Real-Time Lion Detection In Forest Near Village Area Using YOLO

Rinkal Jain
Department of Computer Science and Engineering
School of Technology, Pandit Deendayal Energy University.
Gandhinagar, Gujarat.
0000-0003-3282-6870

Parth Kher
Department of Computer Science and Engineering
School of Technology, Pandit Deendayal Energy University.
Gandhinagar, Gujarat.
0009-0005-6841-6330

Chintan Bhatt
Department of Computer Science and Engineering
School of Technology, Pandit Deendayal Energy University.
Gandhinagar, Gujarat.
0000-0002-0423-0159

2025 International Conference on Artificial Intelligence and Machine Vision (AIMV) | 979-8-3315-2697-9/25/$31.00 ©2025 IEEE | DOI: 10.1109/AIMV66517.2025.11203611

***Abstract*—The YOLO (You Only Look Once) algorithm is utilised in this research study to propose a real-time lion identification system for woods close to village areas. With this approach, the possibility of dangerous animals endangering the inhabitants is lessened. This research explains how to utilise APIs to combine the YOLO algorithm with a messaging system so that villagers may receive real-time warnings when they come across potential threats. The data collection technique involves gathering pictures and recordings of lions that are dangerous to people. The YOLO algorithm may be trained to correctly identify lions in photos by labelling them with the annotated data that has been gathered. When implemented in woods close to villages, the YOLO algorithm can identify dangerous animals in real-time through training and optimisation. The algorithm received precision score of 0.78 and 0.73 for YOLO v5 and v8 respectively. The integration of the messaging system guarantees that villages receive notifications promptly, enabling them to take appropriate safety measures. The system implementation entails configuring the communication system and YOLO algorithm inside the forest, as well as routine testing and observation to ensure everything is operating as it should. By offering an effective and proactive method for lion detection in the area, this research will support both human safety and wildlife conservation.**

***Index Terms*—Lion Detection, Animal Detection, Wildlife Monitoring, Image Processing, Computer Vision, Deep Learning, YOLO Algorithm, Convolution Neural Networks (CNN), Object Recognition, Real-time Detection, Wildlife Conservation, Safety Precautions, Human-Wildlife Conflict.**

I. INTRODUCTION

The coexistence of humans and wildlife in forest regions presents one of the most challenging and complex problems worldwide. Particularly in India, the highly endangered Asiatic lion finds its habitat within the Gir National Park and Wildlife Sanctuary [1]. Despite its conservation significance, the proximity of this habitat to neighbouring communities significantly elevates the risk of lion-human conflicts. Such encounters pose severe threats, potentially resulting in injuries or fatalities for both humans and lions. In addressing this critical issue, the utilisation of technology, particularly computer vision algorithms, emerges as a pivotal approach to enhancing the safety of both people and wildlife [2].

One such computer vision technology utilised for lion detection in Gir National Park is You Only Look Once (YOLO) [3]. YOLO aims to reduce the likelihood of conflict and improve the safety of both humans and lions. In this study, we propose the installation of a lion detection system in the Gir National Park region, leveraging YOLO [4]. Data collection methods include drone monitoring and camera traps, which facilitate the annotation of lion boundaries and other pertinent details. Through training the YOLO algorithm on annotated images, real-time identification of lions becomes achievable [5].

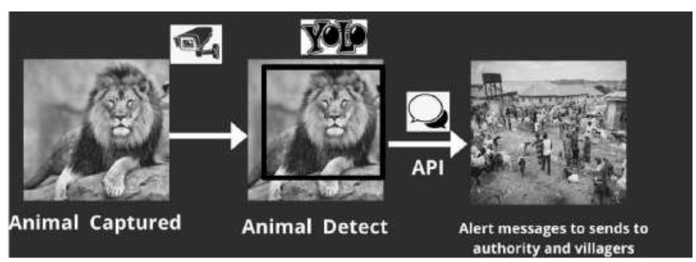


Fig. 1: Flow Diagram

Upon successful training of the lion detection system, its implementation enables continuous monitoring of lion activities in the area. Instances of lion sightings trigger notifications to nearby villagers through a connected messaging system integrated with the lion detection algorithm [6]. This comprehensive approach not only mitigates the risk of human-lion conflicts but also aids in safeguarding the endangered Asiatic lion species.

This research paper presents a comprehensive overview of the proposed lion detection system, covering aspects such as data collection, YOLO algorithm training, deployment in the field, integration of the messaging system, and strategies for reducing human-lion conflicts [7]. The findings and results of our research not only contribute to the safety of nearby villagers but also serve as a valuable reference for addressing conflicts between humans and wildlife globally [8]. This study showcases how technology can be harnessed to protect communities residing near wildlife habitats by leveraging computer vision algorithms and messaging systems.

