electronics

MDPI

Article

YOLOv8-PoseBoost: Advancements in Multimodal Robot Pose Keypoint Detection

Feng Wang [1,*], Gang Wang [2,3] and Baoli Lu [4,5,*]

[1] Engineering and Technology College, Hubei University of Technology, Wuhan 430068, China
[2] School of Computing and Data Engineering, NingboTech University, Ningbo 315100, China; wanggangnit@nit.zju.edu.cn
[3] Department of Bioengineering, Imperial College London, London SW7 2AZ, UK
[4] School of Computing, University of Portsmouth, Portsmouth PO1 3HE, UK
[5] Institute of Semiconductors, Chinese Academy of Sciences, Beijing 100083, China
* Correspondence: 20124364@hbut.edu.cn (F.W.); baoli.lu@port.ac.uk (B.L.)

Abstract: In the field of multimodal robotics, achieving comprehensive and accurate perception of the surrounding environment is a highly sought-after objective. However, current methods still have limitations in motion keypoint detection, especially in scenarios involving small target detection and complex scenes. To address these challenges, we propose an innovative approach known as YOLOv8-PoseBoost. This method introduces the Channel Attention Module (CBAM) to enhance the network's focus on small targets, thereby increasing sensitivity to small target individuals. Additionally, we employ multiple scale detection heads, enabling the algorithm to comprehensively detect individuals of varying sizes in images. The incorporation of cross-level connectivity channels further enhances the fusion of features between shallow and deep networks, reducing the rate of missed detections for small target individuals. We also introduce a Scale Invariant Intersection over Union (SIoU) redefined bounding box regression localization loss function, which accelerates model training convergence and improves detection accuracy. Through a series of experiments, we validate YOLOv8-PoseBoost's outstanding performance in motion keypoint detection for small targets and complex scenes. This innovative approach provides an effective solution for enhancing the perception and execution capabilities of multimodal robots. It has the potential to drive the development of multimodal robots across various application domains, holding both theoretical and practical significance.

Keywords: multimodal robots; pose keypoint detection; small object detection; CMBAY; YOLOv8

check for updates

Citation: Wang, F.; Wang, G.; Lu, B. YOLOv8-PoseBoost: Advancements in Multimodal Robot Pose Keypoint Detection. *Electronics* **2024**, *13*, 1046. https://doi.org/10.3390/electronics13061046

Received: 2 February 2024
Revised: 23 February 2024
Accepted: 28 February 2024
Published: 11 March 2024

1. Introduction

In today's era of rapid technological advancement, research in the field of multimodal robotics is increasingly becoming a focal point in both academic and industrial domains [1,2]. This area of study is dedicated to enabling robots to achieve a more comprehensive and accurate perception and understanding of their surrounding environment by simultaneously harnessing information from multiple sensors. Motion keypoint detection, as a prominent and critical issue within multimodal robotics research, directly pertains to the key nodes in a robot's perception of its own motion and its environment. It also significantly influences a robot's performance in handling complex tasks. Particularly in applications such as human–robot collaboration, environmental perception, and real-time decision-making, the accurate detection of motion keypoints plays a paramount role in the successful execution of tasks by robots [3,4]. Moreover, in applications like environmental perception and real-time decision-making, accurate motion keypoint detection forms the cornerstone of a robot's ability to navigate complex environments, avoid obstacles, and adapt its actions in response to dynamic changes in its surroundings. By leveraging information from multiple sensors, including cameras, LiDAR, and inertial measurement units

(IMUs), robots can capture rich and diverse data streams, enabling them to perceive and interpret their environment with unprecedented depth and accuracy.

