-kapan veri setindeki tespit başarısı değerlendirilmiştir. Foto-kapan veri setinde yapılan uygulamada yüksek tespit başarısı elde edilmiştir.

1 GİRİŞ

Derin öğrenme, hiyerarşik mimarileri kullanarak verideki üst seviye soyutlamaları öğrenmeye çalışan makine öğrenmesi alt alanıdır [1]. Geleneksel makine öğrenme tekniklerinin, doğal verileri ham formlarında işleme yetenekleri sınırlıdır [2]. Derin öğrenme yöntemleri, her biri temsili daha yüksek ve daha soyut düzeyde bir temsile dönüştüren basit ama doğrusal olmayan modüller oluşturularak, çoklu temsil seviyelerine sahip temsil-öğrenme yöntemleridir. Derin öğrenmenin en önemli özelliği, öznitelik öğrenme işleminin insanlar tarafından tasarlanan katmanlar yerine, genel amaçlı bir öğrenme prosedürü kullanılarak veriler üzerinden gerçekleştirilmesidir [3]. Derin öğrenme kullanımının son yıllarda yaygınlaşmasının temel nedenleri işlemci kapasitelerinin artması, donanımların maliyetlerinin düşmesi ve makine öğrenmesi algoritmalarındaki gelişmeler olarak sıralanabilmektedir [1].

*e-Posta: emrah.simsek@erzurum.edu.tr



Şekil 1. Işık Şiddeti Yetersizliği, Kapanma, Ölçekleme Değişimi, 3D ve Afin Dönüşümü, Parçalı Nesne Bulunması Zorlukları Barındıran Foto-kapan Görüntüleri

Doğal ortamda canlıların görüntülenmesi amacıyla kullanılan foto-kapanlar üzerinde bilgisayar destekli otomatik nesne tespit ve tanıma çalışmalarında, Şekil 1’de görüldüğü üzere ışık şiddeti yetersizliği, kapanma, ölçekleme değişimi, 3d ve afin dönüşümü, parçalı nesne bulunması zorlukları gibi oldukça fazla problem ile karşılaşmaktadır [4, 5, 6, 7, 8, 10]. Bu görüntülerde nesne tespiti için kullanılan yerel veya genel öznitelikler ile örüntü tanıma yöntemleri, kısmen kullanışlı olsa da, insan müdahalesi ve kontrolü gerektirmektedir. Derin öğrenme tabanlı nesne tespit etme ve nesne tanıma çalışmalarında yüksek başarıyla elde edilmesi, bu yöntemlerin otomatik nesne tespit etme ve tanıma çalışmalarının yapıldığı foto-kapan görüntülerinde de kullanılmaya başlanmasına neden olmaktadır. Bu amaçla son yıllarda foto-kapan görüntüleri üzerinde canlı tespiti, tür tanıma [4, 6, 7, 8], vahşi hayvan tanıma [5, 10], canlı sayma gibi işlemler otomatik olarak yapılmaya çalışılmaktadır. Bu çalışmalarda derin öğrenme mimarilerinden Lite Alexnet, VGG-16, NiN(Network in Network), GoogleNet, ResNet, ResNet-50 ve ResNET152 kullanılmıştır.

Derin öğrenme tabanlı evrişimli sinir ağlarında nesne tespiti amacıyla kullanılan OverFeat, Faster R-CNN, ve YOLO mimarileri bulunmaktadır. Bu çalışmada, konumlandırma işleminin etkili ve hızlı bir şekilde yapıldığı evrişimli sinir ağı mimarisi olan YOLO mimarisi ve PASCAL VOC 2012 veri seti ile 80 sınıf için eğitilmiş ağırlıklar kullanılarak, doğal foto kapan görüntülerinde insan tespiti gerçekleştirilmiştir. Veri setindeki zorluklara karşı YOLO mimarisinin performansı incelenmiş ve yorumlanmıştır. Çalışmamızın 2. bölümünde nesne tespiti için kullanılan evrişimli sinir ağı mimarisi olan YOLO anlatılmıştır. 3. bölümde kullandığımız veri seti, 4. bölümde yöntemin nesne tespit ve konumlandırmadaki performansı bulunmaktadır. 5. bölümde ise çalışmada elde edilen sonuçlar yorumlanmıştır.