The evolution of the YOLO algorithm, standing for "You Only Look Once," has been notable since its inception. Initially introduced in a paper by Redmon et al. in 2016 [9], subsequent iterations such as YOLOv2 [10], YOLOv3, and the latest YOLOv8 [11] [6] have brought significant advancements in object detection capabilities. YOLOv8, building upon its predecessors, incorporates deep learning algorithms, particularly convolutional neural networks (CNNs), to enhance object detection precision and performance [12] [13] [3] [14].

This research endeavour seeks to harness the power of YOLO [15] technology to address the pressing issue of human-lion conflicts in forest regions, exemplifying the potential of advanced computer vision algorithms in wildlife conservation and human safety.

II. Methodology

A. *Custom Dataset:*

This research project aimed to acquire notable knowledge and comprehension of lion identification using the structured gathering, evaluation, and examination of information from several sources. Following the data gathering procedure, 500 high-definition (HD) photographs with a minimum resolution of 640x640 pixels were produced as a unique dataset. The overall size of this dataset is 277 megabytes (MB). Images were submitted by a wide range of sources, including individual and professional photographers, viral videos, and local sources. Rich and varied lion photos from several sources are included in this collection to provide a complete picture of lion habitats and behaviours. An effective strategy for lion conservation and a reduction in human-wildlife conflicts were made possible by the selected dataset, which provided the basis for further research and testing. The main aim is to provide important insights for improving lion detection techniques and encouraging human-wildlife cooperation [16] [17] in forest settings by utilising this unique knowledge.

To ensure the robustness and generalizability of the detection model across real-world forest surveillance scenarios, a diverse and comprehensive dataset was curated, encompassing a wide spectrum of environmental and situational variations. The dataset comprises high-resolution images and annotated video frames of lions captured under heterogeneous lighting conditions—including natural daylight (morning and evening), low-light dusk scenarios, and nocturnal settings with infrared (IR) illumination. This temporal diversity equips the model with the ability to detect lions accurately across the full diurnal cycle. Furthermore, the imagery spans multiple ecological terrains such as dense tropical forests, arid grasslands, rocky outcrops, and semi-open scrublands, thereby enhancing the model's adaptability to differing background complexities and visual noise. In addition, the dataset encapsulates various lion postures and behavioral states, including static positions like sitting and lying, as well as dynamic movements such as walking and running. Such intra-class variability is critical for training a model that is resilient to pose distortions, partial occlusions, and motion blur. Collectively, this curated dataset forms the backbone of the system's training architecture, ensuring that the deep learning model develops a nuanced understanding of lion morphology and movement patterns across diverse operational contexts. This deliberate inclusion of real-world variability serves to reduce model bias and significantly elevates performance in unpredictable and dynamic forest environments.

Fig. 2: Dataset at a glance

B. *Image annotations:*

Fig. 3: Lion Annotated Images

For the image annotation stage, we utilised the publicly accessible CV image annotation tool Makesense.ai [18] to annotate a dataset of 500 lion photographs for lion recognition. Human annotators were asked to annotate the photographs by manually drawing bounding boxes around the lion's head structure, body shape, and tail features [19]. Quality control processes were used, and experienced annotators carefully examined and resolved any ambiguities in the photographs to ensure the accuracy of the annotations. The meticulous annotation process improved the dataset with real-world data, establishing its credibility as a reliable source for building a robust lion detection model with real-world applications.

Because of the accuracy of this approach, an exact system that can accurately recognise lions in real-world scenarios has been created [3].

C. Annotation Process and Quality Assurance:

A robust and secure annotation workflow was employed to label lion images with high precision and consistency, leveraging both Label Studio and Makesense.ai as primary annotation tools. Label Studio was installed and configured in an offline local environment, thereby ensuring that all annotation activities were conducted without any external data transmission—eliminating risks associated with data leakage or privacy breaches. This local deployment provided full control over the annotation pipeline and was particularly well-suited for handling sensitive ecological surveillance data. In addition, Makesense.ai was selectively used for its user-friendly browser-based interface. Importantly, Makesense.ai performs annotations locally within the browser, without uploading images to the cloud, maintaining a high level of data confidentiality.

