Grenze International Journal of Engineering and Technology, June Issue

Automated Animal Detection and Classification using Machine Learning Approach

Balakrishna K[1], Niveditha H R[2], Dhanushree V[3], Sandesh N G[4] and Paramesha R[5]
[1,4]Dept of ECE, Maharaja Institute of Technology Mysore, Karnataka, India
Email: balakrishnak_ece@mitmysore.in
[2]Dept of ECE, PES College of Engineering Mandya, Karnataka, India
[3]Dept of ECE, SJBIT Bangalore, Karnataka, India
[5]Dept of ECE, Polytechnic College, Mirle, Karnataka, India

***Abstract*— In this paper, machine learning approach algorithm models such as the Region-based Convolutional Neural Network (R-CNN) classification model, Single Shot multi-box Detection (SSD), and You Only Look Once (YOLO) regression model are used for the detection and classification of animals. For experimentation purposes, 2000 datasets of four classes of animals (elephant, cheetah, cow, and horse) were collected using remote surveillance cameras. By eliminating bounding boxes and gradually reducing convolution filters, SSD and YOLO improve computer vision tasks in speed, accuracy, and efficacy for relatively low-resolution images compared to R-CNN. The proposed classification algorithm model R-CNN gives a mean average precision (mAP) of 95.22%, whereas the regression algorithm models SSD and YOLO give mAPs of 98.32% and 98.89%, respectively.**

***Index Terms*— Anchor Box; Class Prediction; Machine Learning; Neural Network; Object Detection; Region-based Convolutional Neural Network (R-CNN); Single Shot multibox Detection (SSD); You Only Look Once (YOLO).**

I. INTRODUCTION

In recent years, the use of remote surveillance cameras in wildlife research and management has created many opportunities for researchers in the field. The purpose of surveillance cameras may vary widely depending on applications, but detecting animals in the frames is a common need for all applications. The surge of interest in surveillance cameras is rapidly increasing due to challenges associated with recognizing Region of Interest (ROI) from many input datasets for animal identification. Nowadays, animal detection plays a significant role in the application of animal identification, animal counting, and avoiding animal-human conflicts near forest areas [1]. Thus, there is ample interest in developing an automated detection model. In a conventional method, humans are used to identify the object and spot the class through visual details in photographs or videos. The agenda is to train the computer to do something similar to humans by enforcing knowledge of what an image contains, called computer vision [2]. Computer vision technology allows one to identify and locate objects in an input image or video by mimicking human visual capabilities to teach computers to interpret the visual world, which is defined as object detection in Machine learning (ML) and object recognition in Deep learning (DL) [3].
Machine learning is a branch of Artificial Intelligence (AI) and a subset of computer vision that focuses on imitating human behavior using data and improving its performance gradually [4].

Grenze ID: 01.GIJET.12.2.295_21

This technology can detect and locate objects with multiple blocks of convolution and pooling layers to extract features such as textures, shapes, edges, etc. An object detection framework generally has working stages such as the selecting ROI through the anchor box and bounding box, extracting features using Convolutional Neural Network (CNN), classifying each proposal, and finally, object localization using bounding box regression [5]. Several models have been tried in the application of object detection and classification, such as Region-based Convolutional Neural Network (R-CNN), Faster R-CNN, Mask R-CNN, Single Shot multi-box Detection (SSD) and You Only Look Once (YOLO) [6,7].

This paper proposes a new approach to detect and classify the animals appearing in the surveillance camera based on the machine learning approach. For an experiment, purposes considered four classes of wildlife and domestic animals, such as elephants, cheetahs, cows, and horses, for the classification and identification in real-time of images collected from remote surveillance cameras. Which are those animals that majorly conflict with human beings in their farm fields by grazing their crops and even harming human life? Even this application can be adopted for counting wildlife animals during an animal census from a remote place. Therefore, we need a relatively efficient and reliable model for the detection and classification of animals from a remote location using surveillance cameras.

II. RELATED WORKS

Verma G K and Gupta P. [8] worked on monitoring and detecting wildlife animals through a camera-trapped network using a deep convolutional neural network. Here, the authors collected datasets of 20 categories of species, where each species had around 100 images collected, including daytime and nighttime formats, resulting in a 24x7 monitor. The experimentation was carried out for a pre-trained ImageNet ILSVRC (ImageNet Large Scale Visual Recognition Challenge) classification model with a ten-fold cross-validation approach. Here, nine folds are considered for training the model, and the remaining one fold is for testing purposes out of the feature datasets available. The machine learning classifiers such as Support Vector Machine (SVM), K-Nearest Neighbor (KNN), and Ensemble classifiers are applied and have an accuracy range of 89.5 to 91.4%.