However, despite significant advancements, there are still some shortcomings to address. Firstly, there are challenges in small target detection, limiting the performance of multimodal robots in perceiving small-sized objects [5,6]. This includes the capture and analysis of subtle motions, and current methods have not yet reached an ideal level when dealing with these small targets. This limitation may pose constraints for robots that need to operate in confined spaces or perform precise manipulations. Secondly, when facing the challenges of complex environments, the detection of motion keypoints by robots becomes even more challenging in highly dynamic and rapidly changing scenarios. This may involve complex situations such as interactions between multiple objects, changes in lighting conditions, and occlusions, making it difficult for traditional motion keypoint detection algorithms to achieve satisfactory results in such contexts. In these scenarios, robots may exhibit lower accuracy and robustness. Therefore, the goal of this research is to address these issues of small target detection and motion keypoint detection in complex environments through innovative approaches [7,8]. By overcoming these limitations, we aim to enhance the perception and execution capabilities of multimodal robots in real-world applications, enabling them to better adapt to complex and ever-changing environments.

In the past, there have been two primary approaches to pose estimation: Top-Down and Bottom-Up. The Top-Down approach is a paradigm of keypoint detection that starts from the overall target and gradually refines the localization of keypoints. In the context of motion keypoint detection in multimodal robots, the Top-Down approach begins with target identification and then precisely locates keypoints within the target region. This method excels in scenarios involving multiple targets, particularly in applications like human pose estimation. Initially, the Top-Down approach employs advanced object detectors such as Faster R-CNN or YOLO to identify target regions in the image [9,10]. This provides foundational information for subsequent keypoint localization, including the target's position and confidence score. Next, in the keypoint localization phase, specific keypoint localization networks like the Hourglass Network are used to make high-precision predictions for keypoints within the target region. Finally, post-processing and optimization steps are applied to ensure the accuracy and robustness of the detection results. The advantage of the Top-Down approach lies in its ability to adapt to situations requiring simultaneous handling of multiple targets, providing robust support for robots in complex tasks [11,12].

In contrast, the Bottom-Up approach represents another paradigm in keypoint detection. Its distinctive feature is the direct detection of all possible keypoints throughout the entire image, followed by the association of these points to form complete objects. In the context of motion keypoint detection for multimodal robots, the Bottom-Up method processes the entire image directly and is suitable for scenarios requiring dense keypoint detection, offering an efficient solution for robots tasked with complex and dense detection tasks. First, in the keypoint detection phase, a dense keypoint detector like OpenPose is employed to directly process the entire image, providing keypoint estimations for each pixel. Subsequently, through the association and merging phases, adjacent keypoints are connected to form the parts of a human body or other objects, resulting in a complete set of keypoints. Finally, posture evaluation and filtering steps assess various postures formed and select the most suitable ones. The advantage of the Bottom-Up approach lies in its direct processing of the entire image, making it well-suited for dense and complex scenes, thereby providing an efficient solution for robots in need of dense keypoint detection tasks.

However, both Top-Down and Bottom-Up approaches share common limitations in the field of motion keypoint detection. Firstly, they face challenges in small object detection, as both methods may be limited in perceiving small-sized targets. Capturing and analyzing fine-grained movements can be relatively difficult for these approaches. This limitation can be a constraint for robots that need to perform tasks or precise manipulations in confined spaces. Secondly, they both encounter challenges when dealing with complex scenes. In highly dynamic and rapidly changing environments, both Top-Down and Bottom-Up approaches may be disrupted by factors like interactions between multiple objects, variations in lighting conditions, and occlusions. These complexities make traditional motion keypoint detection algorithms struggle to achieve ideal performance in such scenarios, potentially resulting in reduced accuracy and robustness for robots in these challenging environments.

To address the aforementioned constraints, we present YOLOv8-PoseBoost. This model introduces the Channel Attention Module (CBAM) to sharpen the network's focus on small targets and boost sensitivity to small-sized pedestrians without significantly escalating computational complexity. Moreover, it integrates four distinct-sized detection heads within the backbone network, empowering the algorithm to thoroughly identify pedestrians of diverse sizes in images. Subsequently, we introduce dual cross-level connectivity channels between the backbone network and the neck, augmenting feature fusion capabilities across shallow and deep networks, thereby enhancing information exchange and mitigating missed detections for small-sized pedestrians. Additionally, we integrate the Scale Invariant Intersection over Union (SIoU) to redefine the bounding box regression loss function, expediting training convergence and refining detection accuracy. These pioneering strategies aim to surmount prior method limitations, ultimately resulting in YOLOv8-PoseBoost's superior performance in detecting small targets and motion keypoints within intricate scenes.

