An Automated Mammals Detection Based on SSD-Mobile Net

Elham Mohammed Thabit A. Alsaadi and Nidhal K. El Abbadi

1 College of Information Technology, University of Babylon, Iraq
2 Computer Science Department, College of Science, University of Kerbala,

E-mail: elham.thabit@uokerbala.edu.iq

Abstract. Animal detection techniques are one of the researcher's interests and challenge. There are many difficulties faces by the researchers in this field that reduce the detection performance and efficiency, such as variation of image illumination, animal occlusion, the similarity of animal colors with background environment, etc.

Multi-label Image Detection and classification of Mammals animals is the goal of this paper which we proposed to achieve in this proposal by using Single Shot Multi-Box Detector (SSD) and MobileNet v1 coco_2017 model. Localizing and classifying multiple objects (animals) of the Mammal category in digital images is another goal. The suggested SSD is regarded as a more accurate, fast, and efficient way to detect objects of different sizes based on deep learning technology.

In this proposal, we used 2000 images in the network were collected from the standard dataset (such as Caltech 101) and the net. The SSD framework improves the detection and recognition processes of Convolution Neural Network (CNN). During the prediction time, the network introduces scores to the presence of every object class and bounded each object in the image with a box. Each box has a label that indicates the type of the object and the score represents the probability of the relationship of the object to that type. Boxes during the process are modified for getting the best matching to the object's shape. The experimental results of this work proved the efficiency of classifying and detecting animals even in the variation of illumination, pose, and occlusion. Detection and classification accuracy is up to 98.7 %. This suggestion is more reliable and accurate than other similar works and detects a wide range of Mammals animals, unlike other similar works.

Keywords: Deep Learning, Convolutional Neural Networks, Animals Classification, Mammals, Reptiles.

1. Introduction

Visual monitoring in animal scenes is currently one of the most distinguished research issues in the field of Computer Vision (CV), but the ways of detecting and understanding dynamic objects are still inaccessible [1], [2]. Animal detection methods are helpful to solve various problems such as prevent risky animal intrusion in a residential area [3]. It is also a part of a large work for a robot to detect and classify objects in critical situations such as natural disasters or other accidents which need to different objects such as human, animal and other [4]. Another example is self-driving cars, for which accurately detecting pedestrians, animals, street signs, or other vehicles, is a crucial feature when maximizing the safety of such systems. As well, it provides a great value in biomedical laboratories by monitoring laboratory animals using a powerful following method which can extract a rodent from a
frame under an uncontrolled environment [5]. Furthermore, it increases the efficiency and productivity of laboratory staff by reducing the time spent in direct monitoring of animals, also, to have a better understanding of animal behavior. Deep (CNN) or ConvNet has accomplished important success in the computer vision field, like target detection, target tracking, image classification, and semantic image segmentation [6]. It is a multi-layered neural network with a special architecture to detect complex features in data. Object detection means to predict the location of an object along with its category in static images which is one of the most critical computer vision problems [7], [8]. It often uses extracted characteristics and learning algorithms to identify the object belonging to a certain category of objects in a static image [9]. Several types of CNNs are existing; MobileNet is considered as one of the important applications for this type of CNN networks [10]. The architecture of “Depthwise Separable Convolutions” greatly decreases the number of parameters compared to the normal CNN that has the same depth. This results in lightweight neural networks. The MobileNet-SSD Model is a Single-Shot Multibox Detection (SSD) used as the pre-trained models for detecting and classification multi animals in difficult cases like detect part of an animal in lighting/illumination conditions [11], [8]. This combining between them gives us fast and sufficient performance to detect and classify the object. This architecture is more suitable for mobile and vision-based mobile applications where there is a lack of computing power [12]. SSD approach utilizes deep neural network technique to detect and classify the target objects and this limits the output area for bounding boxes in the default boxes set over different aspect ratios and scales per each spatial position of the feature map. At the prediction time, the network introduces scores to the presence of each object class in every default box. As well, it inserts modification to these boxes for getting the best matching of the object’s shape. Then, the current network collects predictions from multi-feature maps with various scales for handling objects of different volumes.