The annotation task involved drawing bounding boxes around lions in a wide range of poses and scenes, capturing behavior such as resting, walking, and running in diverse landscapes including forests, rocky terrain, and open grasslands. The annotation process was carried out by a team of three skilled annotators, all trained in visual object tagging for wildlife datasets. To ensure uniformity in labeling practices, all team members followed a standardized annotation guideline, which defined clear protocols for object boundaries, visibility thresholds, and multi-object scenes.

To maintain quality and reduce subjectivity, annotations underwent a consensus review process, where discrepancies were collaboratively resolved and revalidated. This peer-verification step helped enforce labeling consistency and reduced intra-annotator variation. Inter-annotator agreement was monitored to identify and correct any deviations from the established standards. Special attention was given to maintaining a balanced distribution of annotated instances across different classes and conditions to improve the dataset's representativeness and training efficiency. Through this structured and privacy-conscious annotation strategy, the dataset achieved high reliability, laying a strong foundation for accurate model training and deployment in real-world forest surveillance systems.

D. Model Training and Computational Setup:

In the interest of developing a highly accurate and generalizable detection model, the dataset was systematically divided into 80 percent for training and 20 percent for testing, adhering to standard best practices in supervised learning. Prior to training, all images were rescaled to 640×640 pixels to align with the input constraints of the YOLOv5 architecture. Comprehensive preprocessing and augmentation techniques were applied to increase the variability and robustness of the training data. These included random rotations, horizontal and vertical flips, contrast enhancement, and brightness adjustments, all of which simulate diverse environmental conditions and improve the model's resilience to noise and distortion.

The training process was carried out using the PyTorch framework, chosen for its flexibility, scalability, and ease of deployment in research environments. The experiments were executed on Google Colab Pro+, utilizing a cloud-based NVIDIA Tesla T4 GPU (16 GB VRAM) coupled with 25 GB of system RAM and a Ubuntu 20.04 Linux backend. This setup provided a cost-effective yet powerful computing infrastructure, well-suited for deep learning workloads. The cloud environment also allowed for seamless integration with Google Drive for dataset management and ensured persistent storage of training logs and model checkpoints. The combination of systematic data handling, rigorous preprocessing, and efficient training infrastructure contributed significantly to the performance and stability of the final detection model.

E. System Integration:

To facilitate real-time communication and ensure prompt mitigation of human-wildlife conflict, particularly in the case of lion intrusions, this study integrates a robust telecommunication framework into the detection pipeline. Upon the successful identification of a lion through the YOLO-based object detection model, the system triggers an automated alert mechanism powered by the Twilio Programmable Messaging API. This telecommunication API enables instantaneous dissemination of SMS notifications to a predefined set of stakeholders—including forest rangers, wildlife officials, and residents in high-risk zones. The alert message encapsulates critical information such as the timestamp, location identifier (e.g., camera ID or geotag), and an urgent warning advisory. These notifications are programmatically dispatched via secure HTTPS requests using Twilio's RESTful API, ensuring minimal latency—typically within a span of 2 to 5 seconds. The integration supports dynamic message generation and bulk messaging capabilities, enhancing scalability across geographically dispersed populations. Additionally, the system incorporates a throttling mechanism to prevent alert spamming from consecutive detections and supports future extensibility to alternative communication channels such as WhatsApp, voice calls, and email notifications. This real-time telecommunication layer transforms the passive surveillance system into a proactive early warning network, significantly amplifying situational awareness and enabling swift human response. The implementation underscores the pivotal role of intelligent communication infrastructures in augmenting AI-driven wildlife monitoring systems and promoting coexistence through technological intervention.

III. YOLOv5 Model:

YOLOv5, or "You Only Look Once," is an object identification technique that was first described by Joseph Redmon and Ali Farhadi in 2016. The most recent version of the Yolo architectural series was released in 2020 [14]. The YOLO algorithm uses several technology components to function. For example, a deep neural network (DNN) and feature pyramid networks are used to extract features of different sizes. Consequently, the model can capture more accurate data in real time. Additionally, it uses anchor boxes, or pre-established reference boxes, to improve localisation accuracy [20]. Moreover, it uses advanced techniques, including data augmentation, focus loss, and transfer learning, to improve its accuracy and generalisation ability. YOLO5 introduces innovations in model architecture, optimisation strategies, and training methodologies [21]. For example, it uses bounding box regression to do class prediction in a single step. When it was released in 2020, it was one of the first YOLO products to offer noticeable improvements to accuracy, speed, and flexibility for a wide range of object recognition applications [12].