2 MATERYAL VE METOD

Evrişimsel Sinir Ağları (Convolutional Neural Networks) çoklu katmanların etkili bir şekilde eğitilerek kullanıldığı derin öğrenme yaklaşımlarından biridir. Bir evrişimsel sinir ağı Şekil 2’de gösterildiği üzere evrişimsel katmanlar, havuzlama katmanları ve tam bağlı katmanlardan oluşmaktadır [1].



Şekil 2. Evrişimli Sinir Ağları Mimarisi [1]

2.1 Evrişim Katmanı (Convolutional Layer)

Evrişim katmanlarında, çeşitli çekirdekler kullanılarak görüntünün çeşitli öznitelik haritaları oluşturulmaktadır [1, 11]. Matematiksel olarak, (i, j) konumunun m. katman n. öznitelik haritasının öznitelik değeri olan $z_{i,j,n}^m$ aşağıdaki denklem 1 ile hesaplanmaktadır [11]:

$$z_{i,j,n}^m = w_n^{mT} x_{i,j}^m + b_n^m \quad (1)$$

w_n^m ve b_n^m sırasıyla m. katmanın n. filtresinin ağırlık vektörü (weight) ve sapma (bias) terimidir ve $x_{i,j}^m$ ise m. katmanın (i,j) konumunda ortalanmış girdi parçasıdır [11].

Aktivasyon fonksiyonu, doğrusal olmayan özniteliklerin algılanabilmesi amacıyla kullanılmaktadır. Evrişimli öznitelik $z_{i,j,n}^m$ için aktivasyon değeri olan $a_{i,j,n}^m$ aşağıdaki denklem 2 ile hesaplanmaktadır:

$$a_{i,j,n}^m = a(z_{i,j,n}^m) \quad (2)$$

Uygulamalarda genellikle sigmoid, tanh ve ReLU aktivasyon fonksiyonları kullanılmaktadır. ReLu (Rectified Linear Unit), aşağıdaki denklem 3 ile hesaplanmaktadır, $z_{i,j,k}$ k. kanaldaki (i,j) konumundaki aktivasyon girişidir [11]:

$$a_{i,j,k} = \max(z_{i,j,k}, 0) \quad (3)$$

2.2 Evrişim Katmanı (Pooling Layer)

Genel olarak bir havuzlama katmanı, evrişimli bir katmandan sonra gelmekte ve öznitelik haritalarının ve ağ parametrelerinin boyutlarını azaltmak için kullanılmaktadır [1, 11]. $a_{i,j,n}^m$ öznitelik haritası için havuzlama fonksiyonu aşağıdaki denklem 3 şeklinde uygulanmaktadır:

$$y_{i,j,n}^m = \text{pool}(a_{f,g,n}^m), \forall (f,g) \in R_{ij} \quad (4)$$

R_{ij} , (i,j) konumunun yerel komşuluklarına ifade etmektedir [11].

2.3 Tam Bağlı Katmanlar (Fully Connected Layers)

Son havuzlama katmanının ardından 2 boyutlu öznitelik haritalarının daha fazla öznitelik gösterimi için 1 boyutlu öznitelik vektörüne dönüştüren birkaç tam bağlı katman kullanılmaktadır [1].

Evrişimli sinir ağlarının çıktısı katmanı olan son katmanında, sınıflandırma görevi için genellikle softmax operatörü veya CNN ile birleştirilebilen SVM kullanılmaktadır. Belirli bir görev için en iyi parametreler, bu görevde tanımlanan bir kayıp fonksiyonunun en aza indirilerek elde edilebilmektedir. N adet istenen girdi-çıkı ilişkisi $\{(x^{(n)}, y^{(n)})\}; n \in [1, \dots, N]\}$ için, $x^{(n)}$, n. girdi verisidir, $y^{(n)}$ hedef etiketleridir ve $o^{(n)}$ ise CNN'in çıkışıdır. Bu ilişkilerin CNN kaybı aşağıdaki denklem 5 ile hesaplanmaktadır [11]:

$$L = \sum_{n=1}^N l = (\theta; y^{(n)}, o^{(n)}) \quad (5)$$

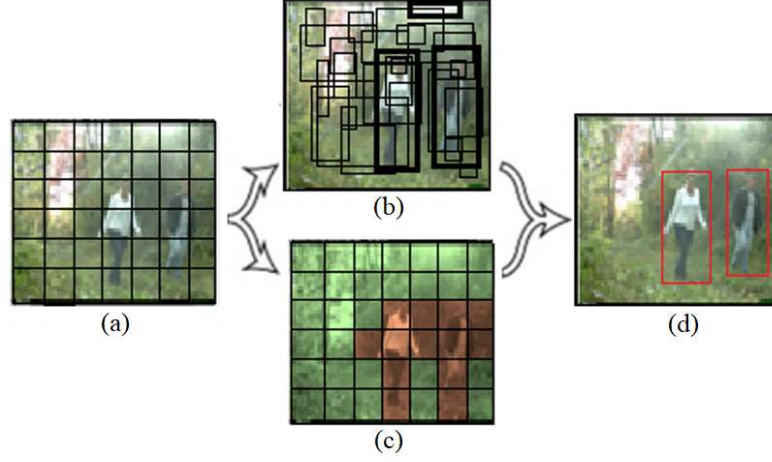
Evrişimli sinir ağlarının popüler hale gelmesi, 5 evrişim, 3 tam bağlı katmandan oluşan AlexNet [12]'in ImageNet yarışmasında birinci olması ile başlamıştır. Daha sonra önerilmiş VGG [22] isimli mimaride ise, daha fazla evrişimli katman eklenerek, tüm evrelerde küçük evrişim filtrelerinden yararlanarak sinir ağının derinliğinin artırılmıştır. Bu gelişmeden sonra 22 katmanı bulunan daha derin bir evrişimli sinir ağı olan GoogleNet [13] önerilmiştir. Bu gelişmelerden sonra aynı modeller ile nesne tespit etme ve anlamsal bölütleme işlemlerinin yapılabilmesi amacıyla yeni yapılar türetilmiştir. En yaygın olanları RCNN (Region based Convolutional Neural Network) olarak ifade edilen bölge tabanlı evrişimli sinir ağları ve FCN (Fully Convolutional Network) tamamen evrişimli ağıdır [1]. RCNN'nin ana fikri, bölgesel olarak çoklu nesne önerileri oluşturmak, evrişimli sinir ağları kullanarak bu önerilerin özniteliklerini çıkarmak ve bu pencereleri doğrusal SVM ile sınıflandırmaktır [2].

2.4 YOLO (You Only Look Once: Unified, Real-Time Object Detection)

YOLO (You Only Look Once: Unified, Real-Time Object Detection), GoogleNet mimarisinden esinlenilmiş, görüntüdeki nesnelere ait uzamsal olarak ayrılmış sınırlayıcı kutularının sınıf olasılıklarının regresyon problemi olarak alındığı, tek bir sinir ağında bu sınıf olasılıklarının tahmin edildiği bir evrişimli sinir ağları mimarisidir [14]. Evrişimsel sinir ağları kategorisinden olan bu mimaride [1] standart olarak saniyede 45 görüntü, hızlı YOLO mimarisinde ise saniyede 155 görüntü gerçek zamanlı işlenebilmektedir. Literatürdeki en gelişmiş yöntemlere göre daha az yerleştirme hatası yapmaktadır [14]. Mimarinin son katmanında doğrusal bir aktivasyon fonksiyonu kullanılırken, diğer tüm katmanlarda denklem 6'da gösterilen sızdıran düzeltilmiş (leaky rectified) doğrusal aktivasyon kullanılmaktadır [14]:

$$\emptyset(x) = \begin{cases} x, & \text{if } x > 0 \\ 0.1x, & \text{otherwise} \end{cases} \quad (6)$$

YOLO mimarisinde, 24 adet evrişim katmanı, bu katmanların devamında 2 adet tam bağlı katman bulunmaktadır. Alternatif 1x1 evrişimli katmanlar, önceki katmanlardan gelen özneliklerin boyutunu azaltmaktadır [14].