Tabak M A et al. [9] worked on classifying the five species of animals using a machine learning algorithm through motion-activated camera images. The authors manually considered five locations across the United States and one location from Canada to capture datasets, and the United States datasets were used for training the model through the ResNet-18 architecture using a tensor framework. Here, we randomly selected 90% of the datasets to train the model, and 10% were used to test the model's performance for each species. The developed model performed with an 85%–97.5% accuracy range, considering the highest confidence score level. Even a classification model can classify around 2000 images per minute for a 16-gigabyte RAM system.

Balakrishna K et al. [10] worked on identifying intruder animals to protect the cropland using the Internet of Things (IoT) and machine learning. For experimentation purposes, the authors considered five classes of animals with 60 images of each species for 300 images in the datasets. The developed algorithm to detect and classify the animals was done using machine learning algorithms such as R-CNN and SSD. The SSD performed better than the R-CNN algorithm, considering the computation speed and mean Average Precision (mAP) accuracy of 89.32% compared to 85.22%. Here, real-time application for detection and classification suggests SSD suits best comparatively to the R-CNN approach.

Norouzzadeh M S et al. [11] worked on identifying, classifying, and counting wild animals using deep learning algorithms through camera-trapped images. We considered around 3.8 million image snapshot datasets for experimentation to identify, classify, and count 48 behavior species (moving, resting, and grazing). Using a motion-sensor-based camera here leads to the inexpensive, unobtrusive, and real-time collection of datasets. A deep learning algorithm trained using a convolutional neural network (ResNet-152) to detect, classify, and count the animal species with an accuracy of around 93.5% from the best confidence score level.

Chandrakar R et al. [12] worked on detecting and recognizing animals using a deep convolutional neural network through genetic segmentation. The authors developed a convolutional neural network algorithm with three layers; here, a genetic algorithm is used to detect animals with high accuracy for the segmentation process. The proposed algorithm is compared with standard detection algorithms and existing methodologies, which have higher error rates due to high false-positive and negative rate detection. For training and testing purposes, create a database for two classes with ten pictures in each class, and the overall efficiency shows that it varies from 98.7% to 88.1%.

Several state-of-the-art detection, recognition, and classification models were developed through the Convolutional Neural Network algorithm to classify the animals. But still, infancy in achieving efficiency and

performance for the detection and classification of animals through camera trap images and real-time detection and classification has not received much consideration so far.

III. PROPOSED METHODOLOGY

In the proposed methodology, we trained the model to detect and classify the animals according to their classes based on the intelligence of the machine learning approach, as shown in Figure 1. The proposed model consists of two steps: a training step and a testing step to detect and classify the objects in the collected datasets from remote surveillance cameras.

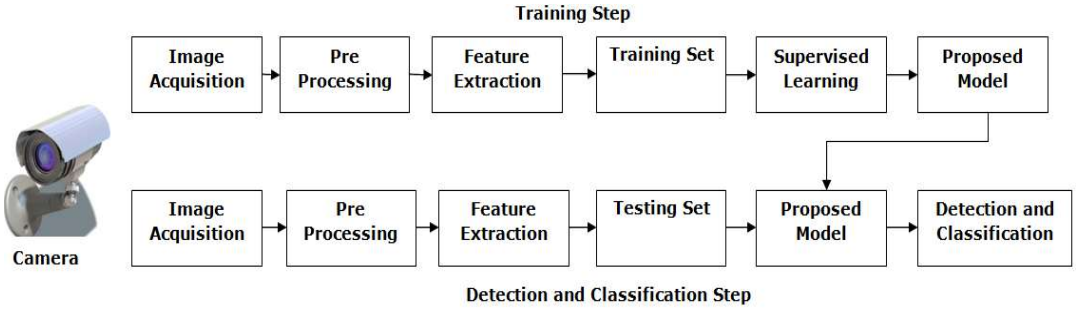

Figure 1: Flow diagram of the proposed model.