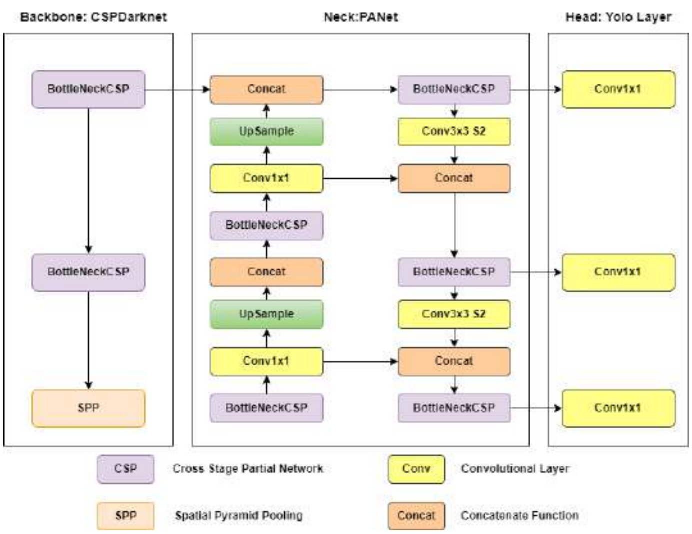


Fig. 4: Architecture Of YOLOV5 Model

IV. YOLOv5 Architecture

The backbone network processes the fine-grained visual data. The detection network then refines the image data that the neck network has analysed. The detection network produces the final detection output [22]. The design is displayed in Figure 1, along with the BottleneckCSP Bottleneck module with convolution layers and the Spatial Pyramid Pooling (SPP) module. The building images are from the paper "Improving YOLOv5 with Attention Mechanism for Detecting Boulders from Planetary Images" [17] [13]. The YOLOv5 object detection model is based on a single-stage neural network. It looks at the entire image to predict bounding boxes and probabilities of classes for each object. The structure of the model is in three phases.

A. Backbone

The CSPDarknet backbone network's ResNet iteration is used to construct YOLOv5. This variation varies in that it performs object-detecting tasks with more efficiency and accuracy. A series of layers, known as residual blocks when grouped together and as individual layers as convolution layers, make up the core of the CSPDarknet. The residual blocks help to increase the accuracy and efficiency of the network, while these convolution layers collect characteristics from the input picture at various scales [22].

B. Neck

In the neck network, information collected by numerous backbone network nodes is combined to create a set of object detection feature maps. The neck of YOLOv5 functions as a feature pyramid network, which enables the model to identify objects of various sizes. Numerous high-level top-down and low-level bottom-up route nodes make up the PANet neck. More precisely, bottom-level route nodes are transmitted from the backbone to the upper levels of the neck network, while top-level pathway nodes are relocated from the backbone network to the lower levels. These pairings of top-level and bottom-level routes could allow the same network to generate many feature maps containing details about different-sized objects [22].

C. Head

For every object in an image, the head network is in charge of the prediction of the bounding box and class probabilities. YOLOv5 makes use of the head network's effectiveness and simplicity, which has shown outstanding results in real-world circumstances. Multiple convolution layers together with fully connected layers make up the YOLOv3 head. The fully connected layers forecast the bounding box for each object in the picture, while the convolution layers extract features from the feature map generated by the neck network [22].

V. Training the YOLOv5 Model

The model is trained using supervised learning on annotated object photographs, which make up the training dataset for YOLOv5. The model learns by decreasing the loss between predicted bounding boxes, class probabilities, and ground truth labels [16].

The loss function utilised in the YOLOv5 model's training consists of three terms:

- **Bounding box loss:** The difference between the bounding boxes that are expected and the ground reality.
- **Classification loss:** The difference in class probabilities between the predictions and the ground truth is expressed by this statement.
- **Confidence loss:** When the model predicts bounding boxes for items that aren't in the picture, it gets penalised.

VI. YOLOv8 Model

YOLOv8 is a one-step object identification model that can identify items in an image in a single pass [3]. It is far faster due to this feature than two-stage algorithms like R-CNN, which require several image passes. YOLOv8 adds various improvements to the EfficientDet architecture, which is considered the most advanced object detection model.