1.1. Research Problem
Animal detection and recognition is still a challenging task at the moment and there is no distinctive technique that provides a strong and efficient solution to all situations.
To summarize some of these challenges as follows:
1. The great variation in the appearance and the size of animals belonging to a particular category.
2. Lighting / Illumination Conditions: The image color of the picture is highly susceptible to light intensity and light direction variations.
3. Detect and classify animals are placed in arbitrary poses in the cluttered and the occluded environment, and in rotate states.

1.2. Paper Contributions
The contributions of this paper are presented in this section as follows:
1. Detecting the single and multi-animals of the Mammals category and also no animal category in case there is no mammal animal inside the image.
2. Detecting all types of the mammals’ category.
3. Detecting and classifying the animals when the variation of image light/illumination, in a blurry and dark image.

2. Related Work
This section briefly presents the existing related works on the classification and tracking of animals. Norouzzadeh, et al. (2018) trained Deep Convolutional Neural Networks (DCNNs) to recognize, amount, and describe behaviors of 48 species in 3.2 million images (Dataset Snapshot Serengeti). This neural network automatically recognized animals with 93.8% accuracy. Their results suggested that Deep Learning (DL) enables an inexpensive, large-scale, unobtrusive, and real-time gathering of wealth information around large numbers of wild animals. To test how best DL can automatic information elicitation from images of the camera trap, they collected millions of labeled data from the
Snapshot Serengeti (SS) dataset, the latest development of (DNN) engineering, and modern supercomputing [13].

Verma and Gupta (2018) focused on monitoring and analysis of wildlife through detection of the animal from natural scenes obtained by camera trap networks. They implemented a model of animal recognition using a self-learned feature of feature Deep CNN. Then this set of features was used for the process of classification using the state of art algorithms of Machine Learning (ML), which supported k-nearest neighbor, vector machine, and ensemble tree. Their results presented an accuracy of 91.4% [14].

(Schneider, et al. 2018) proposed network to train and compare two classifiers of deep learning object detection, Faster YOLO v2.0, and R-CNN to recognize, classify, and localize species of animals within images of camera-trap using datasets. The results were with average accuracies of 93.0% and 76.7% on the two datasets [15].

Xuefeng Liu et al. (2019) suggested a MobileNetV2-based embedded system combined with deep learning and learning transfer. This model aims to classify images of marine animals in an efficient way and in real-time. Firstly, these images were collected by using a robot equipped with a built-in device underwater. Subsequently, the proposed model has been created depending on CNN according to the images of “marine animals” which could guarantee the requirements in real-time. Then, for further enhance rankings, learning transfer was used. After that, the model was trained using selected images of marine animals. Lastly, this trained model has been applied in the embedded devices and categorized underwater images of marine animals in real-time [16].

Deboleena Roy, et al. (2020) proposed an adaptive hierarchical network architecture made up of DCNNs which can evolve and learn as new data becomes available. This network organized the incrementally accessible data into feature-oriented super-classes and enhanced it by applying self-growth functionality to existing hierarchical CNN models. Compared to fine-tuning a deep network, the proposed hierarchical model achieved an important reduction in training effort while retaining competitive accuracy on CIFAR-10, and CIFAR-100 [17].

3. SSD MobileNet Pre-Trained Model

MobileNet SSD model is one of the pre-trained models which includes integrating the SSD and MobileNet model [18]. SSD is a popular algorithm used as an object detector while Mobilenet is a convolutional neural network used as a features extractor to produce high-level features [19]. It is pre-trained on some public datasets like the Common Objects in Context (COCO) dataset for detecting and classifying multi-objects. The architecture of the SSD MobileNet model is lightweight in its architecture which is more suitable for mobile and embedded based vision applications where there is a lack of computing power. This architecture was proposed by Google. It uses depthwise separable convolutions which basically means it performs a single convolution on each colour (input) channel rather than combining all three and flattening it. The pointwise convolution then applies a 1x1 convolution to combine the outputs of the depthwise convolution. This factorization has the effect of drastically reducing computation and model size [20], [21].