Object Detection

In computer vision technology, classifying the object category based on the image of interest is a challenging task, defined as object detection or recognition. The object detection algorithm recognizes the object of interest from the input image through the Bounding Box or Anchor Box [13]. The anchor box looks for an object of interest in the large region of the input image and determines it. Once it determines objects from the area, it predicts the ground truth bounding boxes more precisely, adjusting the regions boundaries [14]. The input image height is taken as '*h*', width is taken as '*w*', scale is taken as '$S \in (0,1)$' and aspect ratio as '$r > 0$', then the anchor box width and height are taken as $ws\sqrt{r}$ and $hs/\sqrt{r}$. Generate multiple anchor boxes with varying shapes, scale ($S_1, S_2, S_3, \ldots\ldots, S_n$) and aspect ratio ($r_1, r_2, r_3, \ldots\ldots, r_m$). The complexity of computation is very high to cover all ground truth bounding boxes. So containing a combination of S_1 and r_1 is considered as given in Equation 1:

$$(S_1, r_1), (S_1, r_2), \ldots\ldots, (S_1, r_m), (S_2, r_1), (S_2, r_2), \ldots\ldots, (S_n, r_m) \quad (1)$$

Total '$wh(n + m - 1)$' anchor boxes will be generated for the entire input image, having '$(n + m - 1)$' anchor boxes centered on the same pixel. When the ground truth bounding box is known, and the anchor box surrounds the object of the image, the similarity is measured using the Jaccard index [15]. If two boxes are considered sets like '*A*' and '*B*', then the Jaccard index is taken as the ratio of the size of their intersection to the size of their union, as shown in Equation 2.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} \quad (2)$$

A similarity check for two bounding boxes is done by the Jaccard index, known as Intersection over Union (IoU), defined as the ratio of the intersection area to the union area [16]. The range of IoU varies from '0' to '1', where '0' refers to no overlap of bounding boxes and '1' refers to bounding boxes are overlap and equal.

In training the datasets, each anchor box is considered to detect the object having 'class' and 'offset' labels of the ground truth bounding box relative to the anchor box. Anchor boxes considered has $A_1, A_2, A_3, \ldots\ldots\ldots\ldots, A_{n_a}$ and Ground truth bounding boxes considered has $B_1, B_2, B_3, \ldots\ldots\ldots\ldots, B_{n_b}$ for a given image (where n_a refers multiple anchor boxes and n_b refers multiple ground truth boxes), here $n_a \geq n_b$ for the matrix taken $X \in R^{n_a \times n_b}$. The IoU of anchor box A_i and ground truth bounding box B_j of i^{th} row and j^{th} column for matrix element x_{ij} taken to find the largest elemental value to denote its row and column indices [17, 18]. This step is carried out until all the elements in n_b columns in the matrix are discarded, which assigns the anchor box to the bounding box according to the threshold. Based on this, each anchor box is labeled for class and offset. The class will be labeled as same as the ground truth bounding box 'B' for anchor box 'A', but the offset will be labeled considering the relative position between the central coordinates of two boxes as given in Equation 3. The offset for each anchor box is labeled using equation 3. The central coordinates are considered as (x_a, y_a) and (x_b, y_b), widths as w_a and w_b and their heights as h_a and h_b.

$$\left(\frac{\frac{x_b - x_a}{w_a} - \mu_x}{\sigma_x}, \frac{\frac{y_b - y_a}{h_a} - \mu_y}{\sigma_y}, \frac{\log\frac{w_b}{w_a} - \mu_w}{\sigma_w}, \frac{\log\frac{h_b}{h_a} - \mu_h}{\sigma_h}\right) \quad (3)$$

For transformation, default values are considered such as $\mu_x = \mu_y = \mu_w = \mu_h = 0$, $\sigma_x = \sigma_y = 0.1$ and $\sigma_w = \sigma_h = 0.2$ (Default values are considered based on the Homography matrix having entries between 0 to 1).
The anchor box predicts the class in an image during prediction, generating multiple anchor boxes based on the expected offset value. When there is a significant overlapping of anchor boxes surrounding the same object, we must merge the similar anchor boxes surrounding the same object using the non-maximum suppression (NMS) method [19]. The NMS method predicts the confidence score for the anchor box, where the lower confidence predicted score value will be removed, and the highest confidence scored value will be considered for labeling the class.

A. R-CNN Model

In computer vision and image processing, R-CNN is a classification algorithm designed to detect the object in the input image by defining its boundaries. It operates with a selective search approach to extract the ROI, which will be captured in rectangular boundaries [20]. The selective search uses sliding filters of different sizes on the input image to extract the object; here, the computation effort will increase as the number of sliding filters increases. The segmentation for the input image is done by providing different colors to the input image to separate the object after applying the exhaustive search approach, which uses a greedy algorithm to check the similarity between the regions as given in equation 4. The extracted ROI goes through the CNN model after resizing the image to a fixed capacity, such as $227 \times 227 \times 3$ images. The resized image will be processed through CNN to extract the object of size 4096. Finally, for the segregation of the classes, a Support Vector Machine (SVM) classifier is used, as shown in Figure 2 [21].