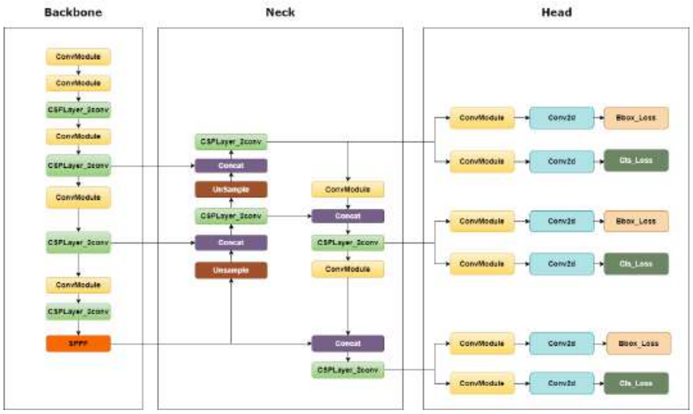


Fig. 5: Architecture Of YOLOV8 Model

- A new backbone network called Cross Stage Partial Connections (CSPDarknet) is outperforming its predecessor in terms of accuracy and efficiency.
- Introducing the Adaptive Anchor Box Design, a novel anchor box design that can accommodate a variety of object shapes and sizes.
- The improved accuracy of the model is a result of the adoption of an altered loss function known as IOU loss.

A. *YOLOv8 Stages*

YOLOv8 splits the object detection job into three steps:

1) **Feature extraction:** In the first stage of YOLOv8, features are extracted from the input image via the CSPDarknet backbone network.
2) **Prediction:** In the second stage, YOLOv8 predicts the bounding boxes and class probabilities for every object in the image. This method makes use of the adaptive anchor box architecture and the IOU loss function.
3) **Post-processing:** The last stage of YOLOv8 post-processes the second stage predictions to eliminate any overlapping or redundant bounding boxes [17].

B. *YOLOv8 Benefits*

YOLOv8 offers the following benefits over previous YOLO iterations and other object recognition techniques:

- Improved accuracy: YOLOv8 achieves state-of-the-art accuracy on several object identification benchmarks, including Pascal VOC and COCO.
- Faster performance: YOLOv8 performs substantially faster than previous iterations of the algorithm and alternative object recognition methods like R-CNN.
- Enhanced efficiency: Compared to its predecessors and other object detection models, YOLOv8 uses less memory and processing power [17].

C. *YOLOv8 Advantages over YOLOv5*

YOLOv8 offers several advantages over YOLOv5 [23], including:

Model	Label	Max Prediction (mAP)
YOLOv5	Single Lion	0.8
YOLOv8	Single Lion	0.9
YOLOv5	Multiple	0.7
YOLOv8	Multiple	0.8

TABLE I: Model Comparison with Max Prediction (mAP)

Fig. 6: YOLOV5 Batch Label & Predicted Images

- Improved accuracy: YOLOv8 outperforms YOLOv5 by a considerable percentage point, achieving state-of-the-art accuracy on many object identification criteria, including COCO and Pascal VOC.
- Faster speed: YOLOv8 performs significantly quicker than YOLOv5 when working with smaller input images.
- More efficient: YOLOv8 performs as well as YOLOv5 but requires less memory and computing power.
- New features: YOLOv8 has a number of new features that improve the accuracy and efficiency of the model, such as the IoU loss function and the Adaptive Anchor Box Design [3].
  Because of its many improvements over earlier YOLO iterations and other object identification models, YOLOv8 is recognised as a state-of-the-art object detection model [24]. Modern technologies are included in it to improve performance, and it works really well in terms of efficiency, speed, and precision [19].

VII. Experimental Result

YOLO v8 is the latest and greatest object detection model. Compared to YOLOv5 and other predecessor models, YOLOv8 offers several significant improvements in terms of accuracy, speed, and efficiency. It also introduces a number of new features that enhance YOLO's performance. In Figure 6, we compared the speed and performance of each model on

Fig. 7: YOLOV8 Batch Label & Predicted Images

a sample of images. Here, we compare two images side by side. Table 1 illustrates the comparison between Figure 6 and Figure 7.

- Learning rate: 0.001
- Weight decay: 0.0005
- Momentum: 0.9

A. Confusion Matrix Metrics

In the context of YOLO and object detection :

1) True Positives (TP): Identifiable and properly characterised objects are noted.

2) True Negatives (TN): When an object is absent, it is appropriately labelled as such.

3) False Positives (FP): The term describes the potential for false alarms to arise when an object is mistakenly identified when none is there.

4) False Negatives (FN): These are items that are visible in the picture but are not able to be identified or found.

These metrics give key performance indicators by evaluating the model's object detection and recognition capabilities in images. Examining characteristics like decision boundaries, accuracy scores, and others helps researchers gain additional insight into the inner workings of the model. These comprehensions are necessary in order to accurately compute the F1 score, accuracy, and recall, all of which are important in assessing the general effectiveness and reliability of the model [25] [26].