- This paper introduces the YOLOv8-PoseBoost model, which enhances the network's ability to focus on small targets and increase sensitivity to small-sized pedestrians by incorporating the CBAM attention mechanism module, employing multiple scale detection heads, and optimizing the bounding box regression loss function (SIoU).
- To further improve the network's feature fusion capabilities and reduce the rate of missed detections for small-sized pedestrians, this study establishes two cross-level connectivity channels between the backbone network and the neck. Such structural innovations contribute to enhanced model performance in complex scenes.
- The introduction of the SIoU-redefined bounding box regression loss function not only accelerates training convergence but also enhances the accuracy of motion key points detection. These advancements provide a more efficient and precise solution for practical applications, particularly in the domains of small target detection and

In the upcoming article structure, we will organize the content as follows: Section 2 will provide a detailed overview of related work. Section 3 will delve into the key details of our proposed model. Section 4 will focus extensively on our experimental design and results. Finally, Section 5 will serve as the conclusion and discussion of this research.

2. Related Work

2.1. Based on the Top-Down Pose Estimation Method

The Top-Down pose estimation method is a paradigm widely applied in multimodal robot motion keypoint detection. In this field, there are several renowned Top-Down methods, including CPN (Cascade Pyramid Network), Hourglass Network, CPM (Convolutional Pose Machines), and Alpha Pose, among others. These methods have achieved significant accomplishments in motion keypoint detection [1].

Firstly, CPN employs a cascade pyramid structure, enhancing accuracy by progressively refining the location information of motion keypoints through layered pyramid feature extraction. This method exhibits robustness and is suitable for complex scenes and small target detection. Secondly, the Hourglass Network is a classic Top-Down approach characterized by its symmetrical encoding and decoding structure, enabling precise localization of motion keypoints at different scales. The Hourglass Network is commonly used in human pose estimation, with its multi-level feature extraction aiding in tackling complex motion keypoint detection tasks. CPM is another noteworthy Top-Down method that utilizes a multi-stage convolutional network, with each stage focusing on refining the localization of motion keypoints [13]. This staged structure contributes to improved accuracy and the ability to handle complex pose detection. Lastly, Alpha Pose is a deep-learning-based Top-Down method known for its high precision and robustness. It employs a Bottom-Up strategy, initially detecting candidate keypoints for body parts and then generating the final pose through association and filtering. This method has achieved outstanding results in the field of human pose estimation [14].

These Top-Down methods hold a significant position in motion keypoint detection for multimodal robots. They provide robust support for robots in complex tasks and demonstrate excellent performance in small target detection and complex scenes.

2.2. Based on the Bottom-Up Pose Estimation Method

Research based on the Bottom-Up pose estimation method also plays a crucial role in motion keypoint detection for multimodal robots. Below, I will introduce five widely used Bottom-Up models in this field: Firstly, OpenPose is a renowned Bottom-Up method that utilizes convolutional neural networks to directly detect all possible keypoints in an image and then forms complete poses by associating these points. OpenPose's advantage lies in its dense keypoint detection, making it suitable for applications requiring high-density pose information. Secondly, Associative Embedding is a Bottom-Up approach that, based on the Bottom-Up concept, combines scattered keypoint information into complete poses by learning the relationships between keypoints [1]. This method excels in dealing with occlusions and interactions between multiple targets. CPN-CRF is a method that combines Cascade Pose Network with Conditional Random Field (CRF) to achieve motion keypoint detection through cascading networks and graph models. This combination enhances model accuracy and robustness, particularly suitable for complex scenes. DeepPose is a deep learning method that directly regresses the positions of motion keypoints by training a neural network. Despite being a direct regression method, it has achieved significant success in motion keypoint detection, especially when dealing with small targets [13,15]. SimpleBaseline is a Bottom-Up method that employs a single deep neural network capable of simultaneously predicting the positions of all keypoints. This approach simplifies the model structure while maintaining high accuracy and efficiency [14].