$$S(a,b) = Stexture(a,b) + Ssize(a,b) \quad (4)$$

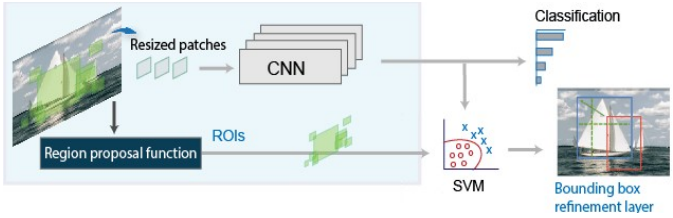

Figure 2: Object detection using R-CNN detector architecture

B. SSD Model

SSD is a regression algorithm model that consists of two parts: one is to extract feature maps, and another is to detect the object in the input image using convolution filters [22]. SSD uses a Visual Geometry Group with 16 layers (VGG16) to extract the feature maps connected with Convolutional layers for high-quality image classification. Here, the image is divided using the grid concept (i.e., 3X3 or 4X4), and each grid cell is responsible for identifying the ROI, which makes four object predictions regardless of the depth of the feature maps for each cell. Where it generates multiple boundary boxes and a relevant confidence score called a multi-box, out of various predictions, the highest predicted score bounding box will be selected, and the remaining boxes [23]. The SSD layers are added to the backbone of the model, and the predictions are interpreted at the spatial location of the final layers, as shown in Figure 3.

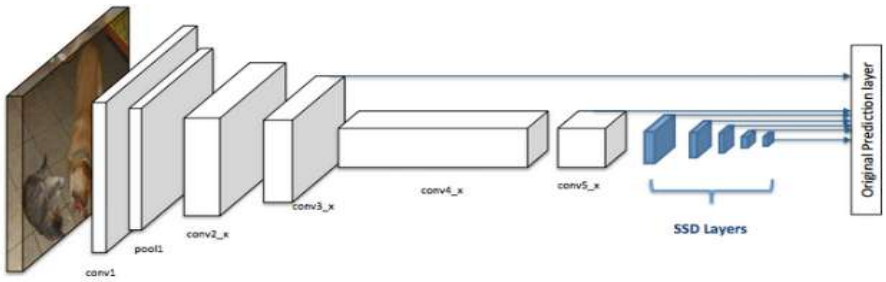

Figure 3: Object detection using SSD detector architecture.

C. YOLO Model

YOLO is a regression algorithm model that works with three techniques: residual blocks, bounding box regression, and intersection over union to detect and recognize an object in an input image [24]. In the residual blocks technique, an input image is divided equally into N×N dimensional grids, and each grid cell will look for an object that appears within it. In the bounding box regression technique, images include attributes like width, height, bounding box center, and a class of objects represented in a single bounding box [25]. The intersection over union technique detects the objects in each grid cell prediction box to give confidence scores. It compares the predicted bounding box with the actual box of the object to check for similarity between them. The combination of all three techniques is applied to detect the output results by using a unique bounding box that fits the object perfectly.

