MALAYSIAN JOURNAL OF SCIENCE
MJS
ISSN 1394 - 3065

5th International Conference on Applied & Industrial Mathematics and Statistics 2025 (ICoAIMS 2025)

Special Issue

Malaysian Journal of Science 45 (Special Issue 2): 25-36 (November 2026) https://mjs.um.edu.my

Evaluation of a Harmful-Animal Detection Method Combining Techniques for Object Detection and Skeleton Estimation

Shinichi Bota[1a]* and Naoyuki Morimoto[2a]

Abstract: This paper presents a new harmful-animal detection method that combines conventional algorithms for image recognition with skeleton estimation techniques. In many places around the world, wild animals encroach on human territory, causing personal injury and damage to crops. One way to address this social problem is to implement a harmful-animal detection system using cameras and object detection algorithms; however, current methods face the challenge of low detection accuracy, limiting their practical application. This study aims to address this challenge through a novel approach that combines conventional algorithms for image recognition with skeleton estimation techniques. The proposed method uses transfer learning on two pre-trained models: YOLOv8—known for its high accuracy and real-time performance—and HRNet, a widely used model for animal skeleton estimation. For training, a dataset extracted from AP-10K was used. In the proposed method, animals in the input image are detected by combining the inference results from two image recognition algorithms. The initial result is obtained from the input image itself, while the subsequent result is derived from a skeleton image created by plotting estimated keypoints and connecting lines. The final inference result is weighted and combined to improve the accuracy of animal detection in the image. The proposed method was evaluated by comparing its detection results with those of a conventional detection method while varying the minimum confidence threshold (*conf*). The evaluation demonstrated that the proposed method performs as well as or better than the conventional detection method. Furthermore, the proposed method exhibited greater overall stability, which simplified the process of *conf* parameter adjustment, making it more user-friendly. These findings suggest that the proposed method can contribute to the development of a more practical and effective harmful-animal detection system.

Keywords: image recognition, object detection, skeleton estimation, YOLOv8, AP-10K

1. Introduction

In many parts of the world, wild animals encroach on human territory, causing personal injury and significant damage to crops—one of the major social issues humanity currently faces. This phenomenon, commonly referred to as human–wildlife conflict, is believed to be driven by competition between humans and wildlife for the same resources, whether food or space (König et al., 2021).

For example, in Malaysia, data up to 2019 show an increasing trend in human–wildlife conflicts since 2012 (Xin et al., 2024), highlighting the persistent nature of this problem. Moreover, wild animals entering human territory often need to be culled, further complicating the situation. As land development and environmental changes caused by global warming continue, these conflicts are expected to persist. While discussions about coexistence between humans and wildlife have become more common in recent years, it is crucial to address this issue promptly to establish a society in which humans and animals can coexist.

In response to this problem, harmful-animal detection systems using cameras and object detection algorithms have been developed in the field of information technology; however, effective harmful-animal detection systems capable of significantly reducing wildlife damage have yet to be realized. One of the primary challenges lies in improving detection accuracy, which is hindered by the limited availability of data. The scarcity of animal data can be attributed to several factors: animals do not cooperate in data collection, the wide variety of animal species, and the low populations of some species, making it difficult to gather large datasets (Jiang et al., 2022). Previous studies have explored methods addressing data scarcity, such as data augmentation (Kumar et al., 2024) and synthetic datasets combining generated images (Mu et al., 2020); however, synthetic data, while promising, introduces biases that are difficult to eliminate. Therefore, this study aims to develop a harmful-animal detection system using an alternative approach to address the data scarcity issue.

This paper proposes a new method for animal detection that combines conventional algorithms for image recognition with skeleton estimation techniques. While prior studies have improved detection accuracy using keypoints or applied detected keypoints for behavior classification and analysis (Chen et al., 2023), this study focuses instead on inferring animal types directly from estimated keypoint information. To the best of our knowledge, no previous study has explored this approach. The method proposed in this study integrates the results of conventional detection algorithms with those obtained from

Authors information:
[a] 1577 Kurimamachiya-cho Tsu city, Graduate School of Engineering, Mie University, Mie, JAPAN. E-mail: 425m527@m.mie-u.ac.jp[1], morimoto@info.mie-u.ac.jp[2]

*Corresponding Author: 425m527@m.mie-u.ac.jp

Received: October 28, 2025
Accepted: May 6, 2026
Published: November 15, 2026

DOI: https://doi.org/10.22452/mjs.vol45sp2.3
Malaysian Journal of Science 45 (Special Issue 2): 25-36 (November 2026)

images annotated with skeleton-estimated information. We hypothesize that performance could be improved primarily by utilizing inference results from conventional detection algorithms and supplementing them with inference results derived from skeleton estimation techniques.

Skeleton estimation techniques are used to predict the positions of keypoints from images or videos. During animal detection using YOLO, bounding boxes often accurately detect the size of the detected object; however, they are prone to mistakenly including background elements or incorrectly inferring the animal class. To address this, we propose combining skeleton estimation techniques to increase the information available for inference. Incorporating skeleton information can mitigate the issue of data scarcity and enhance detection robustness. In addition, conventional detection methods often require the optimization of multiple parameters for prediction, such as the minimum confidence threshold parameter. We aim to address this challenge as well.

In this study, we focus on detecting the following classes: cow, horse, and bear. Bears were chosen because of the high risk they pose to humans. If a human accidentally encounters a bear, it is difficult to escape or defend oneself. For example, in Japan and Malaysia, bears are responsible for numerous incidents of crop damage and personal injuries annually (Xin et al., 2024). Cows and horses were selected because their skeletal structures are similar to those of bears and because they inhabit similar regions. Since our method incorporates skeleton information, demonstrating its effectiveness in animals with similar skeletal structures could indicate its broader applicability.

The proposed method uses models that are transfer-learned from the pre-trained models of YOLOv8 (Jocher et al., 2023) and HRNet-w48. YOLOv8 is known as an object detection algorithm with excellent accuracy and real-time performance. HRNet-w48 is a high-accuracy model included in MMPose (MMPose Contributors, 2020), a skeleton estimation algorithm library. The dataset used for training is AP-10K (Yu et al., 2021), which includes 23 animal families and 54 species with high-quality keypoint annotations.