This type of convolution divides the resulting output into two layers: a separate filtration layer, and a combination layer. The combination of these two layers will reduce the size of the model, thereby decreasing computational power requirements. A pre-trained model transfers the learned features or weights to begin the tuning process, and it allows the object detector to be trained with a limited amount of training. It is a very common and powerful method for the technique of deep learning of small image datasets which is to usage a pre-trained network [18].

4. SSD MobileNet Architecture

CNN in SSD is fully convolutional, based on MobileNet which serves as the backbone network [22]. Next, many additional CONV layers gradually decrease in size [23]. To detect smaller objects, the SSD uses extra shayaly layers with higher accuracy. For objects of various sizes, SSD detects multi metrics
by working on multiple convolution feature maps, each of which has the appropriate bounding boxes that predict dozens of categories and box offsets [24], [25]. In SSD, for the primacy of every object category inside each bounding box, a score is generated, followed by adjusting the bounding box to best match the shape of the object before detection [11]. SSD accomplishes its goal with the help of the multi-tasking loss function which represents the difference between the predicted and actual values [24]. It is created for getting the minimal of that function for optimizing the model and improving the accurate predictions. SSD is a feed-forward CNN, it associates each cell in the feature maps used for prediction with a set of the default bounding boxes (it is called also anchors) [24]. The algorithm predicts the offsets relative to a default box in the cell and a confidence score that expresses the presence of a target object class inside the default box [22]. For making the bounding box detectors particular, so each detector tries to predict just one object and various detectors will find various objects. SSD assigns each bounding box's detector to a particular position in the image. In this manner, detectors learn to specify objects at specific locations. MobileNet SSD model is shown in figure 1.

![MobileNet SSD Model](image)

*Figure 1: MobileNet SSD Model [26]*

5. **Research Methodology**

The multi-label image classification model is proposed for detecting and classifying multiple objects (animals) of the Mammal category in static images. SSD MobileNet v1 model is used in this work, figure 2 illustrates the block diagram of the proposed model.
5.1. Pre-processing Stages for Deep Learning
The most common parameters for input image data are the number of images, number of channels, image height, and image width. Several steps of images pre-processing should be implemented before using it in deep learning:

a. Standard aspect ratio image scaling
b. Normalize the input image to the range of (0, 1) and reduce the dimension of the image to meet the standard range value and dimension for the SSD network.
c. Image Augmentations Techniques are used in this work to solve some of the challenges facing the detection and classification of the animal in static images, for example, when the animal may appear truncated or partially occluded, the animals may change their shape, or problem of Lighting/Illumination Conditions.

5.2. Proposed Method of Multi-Label Image Classification
The proposed method of Multi-Label Image Classification is used for the following:

1. Classification + Localization: Classifying an image as a mammal category or no animal, also, localizing the target object inside the image to locate the main object in an image.

2. Object Detection: Detecting all types of mammal categories and drawing bounding boxes around them. Figure 3 shows the localization, classification, and detection of two samples of our results by this method.
In this research, the SSD MobileNet v1 model is used to detect and classify multiple (objects) animals of mammal’s class even in difficult cases like detecting part of the animal in lighting/illumination conditions. Both MobileNets and Single Shot Multibox Detection (SSD) are combined and pre-trained on the COCO_2017 dataset for detecting and classifying multiple objects in difficult cases. It is an efficient CNN architecture for mobile and applications of embedded vision [19]. The Common Objects in Context (COCO) dataset is the public dataset used widely to address the issue of large training datasets and to train models to depend on the image and to overcome of data limitations problem provided by Microsoft. Presently, this dataset performs like some of the best object detection data. It includes (300,000) segmented images with (80) various object categories with highly accurate location labels. This combination of SSD and MobileNet has achieved fast detection speed while maintaining high detection quality. The dataset used in this work includes 2000 colored images in different sizes like (128 x 128, 256 x 256, 512 x 512, 1024 x 1024) pixels of two categories (Mammal category and no animal category).

This model pre-processes the images by resizes the input image sizes into 300x300 pixels to meet the requirements of the SSD mobile net model. This new size of images is used for the process of detection. Detection performs with localizing and classifying multiple objects in still images. The outputs of the proposed model represent four matrixes which mapped to the indices 0 - 3, (Locations, Classes, Scores, and No. of detections) as shown in table 1. 20 epochs have been selected during the network training.