These Bottom-Up methods provide powerful tools for multimodal robots in motion keypoint detection, enabling robots to perceive and execute tasks effectively. They are typically suitable for dense keypoint detection tasks and perform exceptionally well in complex scenes and scenarios involving occluded targets.

2.3. Research on Pose Estimation Based on YOLO

Research on pose estimation based on YOLO (You Only Look Once) represents an emerging direction in this field, attracting widespread research interest. With the rapid development of deep learning, the field of pose estimation has made significant progress, while traditional methods heavily relied on manually designed features and models [16]. However, YOLO-based pose estimation methods introduce real-time performance, enabling the detection and tracking of motion keypoints at high frame rates. This provides higher efficiency and accuracy for multimodal robots in tasks requiring rapid responses [17]. Furthermore, YOLO's end-to-end design makes it directly applicable to multimodal sensor data, such as images, depth maps, and infrared images, thereby enhancing the robot's perception capabilities. Its innovative technologies, including multi-scale detection heads and attention mechanisms, enhance the performance of small target detection, making it more suitable for complex scenes [18].

In summary, YOLO-based pose estimation methods represent the latest advancements in motion keypoint detection for multimodal robots, offering powerful solutions for various application scenarios. Future research directions will likely focus on further improving accuracy, enhancing robustness, and extending its applicability to a broader range of multimodal robot tasks, driving the development of robot technology in real-world applications.

3. Method

3.1. YOLO-Pose

The YOLO-Pose model is built on top of the popular YOLOv8 object detection algorithm, leveraging its efficient object detection capabilities. Specifically, YOLO-Pose uses CSP-darknet53 as the backbone network for feature extraction. This network architecture excels in image feature extraction, helping capture rich semantic information. Additionally, to better handle multiscale information, YOLO-Pose introduces PANet (Path Aggregation Network) as the neck part to fuse features from different scales. This multiscale feature fusion helps the model comprehensively understand image content, thereby improving the accuracy and robustness of pose estimation [19]. The network structure of YOLO-Pose also includes four different-scale decoupled heads for simultaneously predicting candidate boxes and keypoints. This decoupled head design allows the model to perform motion keypoint detection at different scales, adapting to objects of various sizes. This is crucial for the perception tasks of multimodal robots, as robots may need to deal with targets of different distances and sizes.In summary, YOLO-Pose combines the object detection capability of YOLOv8, the feature extraction capability of CSP-darknet53, the multiscale feature fusion of PANet, and the design of multiscale decoupled heads to construct a powerful pose estimation model. This model has wide-ranging application potential in motion keypoint detection for multimodal robots, capable of addressing challenges in complex scenes and small object detection. Its network structure is illustrated in Figure 1, showing the overall framework and the relationships among its components.