Şekil 3. YOLO Mimarisi Nesne Tespiti

Mimarinin son katmanında Şekil 3'te görüldüğü üzere hem nesnelere ait sınıf olasılıkları, hem de bu sınıflara ait sınırlayıcı kutu koordinatları öngörülmektedir. Şekil 3'ten anlaşıldığı üzere, öncelikle görüntü Şekil 3-a'daki ızgara biçiminde bölünmektedir. Görüntüdeki nesnelerin yerleştirilmesi için Şekil 3-b'de gösterilen hedef nesnelere ait konum tahminleri ve Şekil 3-c'de gösterilen birliktelik ifade eden komşu piksellerin belirlendiği ısı haritası kullanılmaktadır. Her bir ızgara parçasına denk gelen Şekil 3-b konumlarıdaki nesnelere ait en yüksek sınıf değeri, o sınırlayıcı kutuya etiket olarak atanmakta olup, çakışan diğer kutular göz ardı edilmektedir.

YOLO'nun 3 numaralı sürümünde tespit başarısının artırılması ve tespit süresinin azaltılması amacıyla evrişimli sinir ağlarında sınıflandırma işlemi için genellikle tercih edilen Softmax katmanı yerine lojistik sınıflandırıcı kullanılmaktadır. YOLO mimarisinde sınırlayıcı kutular 3 farklı ölçekte tahmin edilmektedir. Temel öznelik çıkarma işlemine ek olarak eklenen birkaç evrişimli katman ile sınırlayıcı bir kutu, nesnellik ve sınıf kestirimleri çıkarımları gerçekleştirilmektedir. Sınırlayıcı kutu tahminlerinde önceki sürümlerde olduğu gibi k-ortalama metodunu kullanılmaktadır [15].

Nesne içermeyen tahminlerin elenebilmesi için güven parametresi ve üst üste gelen sınırlayıcı kutuların kabul edilmesi için eşik parametresi olmak üzere 2 parametre kullanılmaktadır. Sınırlayıcı kutular, en yüksek IoU (Intersection over Union) değerine dayanan bir tahmin işlemi kullanılarak etiketlenmektedir. Nesne tespiti için sinir ağı, PASCAL VOC 2007 ve 2012'den eğitim ve doğrulama veri setleri hakkında yaklaşık 135 devir boyunca eğitilmektedir [14]. Darknet sinir ağının kullanıldığı eğitim aşamasında çok ölçekli eğitim, veri çoğaltma ve yığın normalizasyon gibi standart işlemlerin tamamı kullanılmaktadır [15]. Çalışmamızda 80 adet nesne için PASCAL VOC 2012 ile eğitilmiş ağırlık değerleri kullanılarak, foto-kapan görüntülerindeki insan tespiti ve konumlandırma başarısı incelenmiştir.

2.5 Foto-kapan Veri Seti

Foto-kapanlar kullanılarak doğal ortamdan elde edilmiş Şekil 1 ve Şekil 4'te örnekleri gösterilen veri seti [9], içerisinde 100 farklı arka plana sahip, gece ve gündüz kaydedilmiş, 2048x1536 boyutlarında toplam 2880 görüntü içermektedir. Veri setindeki görüntülerin 2150'sinde, bu görüntülerde bulunan nesnelere ait tür, konum ve boyut bilgileri kesin referans (ground-truth) olarak bulunmaktadır.



Şekil 4. Foto-kapan Veri Seti

Veri setinde kesin referans içerisinde etiketlenen görüntülerin 314'ünde sınırlayıcı kutuları belirlenmiş tam veya parçalı insan görüntüsü bulunmakta olup, 1836'ında insan bulunmamaktadır. Veri setinde etiketlenmemiş 2 bölümünde ise 730 görüntü bulunmaktadır ve bu görüntüler tarafımızca yalnızca insan olup olmadığı tespit edilerek etiketlenmiştir.