**Table 1. The model outputs**

| Index | Name            | Description                                                                                                                                 |
|-------|-----------------|---------------------------------------------------------------------------------------------------------------------------------------------|
| 0     | Locations       | They represent (floating-point values which fall in the range [0 – 1]), where object location represents the “top, left, bottom, right” of the bounding boxes at the model. |
| 1     | Classes         | A matrix of two integers is output as floating-point values [0 – 1], every index referring to the class label index for the labels file.         |
| 2     | Scores          | An array of two floating values which are between 0 and 1 represents the probability of detecting a class.                                  |
| 3     | No. of detections | A matrix of length one implicates the term “floating-point value” which is the entire No. of detection results for two objects detected. |
5.3. Algorithm of SSD Detection

The steps of the SSD algorithm which used in this work are as follow:

**Input:** RGB image.

**Output:** Represent four matrices that are mapped to the indices 0 - 3 (Locations, Classes, Scores, and No. of detections).

1. Apply CONV Layers to extract features at different sizes using various filters to get Feature Maps.
2. Get various feature maps at different sizes as output to fed SSD.
3. From each spatial location in Feature Maps, generate MultiClass Classification and Bounding Box Regression.
4. Add CONV layers to get smaller feature maps and continue the steps of detection.
5. Filter the bounding boxes by using IoU metrics, and the hard negative mining.
6. Compute the loss using a collection of classification (softmax), and detection (smooth L1).

5.4. Proposed Model Architecture

Our architecture of the Single Shot MultiBox Detector MobileNet model is based on (Depthwise Separable Convolutions) which is divided into two CONV layers, one layer for the filtering and the other for combining. The MobileNet model implements a single default filter for each neural input channel to initiate feature extraction. After the Depthwise Convolution, a (1 × 1) Pointwise Convolution follows to incorporate the Depthwise Convolution output. All these separable layers are followed by the batch norm, and ReLU nonlinearity expect the last (FC) layer that feeds into a softmax layer to classify as having no nonlinearity. Filters on MobileNet work for each color channel separately unlike the traditional CNN, and later combine the three results in one value. This factorization has the effect of drastically reducing computation and model size.

In this proposed model, (32, 64) filters with sizes (5 x5) are used to extract features from the input images followed by 2 max-pooling (pool size=2). Figure 4 illustrates the proposed model SSD MobileNet.

![Figure 4: Proposed model SSD MobileNet V1](image)

The model is extended by many convolutional layers. In SSD, the priority of creating a bounding box is to the target object with a high score. And then adjusting the bounding box according to the...
location, scale, and aspect ratios to get the best match to the shapes of the object (to the ground truth boxes).

During training time, each feature map is used for predicting bounding boxes and the variety in the size of the feature map allows object detection with various accuracy. Boxes are filtered by IoU metrics and Hard Negative Mining. IoU is a perfect metric to measure the overlap between the predicted box and the ground-truth as computed in equation (1) and as shown in figure 5. The ideal value between them has a 100% IoU, but particularly, the result over 50% is commonly considered a correct prediction. Finally, the predicted box will be determined with maximum overlap with ground truth.

\[
\text{IoU} = \frac{\text{area of overlap}}{\text{area of union}} \tag{1}
\]

![Figure 5: Intersection over union (IOU)](image)

Loss Function is used in this model which is the weighted sum of the loss of confidence “\(L_{\text{conf}}\)” and loss of localization “\(L_{\text{loc}}\)”, as shown in equation 2.

\[
L(x, c, l, g) = \frac{1}{N} (L_{\text{conf}} (x, c) + \alpha L_{\text{loc}} (x, l, g)) \tag{2}
\]

Where:
- \(L_{\text{conf}}\): loss of confidence
- \(L_{\text{loc}}\): loss of localization
- \(N\): number of matched default bounding boxes.
- \(\alpha\): the weight for the localization loss.
- \(l\): predicted box
- \(g\): ground truth box
- \(c\): class score
- \(X\): Input image