To evaluate the proposed method, we compare its performance with a conventional detection method by varying *conf*, a parameter that determines the minimum confidence threshold. Performance is measured using standard metrics, such as precision, recall, and F1 score. The ANIMAL-POSE DATASET (Cao et al., 2019) is used for performance evaluation.

The results demonstrated that the proposed method achieved higher and more stable evaluation metric values than the conventional detection method. Although the proposed method also required the optimization of several parameters, the changes in performance caused by varying these parameters were smaller than those observed with the conventional detection method. This indicated that the proposed method ensured a certain level of accuracy without extensive parameter optimization, making it more user-friendly. Furthermore, when comparing the average performance of both methods, the conventional detection method exhibited a slightly higher recall. However, the proposed method achieved higher precision, resulting in a higher F1 score. Therefore, the proposed method can be considered slightly superior to the conventional detection method.

2. Related Research

Dave et al. (2023) used YOLOv8 to improve the detection accuracy of lion, tiger, bear, and leopard classes using YOLOv8 models, including YOLOv8m, YOLOv8l, and YOLOv8x. YOLO was chosen because of its proven high accuracy in the field of real-time object detection. The authors noted that model performance could be enhanced through data augmentation techniques. Inspired by this research, we decided to adopt YOLOv8 in our study as well; however, unlike Dave et al. (2023), we chose to use YOLOv8n. We also aimed to improve accuracy by using a distinct approach from data augmentation techniques.

Furthermore, using skeleton estimation techniques for detection, Chen et al. (2023) proposed an animal motion prediction model combining object detection and pose estimation to enable autonomous driving systems to assess animal postures more accurately. Initially, they used a convolutional neural network–based object detection method to detect animals in images. They then employed an animal pose estimation method to extract keypoints from the detected animal images. Finally, the images were classified into three categories of motion: running, walking, and standing. After extracting the keypoints, the authors used ResNet and MLP models to classify the pose. The MLP model classifies actions based on the angles of the keypoints. While their study focused on classifying behaviors from keypoint information, our study aims to infer the type of animal from the obtained keypoint data.

3. Proposed Method

3.1 Overview

For the proposed method, a system was constructed that primarily utilizes inference results from conventional detection algorithms and supplements them with inference results derived from skeleton estimation techniques. The overall structure of the proposed method is shown in Figure 1.

This method consists of a combination of three main algorithms: an image recognition algorithm, a skeleton estimation algorithm, and a skeleton image recognition algorithm.

The image recognition algorithm detects animals in the input image and outputs the predicted class, bounding box size, and confidence score. This process is identical to that of standard object-detection algorithms. Additionally, for each detected bounding box, the input image is cropped, and the cropped image is stored. The skeleton estimation algorithm estimates the skeleton from the input image. Similar to the image recognition algorithm, it detects animals in the input image and outputs the predicted class, bounding box size, and confidence score; however, unlike the image recognition algorithm, the input image for this algorithm is a skeleton image generated by overlaying the keypoints predicted by the skeleton estimation algorithm and the

DOI: https://doi.org/10.22452/mjs.vol45sp2.3
Malaysian Journal of Science 45 (Special Issue 2): 25-36 (November 2026)

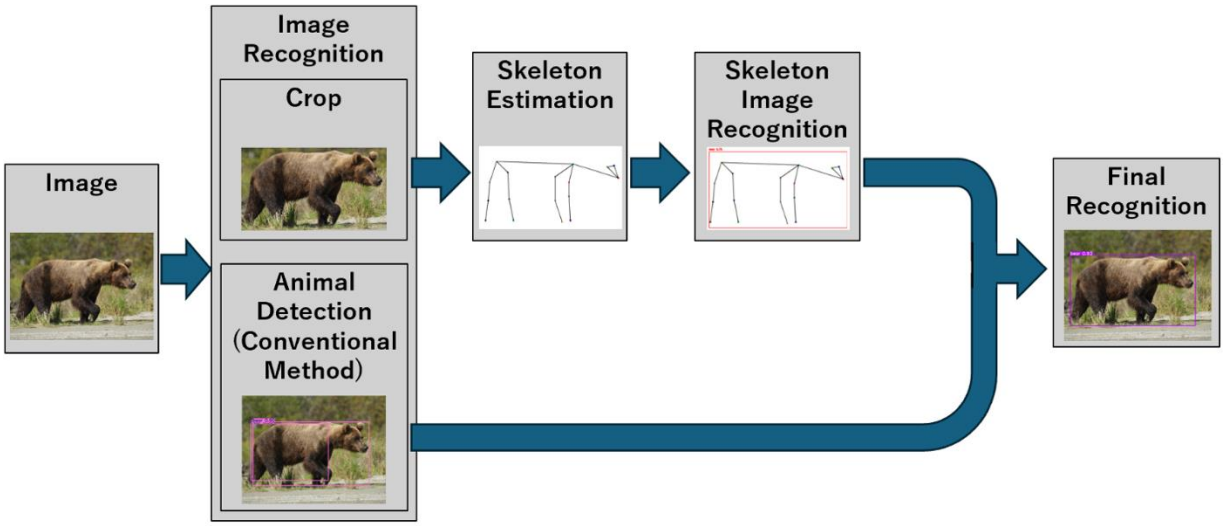


Figure 1. Overview of the proposed method

lines connecting these keypoints onto a white background. In this paper, such images are referred to as skeleton images.

Finally, the system outputs a weighted combination of the inference results from the image recognition algorithm and the skeleton image recognition algorithm.

3.2 Pre-trained Models for Each Algorithm

3.2.1 Dataset

This section describes the pre-trained models used by the three algorithms. The dataset used to create each pre-trained model was a modified version of the AP-10K dataset, containing only data for cows, horses, black bears, and brown bears, which had been extracted and processed accordingly. The authors of the AP-10K dataset explained in their paper that they had recruited 13 well-trained annotators and conducted three rounds of cross-checks and corrections both manually and automatically (Yu et al., 2021). Consequently, this dataset was deemed to possess high annotation quality and was selected for use in this study. While the proposed method uses skeletal information for classification, it was determined that there is no distinction between the skeletons of black bears and brown bears. Therefore, the data for these two species were combined into a new class labeled “bear.” The data corresponding to the three labels were selected from AP-10K and randomly divided into training, validation, and test sets. These data were split into three sets: 60% for training, 20% for validation, and 20% for testing, a partitioning scheme commonly used in machine learning research.