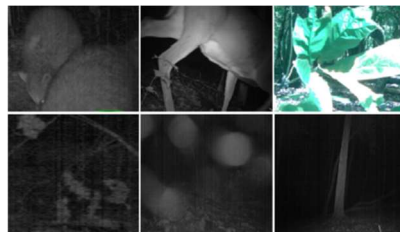
3 BULGULAR

Çalışmada YOLO yöntemine ait güven parametresi olarak 0.5 değeri, eşikleme parametresi olarak 0.3 değeri kullanılmıştır. Veri seti üzerinde yapılan tespit çalışmasında Tablo 1'de görüldüğü üzere 314 insan içeren görüntünün 297'si tespit edilmiştir. Tespit edilemeyen 17 insan görüntüsü Şekil 5'te görüldüğü üzere parçalı nesne görüntüleri içermektedir ve nesnelere ağaçlar bitkiler veya diğer nesnelere ile perdelenmiştir. Şekil 6'da ise insan tespit edilmiş bitki, hayvan veya arka plana ait görüntüler gösterilmektedir.



Şekil 5. Tespit Edilemeyen 17 İnsan Görüntüsü

Tablo 1 incelendiğinde, IoU yöntemine 0.5 parametresi ile %94,6 gibi oldukça yüksek bir oranda tespit başarısı elde edildiği anlaşılmakta, IoU parametresi arttırıldıkça tespit başarısının hızla düşmekte olduğu gözlemlenmektedir. Tablo 2'deki hata matrisinden anlaşıldığı üzere yanlış pozitif değeri oldukça düşüktür.



Şekil 6. İnsan Olarak Tespit Edilmiş Hatalı Görüntüler

Tablo 1. IoU değerlerine göre tespit sonuçları

IoU	Tespit Edilen İnsan Sayısı	Başarı Oranı %
0,5	297	94,6
0,6	291	92,7
0,7	275	87,6

Tablo 2. Veri setinin 1. bölümüne ait hata matrisi

		Gerçek Değer		
		Nesne Var	Nesne Yok	Toplam
Tahmin Edilen Değer	Nesne Var	(TP) 297	(FP) 6	303
	Nesne Yok	(FN) 17	(TN) 1830	1847
	Toplam	314	1836	

Çalışmada sınırlayıcı kutu bilgileri bulunan 2150 görüntüde elde edilmiş anımsama (Recall) değeri 0.95 iken, kesinlik (Precision) değeri 0.98'dir.

Tarafımızca insan var veya yok şeklinde etiketlenen 730 görüntüde ise, Tablo 2'de görüldüğü üzere 219 insan içeren görüntünün 211'i doğru tespit edilmiş olup, 8 tanesi tespit edilememiştir. Tespit edilemeyen insan görüntülerinin gösterildiği Şekil 5 incelendiğinde, nesnelerin ağaç ve yapraklar ile büyük oranda kapandığından görülmediği ve nesnelerin parçalı olarak bulunduğu anlaşılmaktadır.

**Şekil 7.** Tespit Edilememiş 8 İnsan İçeren Görüntü**Tablo 3.** Veri setinin 2. bölümüne ait hata matrisi

		Gerçek Değer		
		Nesne Var	Nesne Yok	Toplam
Tahmin Edilen Değer	Nesne Var	(TP) 211	(FP) 0	211
	Nesne Yok	(FN) 8	(TN) 511	519
	Toplam	219	511	

730 görüntünün bulunduğu veri setinin 2. bölümünde ise anımsama değeri $211/219 = 0.96$ ve kesinlik değeri $211/211=1.00$ olarak elde edilmiştir.

Çalışmada elde edilen sonuçlara göre, kullanılan yöntemin görüntülerde bulunan çok sayıda nesneyi tespit etmede başarılı olduğu anlaşılmış olup, tespiti ait örnek görüntü Şekil 8'de gösterilmektedir. Buna ek olarak yöntemin 1.6 Ghz 8 çekirdek işlemciye sahip bir bilgisayarda görüntü başına ortalama nesne tespit süresi 75 milisaniye olup, gerçek zamanlı çalışmalarda kullanılmasının uygun olduğu anlaşılmaktadır.