6. Experiment Results

The input images used in this model is RGB, and the dataset used consists of 2000 different images. 1000 images were selected for each category (mammals, and no animal). The proposed method was evaluated by the following tests:

6.1. Detection and Classification Single Animal from Different Types of Mammals

In this test, 1000 images of mammals included different types of mammals such as a dog, cat, deer, bear, rabbit, etc. have been selected. This proposed model has detected and classified all types of mammal categories in images and bounded them by a box successfully as shown in figure 6.
6.2. Detection and Classification Single Object of no animal Category
In this test, 1000 images of no animal category which include different types of objects belong to no animal category have been used. This model has detected and classified the objects of this category in images and bounded them by a box successfully as shown in figure 7.

6.3. Detection and Classification Multi animals of Mammals Category
In this test, the model has detected and classified the multi similar and different animals of mammals in one image successfully and bounded them by a box as shown in figure 8.
6.4. Detection and Classification Multi Objects of no animal Category

In this test, the model has detected and classified multi objects of no animal category and bounded them by a box successfully as shown in figure 9.

![Figure 9: Detection and classification multi objects of no animal category](image)

6.5. Detection and Classification Parts of Mammals in Image

In this test, the model has been detected and classified parts of animals displayed in the image successfully, figure 10 shows samples of the detected cut parts of animals.

![Figure 10: Detection and classification cut parts of animal](image)

6.6. Detection and Classification Mammals with Rotate Their Shapes

In this test, the proposed model has detected and classified the Mammals even in case of rotating their shapes in image and bounding them by box successfully as shown in figure 11.
6.7. Detection and Classification the Animals from Dark Images

In this test, the model has detected and classified the animals from dark images successfully and bounding them by a box successfully as shown in figure 12.

6.8. Detection and Classification of the Animals from Blurry Images

This model tested to detect and classify the animals from blurry images; the model successfully detected and classified the animals in the blurry image as shown in figure 13.
Figure 13: Detect and classify animals from blurry images

7. Comparison Between the Proposed Work Against the Others

Most studies are only detecting the specific types of a certain category but they are not detecting all types of animals that belong to a certain category. These studies also are not detecting a part of (object) animal in image, and are not detecting the objects in the dark images or under variation of illumination conditions. Most of them have not collected images; they used the online ImageNet dataset. Table 2 shows a comparison between the proposed work against the other works.

| No.  | Research Goal                                                                 | Method                                                                                     | Test Accuracy |
|------|-------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------|---------------|
| [27] | Detecting of large fauna in the open savannah of "Maasai Mara National Reserve, Kenya from very altitude resolution GeoEye-1 satellite" images. | By combining the features of "pixels based and object-based approaches of image classification". | 90-96%        |
| [28] | Identifying, and recognizing the animal images for Monitoring of Automated Wildlife” Bird, Rat, Bandicoot, Rabbit, Mouse, Cat”. | Using Deep CNNs.                                                                           | 94.2 %        |
| [29] | Recognizing and classifying animals located in "Slovak country" which is called "wolf, fox, brown bear, deer, and wild boar". | Using a collection of some approaches “SIFT - Scale-invariant feature transform, SURF - Speeded Up Robust Features”. | (50 , 80, 86)% |
| [13] | Detecting animals in images, identifying species, and counting them.          | Applying DNNs to automatically extract features from the images in the wild animal's dataset called "Snapshot Serengeti (SS) dataset". | 93.8%        |
### Table 1: Animal Tracking and Classification