IV. DISCUSSION

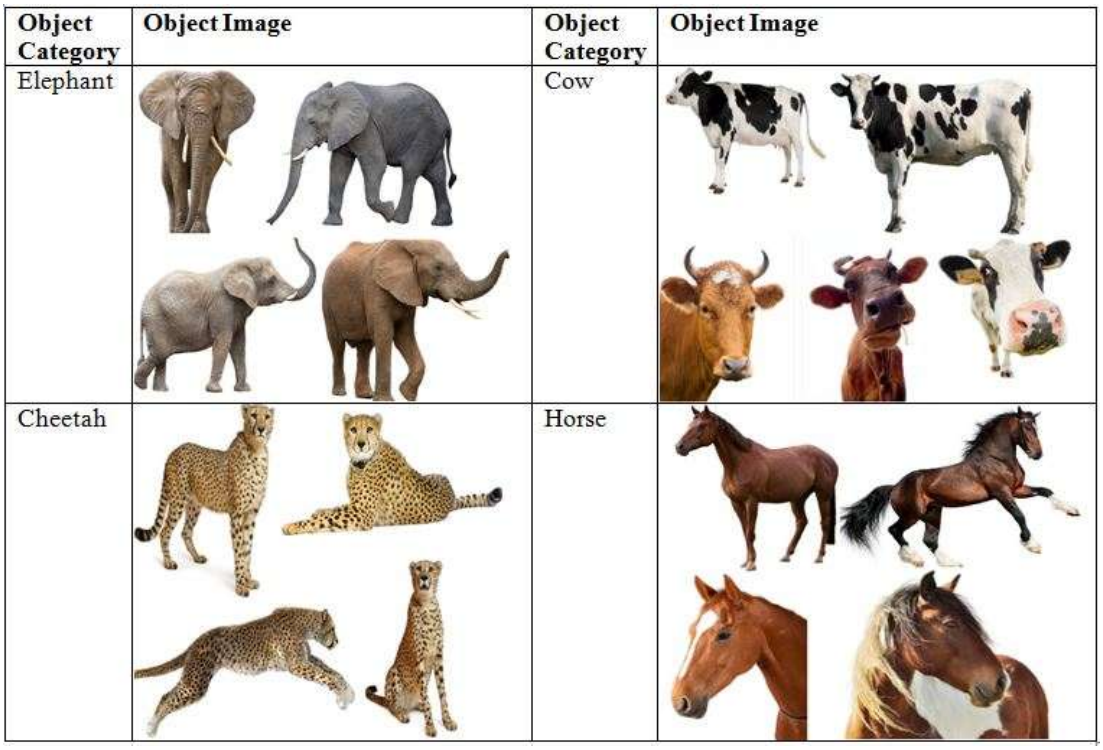

Figure 4: Dataset samples of animals

For detection and classification purposes, we collected around 500 images of each animal using remote surveillance cameras, including elephants, cheetahs, cows, and horses, from 2000 datasets used as training and testing images. While capturing the images, we randomly collected datasets under various conditions such as different lighting, backgrounds, blurring, animal movement, shading, etc. The reference sample of collected images is shown in Figure 4. As mentioned in the proposed methodology for object detection and classification, R-CNN, SSD, and YOLO models are applied.

In R-CNN, CNN is pre-trained with ResNet on the ImageNet dataset to classify animals into four classes in Tensorflow software [26, 27]. The selective search approach searches for ROI that contains the target object independently and warps that ROI with varying sizes to have a fixed length as required by CNN. While fine-tuning the CNN, it gives additional classes referred to as 'no object' or background, whereas the warped proposal region now has '4+1' classes with a smaller learning rate. The binary SVM has a feature vector generated through CNN optimizing one forward propagation to detect and classify the samples for each class of animals. Proposed regions with threshold ≥0.3 are taken as positive samples and others are taken as negative samples. Also, to avoid localization errors, the model is trained to predict the correct bounding box offset using CNN vector features [28].

In SSD, the input image is divided into 4×4 grid and each grid cell is detected for objects present in the region of the image [29]. To predict the object, the SSD is trained to match the appropriate bounding box with the anchor box of the input image, as discussed in the proposed section. In implementing the training part, each anchor box aspect ratio is taken as 1.0∶1.0 and zoom level is taken 1.0 to have square boxes to detect the objects starting from values 0.2 to 0.9 and their locations. A total of 5 classes are considered, including the 'no object' or background as one class, a class is predicted by considering the 4 offset values for each of the 8732 default boxes and its dimensions. The confidence threshold of 0.01 is applied to each box, the Jaccard Index value is set to 0.45 and NMS repeats the process until the boxes are formed.

In YOLO, the input image is divided into 3×3 grid and image classification and localization are applied to each grid [30]. To define the classification model, we have considered nine parameters, such as the object being present, four bounding box values specifying the object coordinates, midpoint and dimension and four classes of animals. If the object is not present in the grid, it is taken as 'zero' and the remaining parameters will be taken as undefined. If the object is present in the grid and is taken as 'one', then the model takes the midpoint of the object and specifies relevant values for the remaining parameters. For 9 grids, we will have a nine-dimensional output vector having 3×3 ×9. The IoU and NMS decide whether the predicted box is correct or not by taking the actual box and the predicted box, where the IoU ≤0.5 will be discarded and NMS is applied by combining the two bounding boxes to obtain a single prediction of the object.

V. Results

The experimentation was conducted by training the classification (R-CNN) and regression (SSD, YOLO) algorithm models for detecting and classifying four classes of animals, elephant, cheetah, cow, and horse, for available datasets. The R-CNN extracts features from the input image to perform forward propagation using CNN to predict class from the bounding box region. Whereas, the computation time required is around 40–50 seconds for analysis, which may not suit best for real-time applications [31]. SSD uses default boxes with varying shapes, scales, and aspect ratios, as shown in Table 1. The prediction score will be best for 8732 boxes, considering the better coverage of scale, aspect ratio, and location. The majority of the prediction box doesn't contain the object, so a prediction score of less than 0.01 will be discarded.