The models utilized for the image recognition and skeleton estimation algorithms were trained using a dataset derived from the AP-10K dataset. In this process, the original images and their corresponding label information from AP-10K were converted into YOLO format. For the skeleton image recognition algorithm, the model was trained on a dataset composed of skeleton images generated from keypoint information extracted from AP-10K, along with label information converted into the YOLO format. In the skeleton images, each keypoint was assigned a distinct color, and the keypoints were connected based on the label information, as illustrated in Figure 2. It was assumed that there might be patterns in the distances between and positions of the keypoints; therefore, the keypoints were color coded to facilitate learning.

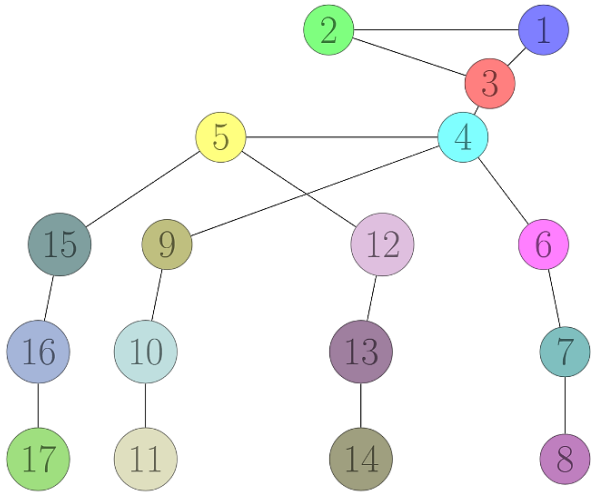


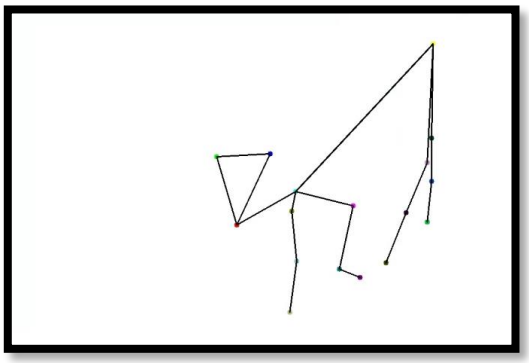
Figure 2. Keypoint connections

To provide a concrete example, the dataset includes images such as the one shown in Figure 3a. Based on the keypoint information extracted from the labels for this image, a corresponding skeleton image was generated, as shown in Figure 3b. The dataset comprising images such as those in Figure 3a was used to train the models for the image recognition algorithm and the skeleton estimation algorithm, whereas the dataset comprising skeleton images, represented in Figure 3b, was employed to train the model for the skeleton image recognition algorithm.

DOI: https://doi.org/10.22452/mjs.vol45sp2.3
Malaysian Journal of Science 45 (Special Issue 2): 25-36 (November 2026)



(a) Base image



(b) Skeleton image

Figure 3. Examples of images used for training each model

3.2.2 Model Training

The image recognition algorithm model was created by applying transfer learning to YOLOv8n, which is known for its high accuracy and real-time performance. This algorithm is the same as the one used for the conventional detection method. The details of its use as a conventional detection method are explained later. To ensure a fair comparison between the proposed method and the conventional detection method, both methods were trained using the same parameters. Transfer learning was conducted using YOLO's default settings, except for the number of epochs, to achieve general animal detection.

Regarding the skeleton estimation algorithm, although several such models exist—including pre-trained models trained on the AP-10K dataset within MMPose—we chose not to use them. Instead, we developed our own pre-trained model. This decision was made because we believe that the accuracy of skeleton estimation would have a significant impact on the results of the proposed method. To improve the accuracy of skeleton estimation, we reduced the number of animal species in the transfer learning dataset from 54 to three and developed a new model.

The model used in the skeleton estimation algorithm was created by applying transfer learning to the pre-trained model hrnet_w48-8ef0771d.pth, available from the MMPose library, which specializes in skeleton estimation algorithms. The code for transfer learning using the HRNet-w48 pre-trained model and the AP-10K dataset is provided in MMPose's GitHub repository (MMPose Contributors, 2020). Using this code as a reference, we implemented the skeleton estimation algorithm.

Similar to the image recognition algorithm, the skeleton image recognition algorithm model was created by applying transfer learning to YOLOv8n. The parameters for training were the same, except for the number of epochs, which were adjusted. As previously mentioned, the dataset used for transfer learning consisted of skeleton images. The representative parameters used for training the models are presented in Table 1.

Table 1. Parameters

Argument	Image recognition algorithm	Skeleton estimation algorithm	Skeleton image recognition algorithm
Epochs	1000	840	1000
Batch	16	32	16
Image size	640×640	256×256	640×640
Optimizer	Auto	Adam	Auto

3.3 Novel Mechanism of the Proposed Method

3.3.1 Skeleton Estimation Method

The most novel aspect of the proposed method is the integration of skeleton estimation techniques. This section explains the skeleton estimation method, followed by a discussion in the next section on how the inference results from the original and skeleton images are integrated. Finally, the execution flow of the proposed method is described.

The pre-trained model employed in the skeleton estimation algorithm follows a top-down approach. A key assumption of top-down skeleton estimation methods is that they expect a single instance to be present in the input. Therefore, when using such approaches, it is common to initially detect objects using an object detection algorithm, followed by keypoint detection for each object within its bounding box.

In the proposed method, the image recognition algorithm is executed prior to the skeleton estimation algorithm. During this process, the images are cropped according to the inferred bounding boxes, and the resulting cropped images are stored. These cropped images are then used as inputs to the skeleton estimation algorithm. In other words, the input images are the cropped outputs of the image recognition algorithm, with each one expected to contain only a single animal.