| Reference | Task Description | Methodology | Accuracy |
|-----------|------------------|-------------|----------|
| [30]      | Detecting and monitoring cattle with a drone. | Using (CNNs) for recognizing the objects taken in the images. | 87%      |
| [6]       | Tracking a group of unmarked zebrafish. | Suggesting a "semi-automatic multi-organism tracking method" for dealing with partial blockages by using Kalman Filter to keep tracking of every animal. | 87.85%   |
| [7]       | Detecting and classifying “bear, hog, fox, deer, and wolf”, in the image. | Using CNN, this model is compared to high image recognition approaches like, “Principal Component Analysis, Local Binary Patterns Histograms, Linear Discriminant Analysis, and Support Vector Machine”. | 97%      |
| [33]      | Monitoring pets as dogs in smart cities utilizing animal biometrics. | Using the Learning Technique “One-Shot Similarity (OSS) via distance metric. | 96.87%   |
| [12]      | Developing a detector that analyses the images of video from camera traps in real-time. These images are related to "rhinoceros, humans and a set of six common large animals in the African savannah". | Using CNN “ SSD MobileNet V2”. | 90%      |
| **The proposed work** | Detecting, localizing, and classifying mammal in an image. Also, solve the problems of detecting animals in case of part of an animal in image, various illumination environment, and blurry image. | The proposed method used a “CNNs training, combine with pre-trained Single Shot Detector (SSD) MobileNet v1 architecture”. | 98.7% |

### 8. Conclusions

In this work, the SSD and MobileNet-v1 are proved as an accurate and fast method for detection, classification, and localization the animal in static images. Most other studies are only detecting the specific types of a certain category but they are not detecting all types of animals that belong to the Mammals category as we did in the current proposal. Several tests have been done in the SSD MobileNet v1 model, and the model has achieved good results in the detection and classification of the object (animal). The results were very effective, the accuracy of detection using 20 epochs was up 98.7% which is promising compared with the other works. This model successfully detected and classified the single and multiple objects in the static images. Moreover, this model successfully detected and classified the animal in different illumination such as dark and blurry images. Also, it could detect and classify the animal when there is a part of the animal in the image. Each object in the image was detected and bounded by boxes, besides labeling it.

### References

[1] Z. He *et al.*, “Visual informatics tools for supporting large-scale collaborative wildlife monitoring with citizen scientists,” *IEEE Circuits Syst. Mag.*, vol. 16, no. 1, pp. 73–86, 2016.
[2] A. J. Shepley, G. Falzon, P. Meek, and P. Kwan, “Location Invariant Animal Recognition
Using Mixed Source Datasets and Deep Learning,” bioRxiv, 2020.

[3] C. Zhu, T. H. Li, and G. Li, “Towards automatic wild animal detection in low-quality camera-trap images using two-channeled perceiving residual pyramid networks,” in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2860–2864.

[4] H. Dong, G. Sun, W.-C. Pang, E. Asadi, D. K. Prasad, and I.-M. Chen, “Fast ellipse detection via gradient information for robotic manipulation of cylindrical objects,” IEEE Robot. Autom. Lett., vol. 3, no. 4, pp. 2754–2761, 2018.

[5] A. C. Coelho and J. García Diez, “Biological risks and laboratory-acquired infections: a reality that cannot be ignored in health biotechnology,” Front. Bioeng. Biotechnol., vol. 3, p. 56, 2015.

[6] W. Wang, Y. Li, T. Zou, X. Wang, J. You, and Y. Luo, “A Novel Image Classification Approach via Dense-MobileNet Models,” Mob. Inf. Syst., vol. 2020, 2020.

[7] G. Rogez, P. Weinzaepfel, and C. Schmid, “Lcr-net: Localization-classification-regression for human pose,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 3433–3441.

[8] S. Tang and Y. Yuan, “Object detection based on convolutional neural network,” in International Conference–IEEE–2016, 2015.

[9] E. M. T. A. ALSAADI, “Auto Animal Detection and Classification among (Fish, Reptiles and Amphibians Categories) Using Deep Learning.”

[10] A. G. Howard and M. Zhu, “Mobilenets: Open-source models for efficient on-device vision,” Google AI blog, 2017.

[11] L. Leal-Taixé and S. Roth, Computer Vision–ECCV 2018 Workshops: Munich, Germany, September 8-14, 2018, Proceedings, Part VI, vol. 11134. Springer, 2019.

[12] A. Tydén and S. Olsson, “Edge Machine Learning for Animal Detection, Classification, and Tracking.” 2020.

[13] M. S. Norouzzadeh et al., “Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning,” Proc. Natl. Acad. Sci., vol. 115, no. 25, pp. E5716–E5725, 2018.

[14] A. Joly et al., “Lifeclef 2017 lab overview: multimedia species identification challenges,” in International Conference of the Cross-Language Evaluation Forum for European Languages, 2017, pp. 255–274.