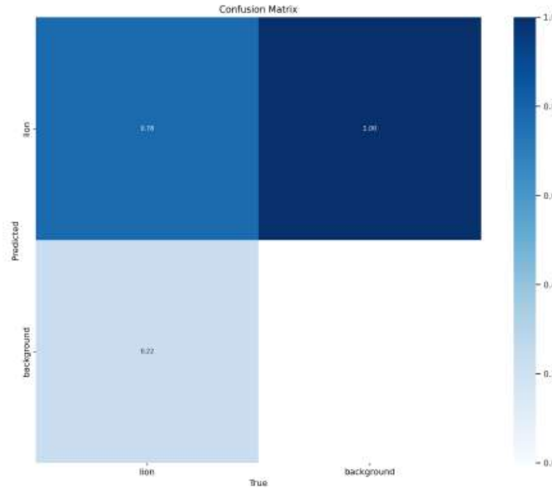
Model	Object	Precision Score	Decision
YOLOv5	Lion	0.78	Good Prediction
YOLOv8	Lion	0.73	Good Prediction

TABLE II: Model Comparison with Precision Score and Decision

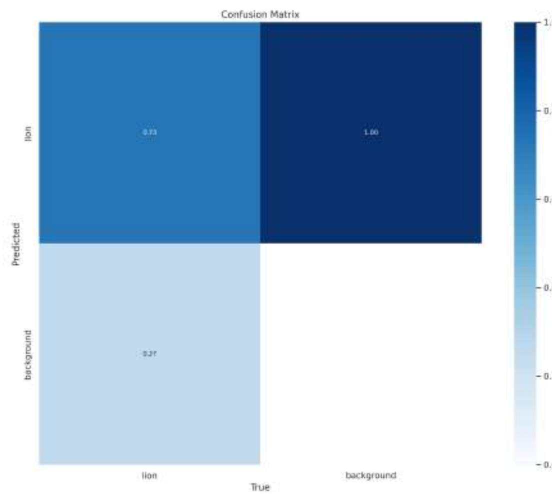
VIII. Results and Discussion:

Model performance and the number of training epochs have a positive association, according to the study. YOLOv5 and YOLOv8 both produced exceptional accuracy outcomes after 50 epochs of intensive testing and training. Therefore, it was amazing that even more significant enhancements to model performance could be achieved with extended training durations. While there was concern about overfitting and the potential for resource consumption, breaking beyond the 50-epoch barrier in YOLO models indicated the potential for improved detection accuracy. The findings emphasised how essential it is to modify the number of training epochs as a practical strategy to tailor YOLO models to specific application needs.

These results highlight the need for a comprehensive approach to model creation and have important implications. The paper emphasises the importance of extended training periods and their impact on model performance, showing the dynamic nature of optimisation strategies in object recognition. This in-depth understanding opens up new avenues for research

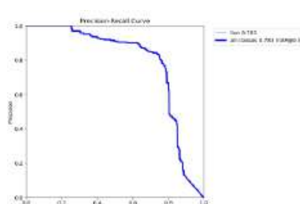


(a) Confusion matrix: YOLOv5 Model

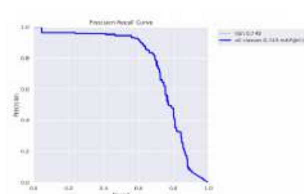


(b) Confusion matrix: YOLOv8 Model

Fig. 8: Confusion matrix comparing performance of YOLOv5 and YOLOv8 in detecting multiple lions