The skeleton estimation algorithm assumes that the input image contains only a single animal and estimates its skeleton. For keypoints in the estimation results with a confidence score greater than 0.5, the following processing steps are performed:

- Keypoints are displayed on a white background image the same size as the cropped image, with each keypoint assigned a different color.
- When keypoints are adjacent, they are connected according to Figure 2.

DOI: https://doi.org/10.22452/mjs.vol45sp2.3
Malaysian Journal of Science 45 (Special Issue 2): 25-36 (November 2026)

In MMPose, the *visualize* function, which performs visual prediction, has a default keypoint visualization threshold of 0.3. For the proposed method, this threshold was set to a higher value of 0.5. The rationale for this decision is that skeleton estimation results are used for subsequent inferences. Using data with low confidence scores would imply relying on data that were likely to have been misdetected during skeleton estimation. If such misdetected data were treated as correct data in later steps, they might negatively impact the accuracy of subsequent inference results. Therefore, the threshold was set higher to minimize this risk.

3.3.2 Integration Method

The inference results from the image recognition and skeleton image recognition algorithms are integrated under specific conditions. To begin with, when a bounding box is detected from the skeleton image estimated by the skeleton estimation algorithm, the results are discarded if the area of the inferred bounding box is less than half the area of that detected by the image recognition algorithm from the input image. This condition was established because when cropped images were created using the image recognition algorithm, there were cases in which the cropped images contained multiple animals or none at all.

The size of the bounding box to be output as the final inference result was set to the size inferred during the execution of the image recognition algorithm. The primary reason for this decision was the difficulty of determining the correct bounding box size from the skeleton image. Skeleton images generated by the skeleton estimation algorithm sometimes contain only a few keypoints, depending on the image. This occurs due to the high threshold setting or the presence of challenging images for estimation. In such cases, inferring the bounding box from the skeleton image becomes problematic. Furthermore, when inferring the bounding box using the skeleton image recognition algorithm, the bounding box is detected within the cropped image. Therefore, the inferred bounding box is, at most, the size of the bounding box detected by the image recognition algorithm, but is usually smaller. As a result, the size of the bounding box to be output as the final inference result was set to the size inferred by the image recognition algorithm.

When the results are integrated, a weighted sum is calculated. The confidence score inferred from the input image is multiplied by 0.7, while the confidence score inferred from the skeleton image is multiplied by 0.3, and the results are summed. The reason for assigning more weight to the confidence score inferred from the input image is that the method's purpose is to rely primarily on the results of conventional detection algorithms, with the results obtained from skeleton estimation techniques serving as supplementary information to improve performance. Under these conditions, the optimal ratio was determined using the evaluation method described in this paper. Consequently, a ratio of 0.7 for the confidence score inferred from the input image and 0.3 for that inferred from the skeleton image was adopted, as it yields the most balanced overall performance. The evaluation results are provided in Appendix Table A. The mathematical formulation of the integration method is presented in Equation (1).

$$\text{score}(R) = \begin{cases} \text{score}(R_1) \times 0.7 + \text{score}(R_2) \times 0.3, \text{if bbox}(R_2) \geq \frac{1}{2} \times \text{bbox}(R_1) \\ \text{score}(R_1) \times 0.7, \text{if bbox}(R_2) < \frac{1}{2} \times \text{bbox}(R_1) \end{cases} \quad (1)$$

In this formulation, R is the final detection result, $\text{score}(R)$ is the final confidence score, R_1 is the detection result from the original image, $\text{score}(R_1)$ is the confidence score of R_1, and $\text{bbox}(\text{R}_1)$ indicates the area of the bounding box derived from the original image. Likewise, R_2 is the detection result from the skeleton image, $\text{score}(\text{R}_2)$ is the confidence score of R_2, and $\text{bbox}(\text{R}_2)$ indicates the area of the bounding box derived from the skeleton image.

Since the final inference results of the proposed method aim to be in the same output format as the YOLO inference results, the summed confidence score is also expressed between 0 and 1, mirroring the YOLO confidence score range. However, it should be noted that because the inference method of YOLO and the calculation method of the proposed method differ, the confidence score output of the proposed method cannot be directly compared to that of YOLO.

3.3.3 Execution Flow

The proposed method assumes 2D images as inputs. First, the image recognition algorithm is executed. Using the pre-trained model, the algorithm then detects animals within each input image. The output includes the class of the detected animal, the bounding box size, and the confidence score for each detection. The confidence score is compared to a predefined threshold. If the score exceeds this threshold, the label information and the cropped image, obtained by trimming the region corresponding to the inferred bounding box, are stored. These cropped images are then used by the skeleton estimation algorithm. The inference results from this step are combined with those from the skeleton image recognition algorithm to detect the animal present in the image.

Next, the skeleton estimation algorithm is executed. In this algorithm, a pre-trained model is employed to estimate the skeleton for each cropped image stored by the image recognition algorithm. The output consists of the positions of keypoints and their respective confidence scores. Keypoints with confidence scores exceeding a threshold of 0.5 are plotted on a white background image the same size as the cropped image. Additionally, the keypoints are connected by lines based on their numbering. The resulting image, which includes both keypoints and connecting lines, is stored as the output of this algorithm. This output image is a skeleton image in which the keypoints and the lines connecting them are drawn on a white background matching the size of the input image.

DOI: https://doi.org/10.22452/mjs.vol45sp2.3
Malaysian Journal of Science 45 (Special Issue 2): 25-36 (November 2026)

Finally, the skeleton image recognition algorithm is executed. This algorithm uses a pre-trained model to infer the animal in each stored skeleton image. The output consists of the animal class, the bounding box size, and the confidence score, similar to the output of the image recognition algorithm. If the confidence score exceeds the predefined threshold, the inference results are stored.

The inference results from the image recognition and skeleton image recognition algorithms are integrated. For each cropped image, the bounding boxes detected by both algorithms are compared. If the overlap between the bounding boxes from the image recognition and skeleton image recognition algorithms exceeds a predefined threshold of 0.5, the confidence scores from both algorithms are weighted and summed. The confidence score from the image recognition algorithm is assigned a weight of 0.7, while that from the skeleton image recognition algorithm is assigned one of 0.3. After calculating the weighted sum, the animal class with the highest confidence score is selected. A text file is generated containing three pieces of information: the detected animal class, the bounding box size detected by the image recognition algorithm, and the highest confidence score.