[15] P. Badre, S. Bandiwadekar, P. Chandanshive, A. Chaudhari, and M. S. Jadhav, “Automatically Identifying Animals Using Deep Learning,” Int. J. Recent Innov. Trends Comput. Commun., vol. 6, no. 4, pp. 194–197, 2018.

[16] X. Liu, Z. Jia, X. Hou, M. Fu, L. Ma, and Q. Sun, “Real-time Marine Animal Images Classification by Embedded System Based on Mobilenet and Transfer Learning,” in OCEANS 2019-Marseille, 2019, pp. 1–5.

[17] D. Roy, P. Panda, and K. Roy, “Tree-CNN: a hierarchical deep convolutional neural network for incremental learning,” Neural Networks, vol. 121, pp. 148–160, 2020.

[18] A. G. Howard et al., “Efficient convolutional neural networks for mobile vision applications,” arXiv Prepr. ArXiv1704.0486, 2017.

[19] Z.-Q. Zhao, P. Zheng, S. Xu, and X. Wu, “Object detection with deep learning: A review,” IEEE Trans. neural networks Learn. Syst., vol. 30, no. 11, pp. 3212–3232, 2019.

[20] E. Suharto, A. P. Widodo, and E. A. Sarwoko, “The use of mobilenet v1 for identifying various types of freshwater fish,” in Journal of Physics: Conference Series, 2020, vol. 1524, no. 1, p. 12105.

[21] S. GHOURY, C. SUNGUR, and A. DURDU, “Real-Time Diseases Detection of Grape and Grape Leaves using Faster R-CNN and SSD MobileNet Architectures.”

[22] R. Verma and C. Arora, “Modeling and implementation of real-time animal detection module for mobility assistant for visually impaired (MAVI) system.” 2017.

[23] W. Liu et al., “Ssd: Single shot multibox detector,” in European conference on computer vision, 2016, pp. 21–37.
[24] J. L. Masache Narvaez, “Adaptation of a Deep Learning Algorithm for Traffic Sign Detection,” 2019.

[25] A. Younis, L. Shixin, S. Jn, and Z. Hai, “Real-Time Object Detection Using Pre-Trained Deep Learning Models MobileNet-SSD,” in Proceedings of 2020 the 6th International Conference on Computing and Data Engineering, 2020, pp. 44–48.

[26] R. Pandey, M. White, P. Pidlypenskyi, X. Wang, and C. Kaeser-Chen, “Real-time Egocentric Gesture Recognition on Mobile Head-Mounted Displays,” arXiv Prepr. arXiv1712.04961, 2017.

[27] Z. Yang, T. Wang, A. K. Skidmore, J. de Leeuw, M. Y. Said, and J. Freer, “Spotting east African mammals in open savannah from space,” PLoS One, vol. 9, no. 12, p. e115989, 2014.

[28] H. Nguyen et al., “Animal recognition and identification with deep convolutional neural networks for automated wildlife monitoring,” in 2017 IEEE international conference on data science and advanced analytics (DSAA), 2017, pp. 40–49.

[29] S. Matuska, R. Hudec, P. Kamencay, M. Benco, and M. Zachariasova, “Classification of wild animals based on SVM and local descriptors,” AASRI Procedia, vol. 9, pp. 25–30, 2014.

[30] A. Rivas, P. Chamoso, A. González-Briones, and J. Corchado, “Detection of cattle using drones and convolutional neural networks,” Sensors, vol. 18, no. 7, p. 2048, 2018.

[31] G. de Oliveira Feijó, V. A. Sangalli, I. N. L. da Silva, and M. S. Pinho, “An algorithm to track laboratory zebrafish shoals,” Comput. Biol. Med., vol. 96, pp. 79–90, 2018.

[32] T. Trnovszký, P. Kamencay, R. Orješek, M. Benčo, and P. Šýkora, “Animal recognition system based on convolutional neural network,” 2017.

[33] S. Kumar and S. K. Singh, “Monitoring of pet animal in smart cities using animal biometrics,” Futur. Gener. Comput. Syst., vol. 83, pp. 553–563, 2018.