TABLE I: SSD Prediction Layers and mAP

Prediction source layers						mAP*	Number of Boxes
38x38	19x19	10x10	5x5	3x3	1x1		
✓	✓	✓	✓	✓	✓	98.32	8732
✓	✓	✓				92.51	9864
	✓					81.57	8664

YOLO takes an entire image in a single instance to predict the object coordinates and probabilities of class with a processing speed of 45 frames per second. The b_x, b_y, b_h and b_w are taken as bounding box coordinates, height and width of the grid, where Table 2 shows the sample dimensional vector for no object present in the grid and class 2 object present in the grid.

TABLE II: YOLO Dimensional Vector

Y								
P_C	b_X	b_Y	b_h	b_w	C_1	C_2	C_3	C_4
0	?	?	?	?	?	?	?	?
1	0.4	0.3	0.9	0.6	0	1	0	0

SSD and YOLO improve the computer vision tasks in terms of speed, accuracy, and efficacy for relatively low-resolution images, eliminating bounding boxes and progressively minimizing convolution filters for our designed models. Figure 5 shows the testing model execution for the proposed model, which is detecting and classifying the animals accurately.

The proposed classification algorithm model R-CNN gives a mean average precision (mAP) of 95.22%. In contrast, the regression algorithm models SSD and YOLO give mAPs of 98.32% and 98.89%, respectively, for 2000 datasets of animal images for 4 classes, as shown in Table 3. However, when we look into the execution speed of the proposed model, R-CNN, it takes around 49 seconds to generate the test results. The main advantage of regression algorithm models like SSD and YOLO is that the training phase and testing result

computation will be high-speed. Compared with the previous works carried out by researchers in the field, Balakrishna K et al., [10] and Verma G K et al., [8] as shown in Table 3, our proposed algorithm model shows better accuracy and efficacy. The YOLO and SSD-based detection algorithms developed work better for the real-time classification of animals for selected datasets in our research.

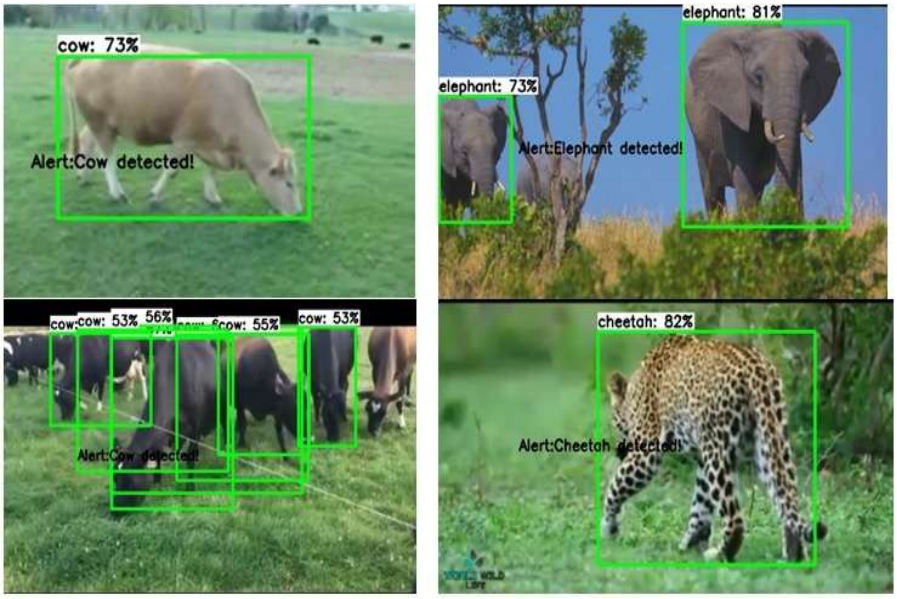

Figure 5: Testing model execution for the proposed algorithm

TABLE III: ALGORITHM PERFORMANCE COMPARISON WITH STATE-OF-THE-ART

State-of-the-Art	mAP* (%)
Proposed R-CNN	95.22
Proposed SSD	98.32
Proposed YOLO	98.89
Balakrishna K et al., [10]	89.32
Verma G K et al., [8]	87.45
*mAP (mean Average Precision)	