The confidence score recorded in the text file is compared to the predefined threshold. If the score falls below this threshold, the cropped image is classified as containing no animal, and the corresponding information in the text file is deleted. If the score exceeds the threshold, the information in the text file is retained as the label for the cropped image. The labels are then organized according to the inference results from the image recognition algorithm, and a label is generated for each input image. An image is generated displaying labels following the output format of YOLO. Each label contains the following information: class information, x center, y center, width, height, and confidence score. The execution flow of the proposed model is presented as a pseudocode in Algorithm 1.

4. Performance Evaluation and Comparison

4.1 Methods for the Evaluation Experiment

To conduct a performance evaluation, a labeled dataset is essential, as it serves as the ground truth for assessing the performance of predicted labels. In this study, the evaluation dataset was derived from the ANIMAL-POSE DATASET, from which only data for three animal classes—cows, horses, and bears—were extracted for the evaluation.

Three widely used metrics in machine learning—precision, recall, and F1 score—were employed for performance evaluation. During the evaluation, the intersection-over-union threshold was set to 0.5. The predicted and ground truth classes were compared to assess performance only if the bounding box overlap exceeded 50%. In YOLOv8, one of the inference arguments was *conf*, which sets the minimum confidence threshold for detection. The default value of this parameter was set to 0.25. In our study, this value was varied for evaluation purposes. For each minimum confidence threshold, evaluation results were calculated for each class as well as for the overall average.

A detailed explanation of the performance evaluation follows. By varying the minimum confidence threshold, the conventional detection method and the proposed method were evaluated using three metrics. The results were compared to assess the effect of combining skeleton estimation techniques. For each method, the minimum confidence threshold was set to [0.2, 0.25, 0.3, 0.4, 0.5, 0.6], including a default value of 0.25. The conventional detection method has only one place for setting the *conf* parameter; hence, it calculates six evaluation results. In contrast, the proposed method has three locations for setting it; hence, 216 evaluation results are computed. Specifically, the *conf* parameter is set at three distinct points: the image recognition algorithm, the skeleton image recognition algorithm, and the weighted integration of the results from each algorithm.

As a method of performance comparison, this study employed box plots focusing on the overall mean evaluation results—specifically, macro precision, macro recall, and macro F1 scores. Initially, a comprehensive performance comparison was conducted between the conventional detection method and the proposed method. Three tables are presented, corresponding to macro precision, macro recall, and macro F1 scores, with each table containing two box plots. The first box plot represents the total results for the conventional detection method, where six evaluation results were obtained by varying the minimum confidence threshold and are grouped into a single box plot. The second box plot represents the total results for the proposed method, where 216 evaluation results were obtained by varying the minimum confidence threshold and are grouped into a single box plot.

Subsequently, the performance variations were examined using box plots when changing the *conf* parameter at the final integration stage of the proposed method, where the inference results from each algorithm are weighted and integrated. Three tables are presented, corresponding to macro precision, macro recall, and macro F1 scores. The proposed method was evaluated by varying the *conf* parameter in six ways, with 36 evaluation results grouped into a single box plot for each minimum confidence threshold. In general, the object detection algorithms showed that increasing the minimum confidence threshold above YOLO's default value improved performance. This was because precision increased as only highly confident detections were retained, whereas recall decreased because of a reduction in the number of detections. As a result, the F1 score, functioning as the harmonic mean of the two, improved, leading to a more balanced detector. The proposed method was examined to determine whether changes in the *conf* parameter led to variations similar to those observed in conventional detection methods.

Furthermore, the average evaluation results for each class and the overall average evaluation results of the conventional detection method and the proposed method are presented, allowing for a comparison of which method demonstrated superior performance on average.

DOI: https://doi.org/10.22452/mjs.vol45sp2.3
Malaysian Journal of Science 45 (Special Issue 2): 25-36 (November 2026)

Algorithm 1 Proposed Model

```
Input: image - Input image
Output: R_list - Final confidence scores

function Main(image)
    (cropped_images, labels) ← ExecuteImageRecognition(image)
    R_list ← []
    for i = 1 to length(cropped_images) do
        crop ← cropped_images[i]
        label ← labels[i]
        (confidence_original, size_original) ← ComputeConfidenceAndSize(label)
        keypoint_label ← ExecuteSkeletonEstimation(crop)
        filtered_keypoints ← FilterKeypointsByConfidence(keypoint_label, 0.5)
        skeleton_image ← OutputSkeleton(filtered_keypoints)
        label_skeleton ← ExecuteSkeletonImageRecognition(skeleton_image)
        (confidence_skeleton, size_skeleton) ← ComputeConfidenceAndSize(label_skeleton)
        if size_skeleton ≥ 0.5 × size_original then
            R ← 0.7 × confidence_original + 0.3 × confidence_skeleton
        else
            R ← 0.7 × confidence_original
        end if
        append(R, R_list)
    end for
    return R_list
end function

function ExecuteImageRecognition(image)
//Infer the animal from an image (conventional detection method)
     return (cropped_images, labels) // Multiple cropped images and their labels
end function

function ExecuteSkeletonEstimation(image)
//Estimate the skeleton from an image
    return keypoint_label // Contains 17 keypoints (x, y, confidence)
end function

function ExecuteSkeletonImageRecognition(image)
// Infer the animal from a skeleton image
    return label
end function

function ComputeConfidenceAndSize(label)
// Compute the confidence score and bounding box size from the label
    return (label.confidence, label.width × label.height)
end function

function FilterKeypointsByConfidence(keypoint_label, threshold)
// Find keypoints exceeding the threshold
    return {kp ∈ keypoint_label | kp.confidence > threshold}
end function

function OutputSkeleton(keypoints)
// Draw keypoints and the lines connecting them on a white background
    return skeleton_image
end function
```

4.2 Results and comparison

As mentioned earlier, the conventional detection method described here used the same pre-trained model as the image recognition algorithm used by the proposed method. Therefore, a model trained through transfer learning using YOLOv8n on the dataset extracted from AP-10K was used.