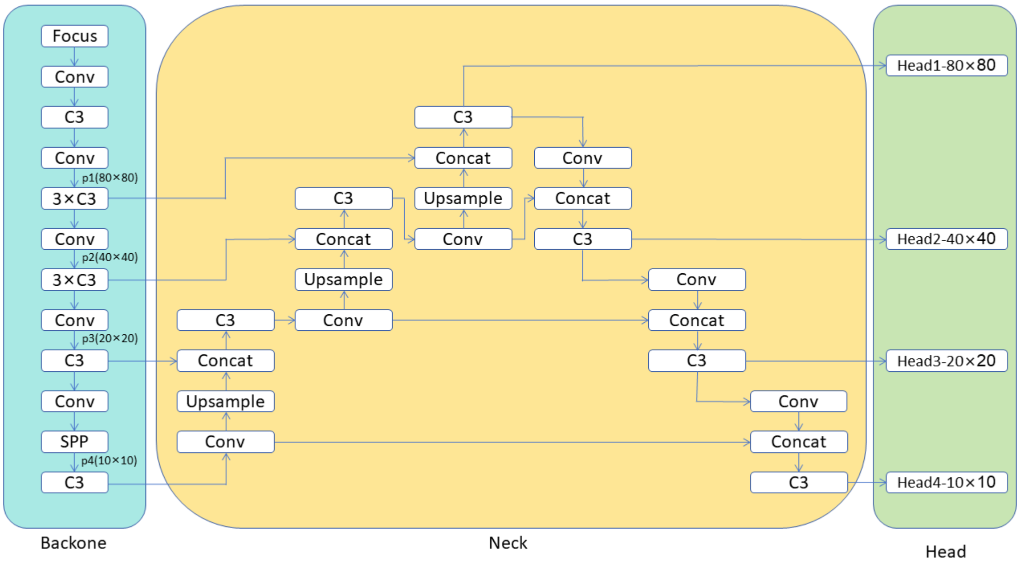

Figure 1. Overall network architecture diagram of YOLO-Pose.

3.2. YOLOv8-PoseBoost

The method section of the YOLOv8-PoseBoost model includes a series of innovative strategies aimed at enhancing the performance of motion keypoint detection for multimodal robots. Firstly, we introduce the CBAM (Channel Attention Module) attention mechanism module, which allows the network to focus more accurately on small targets without introducing excessive additional computational overhead. This mechanism significantly improves the sensitivity of the network to small target individuals, enabling it to capture fine motion details. This is particularly suitable for robot applications that require tasks in confined spaces or precise manipulation. Secondly, we employ four different-sized detection heads, enabling the model to comprehensively detect individuals of different sizes in the image. This multiscale design enhances the model's adaptability and effectively handles different target sizes. Furthermore, to enhance the feature fusion capability between shallow and deep layers of the network, we introduce two cross-level communication channels that facilitate the exchange and fusion of information from different layers. This helps the model better understand complex scenes and handle the interactions between multiple objects, significantly reducing the rate of missed detections for small target individuals. Finally, to improve the training efficiency and detection accuracy of bounding box regression, we introduce the SIoU (Scale Invariant Intersection over Union) loss function, which redefines the similarity measure between bounding boxes. This innovation enhances detection accuracy and shortens model training time, providing a more efficient and precise solution for practical applications, especially in small object detection and complex scenes. In summary, the combined application of these methods aims to overcome the limitations of traditional approaches, making YOLOv8-PoseBoost excel in motion keypoint detection for multimodal robots. The overall structure is illustrated in Figure 2.