VI. CONCLUSION

The automated detection and classification of objects in a region of interest have been continuously upgraded with computational equipment in computer vision technology, which is dependent on machine learning and deep learning methodologies. A classification algorithm model and a regression algorithm model are used to detect the animals and classify them using a machine-learning approach. The classification algorithm model, such as R-CNN, and the regression algorithm model, such as SSD and YOLO, are applied considering the bounding box, anchor box, region of interest, selective search approach, sliding window, segmentation, convolutional neural network, feature maps, intersection over union, non-maximum suppression, etc. The proposed results show that the SSD and YOLO algorithm models work better considering the performance and computation time required for execution. The R-CNN takes around 40–50 seconds, whereas SSD and YOLO perform instantly and execute 45 frames per second. In future work, the communication protocol can be applied to the existing model to transfer information to remote locations through a cellular network, Global System for Mobile (GSM) communication, General Packet Radio Services (GPRS), etc.

ACKNOWLEDGEMENT

I would like to show my gratitude to the Maharaja Institute of Technology Mysore and thank the teaching and non-teaching staff of Dept of ECE. Also, thanks to my parents and friends who all are directly or indirectly supported this research.

A. Funding Statement

The authors did not receive support from any organization for the submitted work.

B. Competing Interests

The authors have no conflicts of interest to declare that are relevant to the content of this article

C. Data availability

Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

REFERENCES

[1] Vidhyalatha, T., Sreeram, Y., & Purushotham, E. (2022). Animal Intrusion Detection using Deep Learning and Transfer Learning Approaches. International Journal of Human Computations & Intelligence, 1(4), 51-60.

[2] Zhou, Q., Li, X., He, L., Yang, Y., Cheng, G., Tong, Y., ... & Tao, D. (2022). TransVOD: End-to-end video object detection with spatial-temporal transformers. IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(6), 7853-7869.

[3] Sharifani, K., & Amini, M. (2023). Machine learning and deep learning: A review of methods and applications. World Information Technology and Engineering Journal, 10(07), 3897-3904.

[4] Xu, Y., Cugurullo, F., Zhang, H., Gaio, A., & Zhang, W. (2025). The emergence of artificial intelligence in anticipatory urban governance: Multi-scalar evidence of China's transition to city brains. Journal of Urban Technology, 32(3), 9-33.

[5] Zhang, H., & Zhang, S. (2024). Focaler-iou: More focused intersection over union loss. arXiv preprint arXiv:2401.10525.

[6] Balakrishna, K., and N. Rajesh. "Design of Remote Monitored Solar Powered Grasscutter Robot with Obstacle Avoidance using IoT." Global Transitions Proceedings (2022).

[7] K. Balakrishna, B. R. Ganesh Prasad, N. D. Dhanyashree, V. Balaji and N. M. Krishna, "IoT based Class Attendance Monitoring System using RFID and GSM," 2021 IEEE International Conference on Mobile Networks and Wireless Communications (ICMNWC), 2021, pp. 1-5, doi: 10.1109/ICMNWC52512.2021.9688482.

[8] Verma, G. K., & Gupta, P. (2018). Wild animal detection using deep convolutional neural network. In Proceedings of 2nd international conference on computer vision & image processing (pp. 327-338). Springer, Singapore.

[9] Tabak, M. A., Norouzzadeh, M. S., Wolfson, D. W., Sweeney, S. J., VerCauteren, K. C., Snow, N. P., ... & Miller, R. S. (2019). Machine learning to classify animal species in camera trap images: Applications in ecology. Methods in Ecology and Evolution, 10(4), 585-590.

[10] Balakrishna, K., Mohammed, F., Ullas, C. R., Hema, C. M., & Sonakshi, S. K. (2021). Application of IOT and machine learning in crop protection against animal intrusion. Global Transitions Proceedings, 2(2), 169-174.

[11] Norouzzadeh, M. S., Nguyen, A., Kosmala, M., Swanson, A., Palmer, M. S., Packer, C., & Clune, J. (2018). Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning. Proceedings of the National Academy of Sciences, 115(25), E5716-E5725.

[12] Chandrakar, R., Raja, R., & Miri, R. (2021). Animal detection based on deep convolutional neural networks with genetic segmentation. Multimedia Tools and Applications, 1-14.

[13] Zhao, B., & Song, R. (2024). Enhancing two-stage object detection models via data-driven anchor box optimization in UAV-based maritime SAR. Scientific Reports, 14(1), 4765.

[14] Balakrishna K.,, & Rao, M. (2019). Tomato Plant Leaves Disease Classification Using KNN and PNN. International Journal of Computer Vision and Image Processing (IJCVIP), 9(1), 51-63. http://doi.org/10.4018/IJCVIP.2019010104.