The performance results for each method across the three metrics are represented as box plots in Figure 4. In addition, the performance of the proposed method across the three metrics,

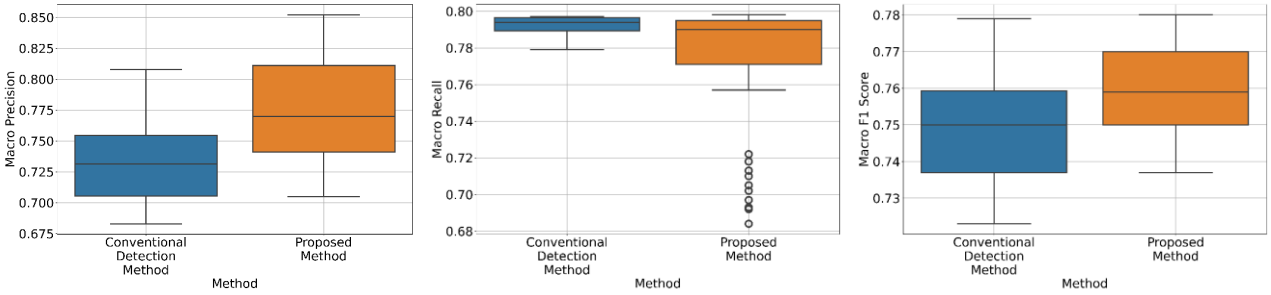
DOI: https://doi.org/10.22452/mjs.vol45sp2.3
Malaysian Journal of Science 45 (Special Issue 2): 25-36 (November 2026)

when the *conf* parameter was varied at the final integration stage, is represented as box plots in Figure 5. Table 2 shows the average performance results for each method across the three metrics.

According to Figure 4, the proposed method demonstrated superior overall performance and stability. This is shown by the smaller spread of the whiskers in the box plots and the numerical values, indicating that the proposed method achieved more stable and overall better performance. In Figure 4c, the maximum values for both the conventional and proposed methods are almost identical; however, the range of the box for the conventional method is wider, highlighting the increased stability of the proposed method. From Figure 4a and b, it can be observed that the proposed method achieved a higher precision than the conventional detection method, although the recall decreased slightly. Comparing these two figures, the box plots of precision and recall for the proposed method are both centered on 0.78, indicating that the proposed method maintained a well-balanced performance. Therefore, from Figure 4, it can be concluded that the proposed method exhibits both good balance and high stability.

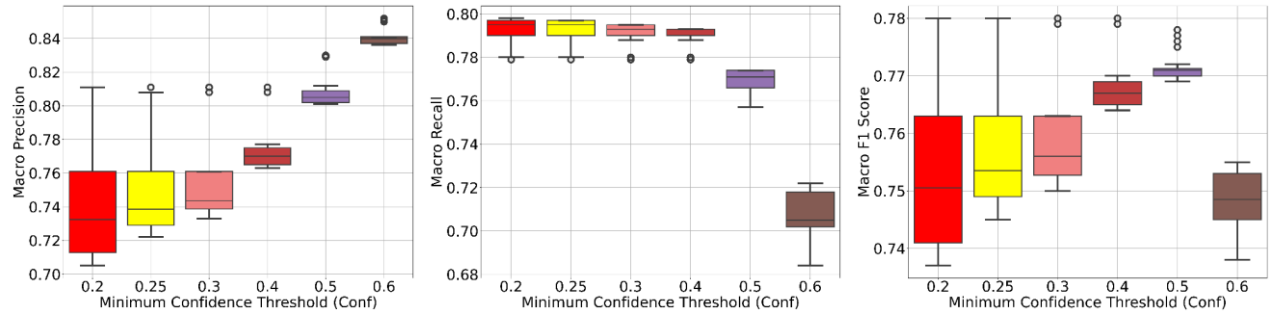
Figure 5 displays the performance variations of the proposed method by adjusting the *conf* parameter at the final integration stage, where the inference results from each algorithm are weighted and integrated. The data confirmed that the highest performance stability was achieved when the *conf* parameter was set to 0.5. Additionally, for scenarios in which recall performance was prioritized while maintaining stable detection, a *conf* parameter of 0.4 was an effective option.

Here, we discuss not only the overall evaluation results but also those for each animal class. Prior to this research, YOLO generally detected bounding box sizes with high accuracy, whereas the inference for animals was occasionally inaccurate. The data presented in Table 2 reveal that the conventional detection method showed higher performance for recall than for



(a) Macro precision (b) Macro recall (c) Macro F1 score

Figure 4. Performance comparison of the two different methods



(a) Macro precision (b) Macro recall (c) Macro F1 score

Figure 5. Performance of the proposed method for different *conf* parameters

Table 2. Mean Performance Difference Across Methods

Method	Class	Precision	Recall	F1 score	Macro precision	Macro recall	Macro F1 score
Conventional detection method	Cow	0.736	0.600	0.660	0.736	0.792	0.749
	Horse	0.586	0.898	0.709			
	Bear	0.885	0.876	0.880			
Proposed method	Cow	0.796	0.580	0.668	0.779	0.774	0.760
	Horse	0.619	0.895	0.731			
	Bear	0.921	0.845	0.880			

DOI: https://doi.org/10.22452/mjs.vol45sp2.3
Malaysian Journal of Science 45 (Special Issue 2): 25-36 (November 2026)

Table 3. Mean Performance Using Only Keypoint Information

Method	Class	Precision	Recall	F1 score	Macro precision	Macro recall	Macro F1 score
Skeleton	Cow	0.570	0.349	0.416			
Recognition	Horse	0.571	0.783	0.656	0.625	0.515	0.529
Method	Bear	0.734	0.414	0.514			

precision. Generally, precision and recall are regarded as being in a trade-off relationship, requiring a balance between these metrics to improve model performance. From Table 2, it is evident that the proposed method achieved a balanced trade-off between macro precision and macro recall. In addition, the proposed method outperformed the conventional detection method in terms of the F1 score for all classes, demonstrating stable and high performance across all classes. This indicates that the proposed method, by incorporating skeleton estimation techniques, successfully addresses the problem of detecting background objects or making incorrect inferences for animals, which is one of the key objectives of this study.