Şekil 8. Birden Çok Nesne İçeren Görüntülerde Nesne Tespiti

4 SONUÇLAR

Çalışmamızda doğal ortamdan elde edilmiş ve karmaşıklık, ışık şiddeti yetersizliği, arka plan hareketi, kapanma, afin ve 3D dönüşümü, parçalı nesne bulunması, ölçekleme değişimi gibi zorlukları bulunan foto kapan veri setinde derin öğrenme mimarilerinden YOLO ile nesne tespiti gerçekleştirilmiştir. Gerçek zamanlı nesne tespitinin hızlı bir şekilde yapılabilirdiği bu mimarinin eğitim işlemine PASCAL VOC 2012 gibi büyük veri setlerinin kullanılması, nesne tespitinde yüksek başarı elde edilmesini sağladığı anlaşılmıştır. Çok sayıda havuzlama ve evrişim katmanı ile görüntülerdeki derin özniteliklerin çıkarılabildiği, birçok zorluğa karşı dayanıklılık elde edilebildiği ve nesne tespit performansında yüksek başarı elde edilebildiği anlaşılmaktadır. Sonraki çalışmalarda YOLO mimarisini kendi verilerimizle eğiterek gömülü sistemlerde gerçek zamanlı nesne tespiti yapmayı planlamaktayız.

Teşekkür

Bu çalışma Atatürk Üniversitesi Bilimsel Araştırma Projelerini destekleme programı kapsamında FYL-2018-6623 proje numarası ile desteklenmiştir.

Kaynakça

- [1] Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., & Lew, M. S. (2016). Deep learning for visual understanding: A review. *Neurocomputing*, 187, 27-48.
- [2] Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580-587).
- [3] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436.
- [4] Nguyen, H., Maclagan, S. J., Nguyen, T. D., Nguyen, T., Flemons, P., Andrews, K., ... & Phung, D. (2017, October). Animal Recognition and Identification with Deep Convolutional Neural Networks for Automated Wildlife Monitoring. In *Data Science and Advanced Analytics (DSAA), 2017 IEEE International Conference on* (pp. 40-49). IEEE.
- [5] Norouzzadeh, M. S., Nguyen, A., Kosmala, M., Swanson, A., Palmer, M., Packer, C., & Clune, J. Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning.
- [6] Gonçalves, D. N., de Arruda, M. D. S., da Silva, L. A., Araujo, R. F. S., Machado, B. B., & Gonçalves, W. N. Recognition of Pantanal Animal Species using Convolutional Neural Networks.
- [7] Tekeli, U., & Bastanlar, Y. Ham Fotokapan Verisinde Hayvanİçeren Fotoğrafların Belirlenmesi Detection of Images with Animals in Raw Camera-Trap Data.
- [8] Yu, X., Wang, J., Kays, R., Jansen, P. A., Wang, T., & Huang, T. (2013). Automated identification of animal species in camera trap images. *EURASIP Journal on Image and Video Processing*, 2013(1), 52.

- [9] Kays, R., Tilak, S., Kranstauber, B., Jansen, P. A., Carbone, C., Kays, R., Tilak, S., Kranstauber, B., Jansen, P. A., Carbone, C., Rowcliffe, M. J., & He, Z. (2010). Monitoring wild animal communities with arrays of motion sensitive camera traps. arXiv preprint arXiv:1009.5718.
- [10] Andavarapu, N., & Vatsavayi, V. K. (2017). Wild-Animal Recognition in Agriculture Farms Using W-COHOG for Agro-Security. *International Journal of Computational Intelligence Research*, 13(9), 2247-2257.
- [11] Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., ... & Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, 77, 354-377.
- [12] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097-1105).
- [13] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9).
- [14] Tompson, J., Goroshin, R., Jain, A., LeCun, Y., & Bregler, C. (2015). Efficient object localization using convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 648-656).
- [15] Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.