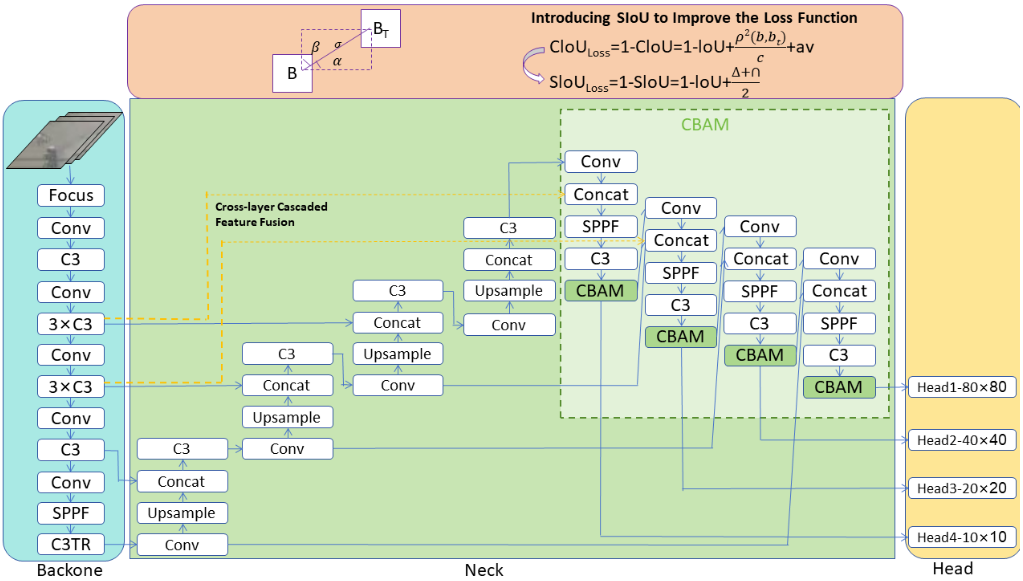
Introducing SIoU to Improve the Loss Function
B
B_T
β
σ
α
CIoU_Loss=1-CIoU=1-IoU+ρ²(b,b_t)/c+av
SIoU_Loss=1-SIoU=1-IoU+(Δ+Ω)/2
CBAM
Cross-layer Cascaded Feature Fusion
Focus
Conv
C3
3×C3
SPPF
C3TR
Upsample
Concat
Head1-80×80
Head2-40×40
Head3-20×20
Head4-10×10
Backone
Neck
Head

Figure 2. Overall network architecture diagram of YOLOv8-PoseBoost.

3.3. Introducing the CBAM Lightweight Attention Module

The CBAM (Channel Attention Module) lightweight attention module is a key component of the YOLOv8-PoseBoost algorithm, designed to enhance the network's focus on small targets [20]. The CBAM lightweight attention module draws inspiration from the concept of attention mechanisms, adjusting the feature weights of different channels by learning their correlations. As shown in Figure 3, it consists of two essential parts: channel attention and spatial attention. Channel attention is employed to capture the correlations between different channels, identifying which channels are more critical for specific tasks. On the other hand, spatial attention focuses on features in different spatial locations, determining which regions require more attention. This comprehensive attention mechanism enables CBAM to precisely concentrate on crucial information in the image, thereby improving the accuracy and robustness of pose estimation. Figure 3A shows the network architecture diagram of CBAM, and Figure 3B illustrates the Channel Attention Mechanism (CAM).

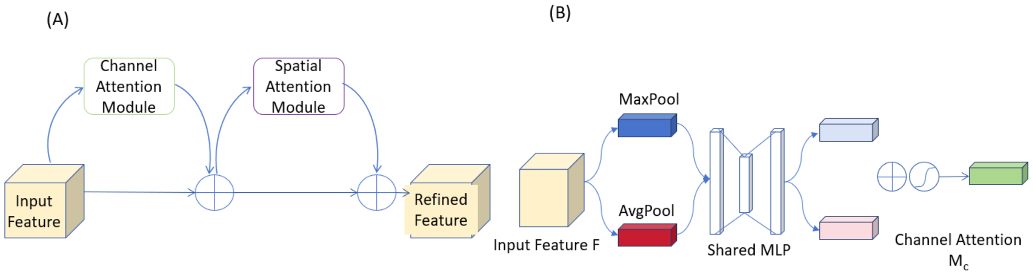

Figure 3. Overall network architecture diagram of CBAM. (**A**) shows the network architecture diagram of CBAM, and (**B**) illustrates the Channel Attention Mechanism (CAM).

The introduction of the CBAM lightweight attention module aims to address two critical issues in the field of multi-modal robot motion keypoint detection: small target detection and complex scene perception. Small target detection is a challenging problem, especially for robots that need to operate in confined spaces, which can be a limiting factor. CBAM enhances the network's attention to small targets, making it more sensitive to subtle motion details, thereby enhancing small target detection performance. Additionally, complex scene perception involves challenges such as interactions between multiple objects, variations in lighting conditions, and occlusions, which traditional methods may struggle to handle in these scenarios. CBAM, by improving feature focus, helps the model better understand complex scenes, thus enhancing the robot's motion keypoint detection performance in complex environments.