The results of inferring an animal solely from the estimated keypoint information are presented in Table 3.

As shown in Table 3, using the same evaluation method as that in Table 2, recall was low. These results suggest that inferring an animal solely from keypoint information is difficult. Thus, in animal detection, keypoint information should be utilized as supplementary data for inference, as employed in this study.

To assess the feasibility of the proposed method for harmful-animal detection, the execution time per image was measured. The experimental environment used for the evaluation is summarized in Table 4.

Table 4. Experimental Environment

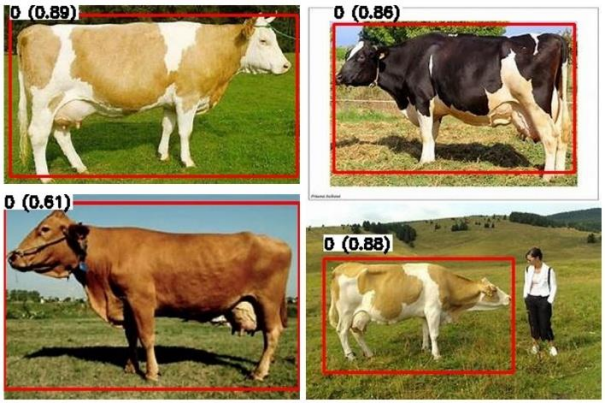
Environment	
CPU	Intel Core i5-12400 2.50 GHz
GPU	NVIDIA GeForce RTX 3080 10 GB VRAM
Memory	16 GB
Architecture	x86_64
Distribution	WSL-Ubuntu 22.04

The experimental measurements indicate that the proposed method required approximately 5.66 seconds per image. In contrast, using only the object detection model resulted in an execution time of approximately 2.33 seconds, representing a 2.4-fold increase. While it can be observed that the proposed method took longer to execute, the comparison results demonstrated that it achieved higher stability without requiring complicated optimization. This enhanced stability improves the robustness of the detection system.

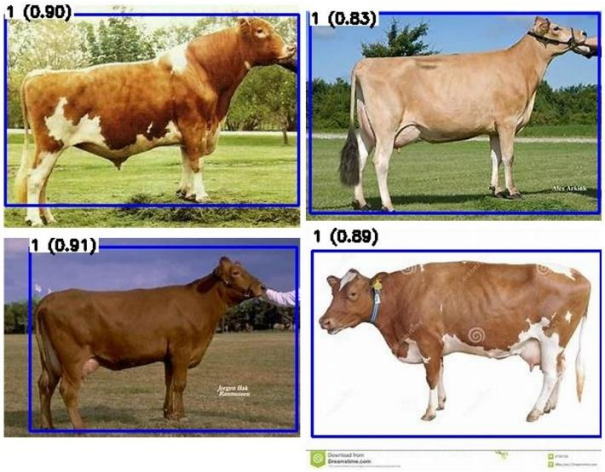
As a result, it is shown that stability can be increased by using skeletal estimation techniques, which we believe may be one way to address the problem of a lack of data. Therefore, it can be concluded that such techniques are highly effective for animal detection tasks.

In a qualitative analysis of the proposed model's performance, among the three categories, bears were classified with high accuracy, as indicated in Table 2. Since the ultimate goal of this study is to develop a method for detecting harmful animals, this result can be considered favorable. However, a notable concern is the low evaluation scores for cows. An analysis of the detection output revealed that cows were frequently misclassified as horses. Furthermore, both the conventional method and the skeleton estimation-based approach exhibited a tendency to do this. This outcome can be attributed to the close similarity in the appearance and skeletal structures of cows and horses.

An analysis was conducted to identify the characteristics of the images that were prone to misclassification. As shown in Figure 6a, images featuring a sagging neck region and a lateral extension of the neck were often classified correctly. In contrast, images without a sagging neck, as observed in Figure 6b, were frequently misclassified. This tendency can be attributed to the fact that horses generally exhibit less neck sagging than cows.



(a) When the detection is correct



(b) When the detection is incorrect

Figure 6. Qualitative analysis of detection

DOI: https://doi.org/10.22452/mjs.vol45sp2.3
Malaysian Journal of Science 45 (Special Issue 2): 25-36 (November 2026)

5. Conclusion

In this paper, we proposed a new method for animal detection that combines conventional algorithms for image recognition with skeleton estimation techniques. The objective of the proposed method is to improve detection accuracy by leveraging inference results from conventional detection algorithms and supplementing them with results derived from skeleton estimation techniques.

For this study, we focused on detecting three animal classes: cow, horse, and bear. Among the datasets containing these species, we utilized a high-quality annotated dataset extracted from AP-10K for training. To evaluate the model's performance, we employed a dataset extracted from the ANIMAL-POSE DATASET. The proposed method consists of a combination of three algorithms: an image recognition algorithm, a skeleton estimation algorithm, and a skeleton image recognition algorithm. The models used in the proposed method include transfer learning applied to pre-trained YOLOv8 and HRNet-w48 models. In generating skeleton images, we devised specific procedures for skeleton estimates outputs and threshold settings. Furthermore, during the integration of inference results from input images and skeleton images, we adjusted the weighting, threshold settings, and output methods throughout the integration process.

The proposed method and the conventional detection method were compared by varying the minimum confidence threshold to assess the performance differences. For the conventional detection method, a model trained via transfer learning on a pre-trained YOLO model using a dataset extracted from AP-10K was employed. The comparison results revealed that while the conventional detection method exhibited a significant performance gap between the nonoptimized and optimized parameter settings, the proposed method demonstrated a smaller performance difference between these two settings, indicating that it maintained relatively high performance, even without parameter optimization. In other words, the proposed method outperformed the conventional detection method in terms of performance stability. As a result, the proposed method can be regarded as a more user-friendly model, as it requires less parameter tuning than the conventional detection method.

The proposed method utilizes skeletal information for animal detection. Accordingly, experiments were conducted on animals exhibiting similar skeletal structures. As a result, the effectiveness of the proposed method was demonstrated for animals with similar skeletal features. This finding indicates that the method is likely to be applicable to animals with dissimilar skeletal structures, although such animals were not included in the dataset used in this study. Additionally, although the dataset primarily contained images of animals in a lateral standing position, it also included images of animals lying down, sitting, and running. Since the method successfully classifies these postures, it is anticipated that this approach can adapt to various animal postures.