(a) Precision & Recall curve: YOLOv5 Model



(b) Precision & Recall curve: YOLOv8 Model

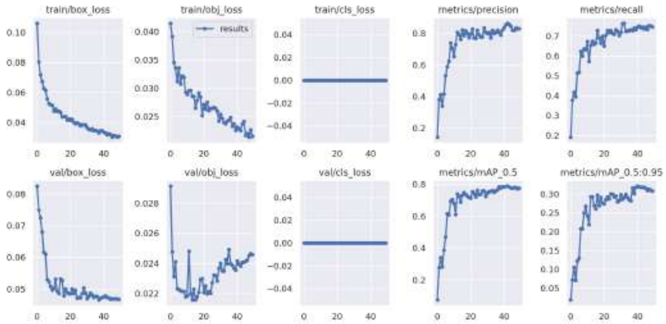


Fig. 10: Result Curves: YOLOV5

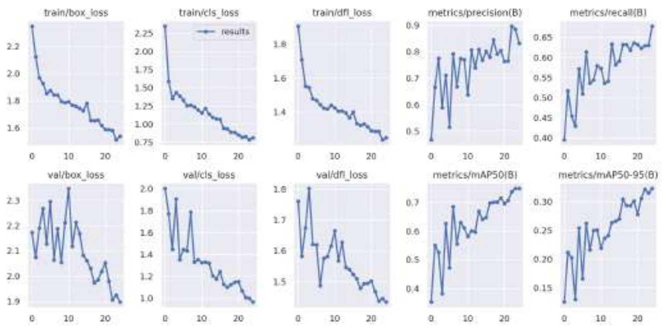


Fig. 11: Result Curves: YOLOV8

and development and lays the groundwork for more sophisticated and effective approaches to model training and real-world application. This discovery has profound implications for the field of AI, encouraging researchers to explore novel approaches and methodologies to enhance the capabilities of object recognition systems further. Such advancements hold the promise of revolutionizing various industries reliant on AI technology, from autonomous vehicles to medical imaging, ultimately benefiting society as a whole.

Each experiment was repeated multiple times, and the mean and standard deviation of precision, recall, and mAP scores were calculated. This approach ensures the robustness and consistency of the model performance across different runs.

IX. Future Work

Future work will focus on enhancing the real-time lion detection system by applying advanced model optimization techniques to improve both accuracy and efficiency. Techniques such as model pruning will be explored to reduce the model's size by removing redundant parameters, enabling faster processing without compromising detection quality. Quantization will be considered to lower the precision of model computations, which can significantly decrease memory usage and increase inference speed, making the system more suitable for deployment on low-power edge devices often used in remote forest areas. Knowledge distillation will also be investigated, where a smaller, lightweight model is trained to mimic a larger, more accurate model, achieving a balance between high detection performance and resource efficiency.

Furthermore, transfer learning with larger and more diverse wildlife datasets, including data from different lion populations and related species, will be employed to improve the model's generalization capability. This will help the system better handle variations in lion appearances, poses, and environmental conditions commonly found in forested areas near villages.

Opportunities for extension include incorporating edge computing platforms such as Jetson Nano and Raspberry Pi for on-site, real-time processing to reduce latency and dependence on network connectivity. Testing the model's detection performance on night-time infrared (IR) images will be valuable for improving monitoring during low-light conditions. Additionally, expanding the system to detect multiple animal species and leveraging temporal context from video sequences could enable path prediction and behavior analysis, further enhancing wildlife monitoring and safety efforts.

X. Conclusion

In conclusion, our training and assessment study comparing YOLOv5 and YOLOv8 for real-time lion identification has produced some exciting findings. We analysed the two models and discovered that, for this specific use case, there was no appreciable difference in their performances. However, the more advanced YOLOv8 version demonstrated a slight edge in accuracy and precision.

Beyond creating efficient lion detection technology, we work on other projects. Rather, we have used state-of-the-art technology to solve a critical problem—namely, the conflict between people and wild animals in the Gir region. Our lion detection algorithms have been integrated with APIs (such as Twilio) to develop a proactive system that can notify appropriate authorities and stakeholders immediately upon the identification of a lion sighting. This ingenious strategy not only saves the lives of humans and lions but also emphasises how important AI-powered solutions are to reducing human-animal conflict.

This research demonstrates how state-of-the-art computer vision technology can safeguard humans and wildlife. Applying these technologies in real-time scenarios has the potential to protect endangered species like lions and encourage harmonious coexistence between local inhabitants and wildlife.

It is important to note that this study specifically focuses on the Gir lion region, where we collected and trained our model on extensive, region-specific data. This focused approach enables us to implement the detection system effectively exactly where Gir lions are known to roam, maximizing its practical impact. While the current implementation is tailored for this area, the methodology and technology lay a strong foundation for future adaptation and expansion to other regions and species as more data becomes available.

Humans' ability to safeguard the environment and its incredible inhabitants is always evolving along with technology.

References

[1] "Managing human-lion conflict." https://cbs.umn.edu/lion-research-center/lion-research/managing-human-lion-conflict.

[2] O. M. Khare, S. Gandhi, A. M. Rahalkar, and S. Mane, "Yolov8-based visual detection of road hazards: Potholes, sewer covers, and manholes," *arXiv preprint arXiv:2311.00073*, 2023.

[3] D. Reis, J. Kupec, J. Hong, and A. Daoudi, "Real-time flying object detection with yolov8," *arXiv preprint arXiv:2305.09972*, 2023.

[4] A. Rosebrock, "Yolo object detection with opencv." https://pyimagesearch.com/2018/11/12/yolo-object-detection-with-opencv/, 2018.