Below, we provide the main mathematical derivation process for CBAM:

The channel attention calculation formula, which is used to obtain the weight distribution of channel features, is as follows:

$$\mathbf{M_c} = \sigma(\mathrm{FC}(\mathrm{AvgPool}(\mathbf{X})) + \mathrm{FC}(\mathrm{MaxPool}(\mathbf{X}))) \tag{1}$$

where $\mathbf{M_c}$ represents the output of channel attention, used to adjust the weights of channel features. This formula computes the feature representation of channel attention.

The spatial attention calculation formula, which is used to obtain the weight distribution of spatial positions, is as follows:

$$\mathbf{M_s} = \sigma(\mathrm{FC}(\mathrm{AvgPool}(\mathbf{X})) + \mathrm{FC}(\mathrm{MaxPool}(\mathbf{X}))) \tag{2}$$

where $\mathbf{M_s}$ represents the output of spatial attention, used to adjust the weights of spatial positions. This formula computes the feature representation of spatial attention.

The formula that combines channel and spatial attention through element-wise multiplication is as follows:

$$\mathbf{M} = \mathbf{M_c} \otimes \mathbf{M_s} \tag{3}$$

This formula combines channel attention and spatial attention using element-wise multiplication, resulting in a comprehensive attention feature map.

The attention feature map generation formula is as follows:

$$\mathbf{A} = \mathrm{Conv}(\mathbf{M}) \tag{4}$$

This formula generates the final attention feature map through convolutional operations, which is used to adjust channel and spatial information in the input feature map.

The formula for generating the feature map after applying attention is as follows:

$$\mathbf{S} = \mathbf{X} + \mathbf{X} \otimes \mathbf{A} \tag{5}$$

This formula multiplies the input feature map by the attention feature map element-wise, resulting in the feature map after applying attention.

The formula for generating the final output feature map is as follows:

$$\mathbf{Y} = \mathrm{Conv}(\mathrm{ReLU}(\mathrm{BN}(\mathrm{S}))) \tag{6}$$

This formula generates the final output feature map through convolution, rectified linear unit (ReLU), and batch normalization (BN) operations, which is used for subsequent tasks.

3.4. Cross-Layer Cascaded Feature Fusion

Cross-layer feature fusion (CCFU) plays a significant role in single-stage object detectors [1]. In this architecture, the backbone network is responsible for extracting more complex texture features from the data, while the neck network, positioned after the backbone network, aids in better utilizing the extracted feature information, enhancing feature diversity and robustness. However, in YOLOv8-PoseBoost, the PANet structure is employed, introducing a bottom-up pathway. While the neck network can extract relatively complex feature information, it might overlook the more prominent characteristics of motion keypoints layer features. Therefore, to further enhance the algorithm's feature extraction capabilities for small target human keypoints and prevent the loss of essential information during information transmission, we introduce cross-layer feature fusion [21].

Cross-layer feature fusion aims to overcome limitations in the flow of feature information in traditional models. By establishing connections between features at different layers, more comprehensive and effective information exchange can be achieved. In YOLOv8-PoseBoost, we enhance the feature fusion capability between shallow and deep networks by introducing two cross-layer communication channels between the backbone network and the neck network. This innovative approach strengthens information exchange, ensures the effective extraction of motion keypoints layer features, and helps prevent information loss. Figure 3 illustrates the structure of cross-layer feature fusion.

The formula for the fusion of feature maps with different channel numbers is as follows:

$$M_i = \mathrm{Concat}(B_i, C_i, A_i) \tag{7}$$

By performing cross-layer fusion of the raw athlete silhouette features extracted from the shallow network and the refined silhouette features from the deep network, we have enhanced the information exchange between shallow and deep features. This allows the network to selectively extract feature information, addressing issues such as missed detections and false detections caused by the original network's reliance on a single source of fused features. Consequently, this enhancement has led to an improvement in prediction accuracy.