Finally, we address the challenges and future perspectives. This study aimed to evaluate whether a combination of object detection and skeleton estimation techniques improves the performance of harmful-animal detection. To achieve this, YOLO, a widely recognized object detection model, and HRNet, a widely used skeleton estimation model, were employed. Future research should investigate which combination of object detection and skeleton estimation models achieve optimal performance for practical implementation. In particular, it is essential to evaluate the performance of transformer-based object detection models when combined with skeleton estimation models. Furthermore, while the proposed method integrates an object detection model with a skeleton estimation model, comparing its performance with methods that combine two object detection models could be an interesting direction for future research.

Ensuring the practical applicability of the proposed method requires addressing several challenges. One key consideration is the impact of low-quality images. In this study, training was conducted using high-quality images from AP-10K, which may have introduced bias. Consequently, when applied to real-world scenarios, preprocessing techniques such as super-resolution may be required to enhance image quality.

In addition, while this study focused solely on image-based systems, real-world applications of animal detection algorithms require compatibility with video processing and real-time inference. Although this study did not focus on real-time performance, there are situations in which real-time capabilities are necessary for practical application in the real world. Therefore, as a potential future direction, pursuing real-time performance should be considered.

Furthermore, although this research utilized 2D images for training, we believe that incorporating 3D images could further enhance the performance of skeleton estimation. Future research should also aim to broaden the scope of the animal species included in the dataset. Ultimately, the goal is to advance this system into a more practical and deployable solution for real-world applications.

5. Author Contributions

S. B. designed the study, developed the methodology, implemented the system, analyzed the data, evaluated the results, and wrote the original draft. N. M. supervised the research and managed the project. All authors revised the manuscript, approved the final version for publication, and agree to be accountable for all aspects of the work. They will ensure that any questions related to its accuracy or integrity are appropriately investigated and resolved.

6. References

Cao, J., Tang, H., Fang, H. S., Shen, X., Lu, C., & Tai, Y. W. (2019). Cross-domain adaptation for animal pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 9498–9507).

Chen, K., Yu, Y., & Zhou, H. (2023). Animal movement prediction based on the corroboration of object detection and pose estimation. *Highlights in Science, Engineering and*

DOI: https://doi.org/10.22452/mjs.vol45sp2.3
Malaysian Journal of Science 45 (Special Issue 2): 25-36 (November 2026)

*Technol*ogy, 39, 997–1004. https://doi.org/10.54097/hset.v39i.6692

Dave, B., Mori, M., Bathani, A., & Goel, P. (2023). Wild animal detection using YOLOv8. *Procedia Computer Science*, *230*, 100–111.

Jiang, L., Lee, C., Teotia, D., & Ostadabbas, S. (2022). Animal pose estimation: A closer look at the state-of-the-art, existing gaps and opportunities. *Computer Vision and Image Understanding*, *222*, 103483.

Jocher, G., Qiu, J., & Chaurasia, A. (2023). *Ultralytics YOLO* (Version 8.0.0) [Computer software]. https://github.com/ultralytics/ultralytics

König, H. J., Ceaușu, S., Reed, M., Kendall, H., Hemminger, K., Reinke, H., Ostermann-Miyashita, E.-F., Wenz, E., Eufemia, L., Hermanns, T., Klose, M., Spyra, M., Kuemmerle, T., & Ford, A. T. (2021). Integrated framework for stakeholder participation: Methods and tools for identifying and addressing human–wildlife conflicts. *Conservation Science and Practice*, *3*(3), e399.

Kumar, T., Brennan, R., Mileo, A., & Bendechache, M. (2024). Image data augmentation approaches: A comprehensive survey and future directions. *IEEE Access, 12,* 187536–187571.

MMPose Contributors. (2020). *OpenMMLab pose estimation toolbox and benchmark*. https://github.com/open-mmlab/mmpose

Mu, J., Qiu, W., Hager, G. D., & Yuille, A. L. (2020). Learning from synthetic animals. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 12386–12395).

Xin, L. Q., Mat Ghani, S. N. A., Rasudin, N. S., Abdul Ghafar, N., Abd Rashid, N. H., Ten, D. C. Y., Saaban, S., Edinur, H. A., & Abdullah, M. T. (2024). Characterisation of human–wildlife conflict and casualties caused by wildlife attacks in Peninsular Malaysia. *IIUM Medical Journal Malaysia*, *23*(01), 83–90.

Yu, H., Xu, Y., Zhang, J., Zhao, W., Guan, Z., & Tao, D. (2021). AP-10k: A benchmark for animal pose estimation in the wild. *arXiv preprint arXiv:2108.12617*.

DOI: https://doi.org/10.22452/mjs.vol45sp2.3
Malaysian Journal of Science 45 (Special Issue 2): 25-36 (November 2026)

7. Appendix

Table A. Evaluation Results for Determining the Weighting Ratio During Integration

Ratio	Class	Precision	Recall	F1 score	Macro precision	Macro recall	Macro F1 score
0.5_0.5	Cow	0.71	0.74	0.72	0.76	0.80	0.78
	Horse	0.76	0.83	0.79			
	Bear	0.82	0.83	0.82			
0.6_0.4	Cow	0.68	0.82	0.75	0.79	0.83	0.80
	Horse	0.76	0.81	0.78			
	Bear	0.92	0.84	0.88			
0.7_0.3	Cow	0.71	0.82	0.76	0.79	0.83	0.80
	Horse	0.75	0.81	0.78			
	Bear	0.90	0.84	0.87			
0.8_0.2	Cow	0.70	0.86	0.77	0.78	0.84	0.80
	Horse	0.76	0.81	0.78			
	Bear	0.87	0.84	0.86			
0.9_0.1	Cow	0.68	0.86	0.76	0.77	0.84	0.80
	Horse	0.75	0.81	0.78			
	Bear	0.87	0.84	0.86			

DOI: https://doi.org/10.22452/mjs.vol45sp2.3
Malaysian Journal of Science 45 (Special Issue 2): 25-36 (November 2026)