[5] D. Ma and J. Yang, "Yolo-animal: An efficient wildlife detection network based on improved yolov5," in *2022 International Conference on Image Processing, Computer Vision and Machine Learning (ICICML)*, pp. 464–468, IEEE, 2022.

[6] M.-L. Lee, S. Behdad, X. Liang, and M. Zheng, "A real-time receding horizon sequence planner for disassembly in a human-robot collaboration setting," in *International Symposium on Flexible Automation*, vol. 83617, p. V001T04A004, American Society of Mechanical Engineers, 2020.

[7] T. Zhao, X. Yi, Z. Zeng, and T. Feng, "Mobilenet-yolo based wildlife detection model: A case study in yunnan tongbiguan nature reserve, china," *Journal of Intelligent & Fuzzy Systems*, vol. 41, no. 1, pp. 2171–2181, 2021.

[8] J. Zhong, M. Li, J. Qin, Y. Cui, K. Yang, and H. Zhang, "Real-time marine animal detection using yolo-based deep learning networks in the coral reef ecosystem," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 46, pp. 301–306, 2022.

[9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[10] R. Kaplan, C. Sauer, and A. Sosa, "Beating atari with natural language guided reinforcement learning," *arXiv preprint arXiv:1704.05539*, 2017.

[11] B. Dave, M. Mori, A. Bathani, and P. Goel, "Wild animal detection using yolov8," *Procedia Computer Science*, vol. 230, pp. 100–111, 2023.

[12] J. Qin, X. Yang, T. Zhang, and S. Bi, "Bi-tst_yolov5: Ground defect recognition algorithm based on improved yolov5 model," *World Electric Vehicle Journal*, vol. 15, no. 3, p. 102, 2024.

[13] D. Thuan, "Evolution of yolo algorithm and yolov5: The state-of-the-art object detention algorithm," 2021.

[14] J. Terven and D. Cordova-Esparza, "A comprehensive review of yolo: From yolov1 to yolov8 and beyond," *arXiv preprint arXiv:2304.00501*, 2023.

[15] A. D. Shetty and S. Ashwath, "Animal detection and classification in image & video frames using yolov5 and yolov8," in *2023 7th International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, pp. 677–683, IEEE, 2023.

[16] M. S. Norouzzadeh, A. Nguyen, M. Kosmala, A. Swanson, C. Packer, and J. Clune, "Automatically identifying wild animals in camera trap images with deep learning," in *Proceedings of the National Academy of Sciences*, vol. 115.

[17] "Yolo github repository." https://github.com/ultralytics/ultralytics.

[18] "Makesense." https://www.makesense.ai/.

[19] J. Yang, W. Zhu, T. Sun, X. Ren, and F. Liu, "Lightweight forest smoke and fire detection algorithm based on improved yolov5," *PLoS one*, vol. 18, no. 9, p. e0291359, 2023.

[20] X. Jia, Y. Tong, H. Qiao, M. Li, J. Tong, and B. Liang, "Fast and accurate object detector for autonomous driving based on improved yolov5," *Scientific reports*, vol. 13, no. 1, p. 9711, 2023.

[21] H. Yar, Z. A. Khan, F. U. M. Ullah, W. Ullah, and S. W. Baik, "A modified yolov5 architecture for efficient fire detection in smart cities," *Expert Systems with Applications*, vol. 231, p. 120465, 2023.

[22] Z. Ye, Q. Guo, J. Wei, H. Zhang, L. Bian, and S. Cao, "Recognition of terminal buds of densely-planted chinese fir seedlings using improved yolov5 by integrating attention mechanism," *Frontiers in Plant Science*, vol. 13, p. 991929, 2022.

[23] I. P. Sary, S. Andromeda, and E. U. Armin, "Performance comparison of yolov5 and yolov8 architectures in human detection using aerial images," *Ultima Computing: Jurnal Sistem Komputer*, vol. 15, no. 1, pp. 8–13, 2023.

[24] F. M. Talaat and H. ZainEldin, "An improved fire detection approach based on yolo-v8 for smart cities," *Neural Computing and Applications*, vol. 35, no. 28, pp. 20939–20954, 2023.

[25] T. Ahmad, Y. Ma, M. Yahya, B. Ahmad, S. Nazir, and A. u. Haq, "Object detection through modified yolo neural network," *Scientific Programming*, vol. 2020, pp. 1–10, 2020.

[26] N. E. Ramli, Z. R. Yahya, and N. A. Said, "Confusion matrix as performance measure for corner detectors," *Journal of Advanced Research in Applied Sciences and Engineering Technology*, vol. 29, no. 1, pp. 256–265, 2022.