[15] Al-Najdawi, N. A., Al-Shawabkeh, A. F., Tedmori, S., Ikhries, I. I., & Dorgham, O. (2025). Comprehensive evaluation of optimization algorithms for medical image segmentation. Scientific Reports, 15(1), 37190.

[16] Cho, Y. J. (2024). Weighted Intersection over Union (wIoU) for evaluating image segmentation. Pattern Recognition Letters, 185, 101-107.

[17] Balakrishna K., Sandesh N.G. (2021) Design of Dynamic Induction Charging Vehicle for Glimpse of Future: Cutting Down the Need for High-Capacity Batteries and Charging Stations. In: Kalya S., Kulkarni M., Shivaprakasha K.S. (eds) Advances in VLSI, Signal Processing, Power Electronics, IoT, Communication and Embedded Systems. Lecture Notes in Electrical Engineering, vol 752. Springer, Singapore. https://doi.org/10.1007/978-981-16-0443-0_16.

[18] Ren, Q., Wang, Y., & Xu, J. (2025). A Segmentation Network for Identification and Segmentation of Overlapped and Blurred GPR Signatures. Measurement Science and Technology.

[19] Vijayendrakumar, D., & Kempegowda, B. (2025). A hybrid framework for wild animal classification using fine-tuned DenseNet121 and machine learning classifiers. IAES International Journal of Artificial Intelligence, 14(3), 2083. https://doi.org/10.11591/ijai.v14.i3.pp2083-2095.

[20] Rudreshgowda, N. H., Kempegowda, B., & Sammilan, A. (2025b). A hybrid intelligent model for prediction of coronary artery diseases using TabNet and multiclass SVM. Indonesian Journal of Electrical Engineering and Computer Science, 40(1), 156. https://doi.org/10.11591/ijeecs.v40.i1.pp156-163.

[21] Saheed, Y. K., Arowolo, M. O., & Tosho, A. U. (2022). An efficient hybridization of k-means and genetic algorithm based on support vector machine for cyber intrusion detection system. International Journal on Electrical Engineering and Informatics, 14(2), 426-442.
[22] Arwidiyarti, D. (2025). Single Shot Multibox Detector (SSD) in Object Detection: A Review. IJACI: International Journal of Advanced Computing and Informatics, 1(2), 118-127.
[23] Arwidiyarti, D. (2025). Single Shot Multibox Detector (SSD) in Object Detection: A Review. IJACI: International Journal of Advanced Computing and Informatics, 1(2), 118-127.
[24] Lee, J., & Hwang, K. I. (2022). YOLO with adaptive frame control for real-time object detection applications. Multimedia tools and applications, 81(25), 36375-36396.
[25] Bao, S., Zhong, X., Zhu, R., Zhang, X., Li, Z., & Li, M. (2019). Single shot anchor refinement network for oriented object detection in optical remote sensing imagery. Ieee Access, 7, 87150-87161.
[26] Balakrishna K.,. (2021). WSN, APSim, and Communication Model-Based Irrigation Optimization for Horticulture Crops in Real Time. In P. Tomar, & G. Kaur (Eds.), Artificial Intelligence and IoT-Based Technologies for Sustainable Farming and Smart Agriculture (pp. 243-254). IGI Global. http://doi:10.4018/978-1-7998-1722-2.ch015.
[27] Balakrishna K.,. (2020). WSN-Based Information Dissemination for Optimizing Irrigation Through Prescriptive Farming. International Journal of Agricultural and Environmental Information Systems (IJAEIS), 11(4), 41-54. http://doi.org/10.4018/IJAEIS.2020100103.
[28] Raut, V., Gunjan, R., Shete, V. V., & Eknath, U. D. (2023). Gastrointestinal tract disease segmentation and classification in wireless capsule endoscopy using intelligent deep learning model. Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization, 11(3), 606-622.
[29] Abdusalomov, A., Umirzakova, S., Tashev, K., Egamberdiev, N., Belalova, G., Meliboev, A., ... & Cho, Y. I. (2025). AI-Driven UAV Surveillance for Agricultural Fire Safety. Fire, 8(4), 142.
[30] Agarwal, V., & Bansal, G. (2024). Automatic number plate detection and recognition using YOLO world. Computers and Electrical Engineering, 120, 109646..
[31] Olorunshola, O., Jemitola, P., & Ademuwagun, A. (2023). Comparative study of some deep learning object detection algorithms: R-CNN, fast R-CNN, faster R-CNN, SSD, and YOLO. Nile J. Eng. Appl. Sci, 1(1), 70-80.