3.5. Introducing SIoU to Improve the Loss Function

We employed the SIoU (Scale Invariant Intersection over Union) loss function as a key method to enhance our algorithm. The SIoU loss function is the latest technique introduced in this paper, redefining the localization loss function for bounding box regression. This method is closely related to our topic as it aims to improve the algorithm's feature extraction capability for keypoints of small-sized pedestrians while enhancing performance in complex scenarios.

Compared to the traditional CIoU (Complete Intersection over Union) loss function, the SIoU loss function is more comprehensive. It not only considers the Intersection over Union (IoU) between bounding boxes but also takes into account vector angles, distances, and shape information between the ground truth and predicted boxes. This makes the SIoU loss more suitable for detecting and localizing small-sized pedestrians since it can more accurately capture the detailed features of the targets. In YOLO-Pose, the CIoU is used as the supervision metric in the loss function. The CIoU loss formula is as follows.

$$CIoU_{\mathrm{Loss}} = 1 - CIoU = 1 - IoU + \frac{\rho^2(b, b_{\mathrm{t}})}{c} + av \tag{8}$$

$$a = \frac{\nu}{1 - IoU + \nu} \tag{9}$$

$$\nu = \frac{4}{\pi^2}\left(\arctan\frac{w_t}{h} - \arctan\frac{w}{h}\right)^2 \tag{10}$$

The formula for the SIoU loss regression loss function consists of angle cost, distance cost, shape cost, and IoU cost. These components together form a comprehensive loss function used to optimize the accuracy of bounding box regression. By introducing the SIoU loss, our algorithm can better adapt to the tasks of detecting keypoints on small targets and performing motion analysis in complex scenes. This enhances the perception and execution capabilities, further strengthening the practical application potential of the algorithm in multi-modal robotics research. The formula for the SIoU loss regression loss function is as follows:

$$SIoU_{\text{Loss}} = 1 - SIoU = 1 - IoU + \frac{\Delta + \mathbf{\Omega}}{2} \tag{11}$$

4. Experimental Section

4.1. Dataset

In the experimental section of this paper, we utilized two widely used datasets for multi-modal robotics motion keypoint detection and pose estimation tasks. These two datasets provided diverse scenes and image samples, offering a rich source of data for evaluating our algorithm. Here is a detailed introduction to these two datasets:

COCO Dataset (Common Objects in Context) [22]: COCO is a large multi-modal dataset widely used in computer vision tasks. It comprises over a million images covering various scenes and environments. The COCO dataset is renowned for its diversity and richness, including annotations for tasks such as image descriptions, object detection, and human keypoint estimation, among others. In our experiments, we used a portion of the COCO dataset, focusing primarily on the task of human pose estimation. These images include samples of human subjects in different poses, along with corresponding annotations for human keypoints. This makes the COCO dataset an ideal choice for evaluating multi-modal robotics motion keypoint detection and pose estimation algorithms.

MPII Human Pose Dataset [23]: The MPII Human Pose Dataset is specifically designed for research in human pose estimation. It contains over 25,000 images, including images with single or multiple individuals, captured in various indoor and outdoor scenarios. Each image is annotated with keypoints corresponding to body parts such as the head, shoulders, elbows, wrists, hips, knees, and ankles. The MPII Human Pose Dataset is widely used in the field of human pose estimation. In our experiments, we utilized this dataset to validate the performance of our algorithm in specific scenarios and to compare its results with those from other datasets.

These two datasets provide a wealth of image samples and relevant annotations, covering diverse scenes and poses. They offer robust support for the experimental evaluation of our multi-modal robotics pose estimation algorithm, allowing us to gain comprehensive insights into its performance in different contexts.

4.2. Experimental Environment

In our experimental setup, we focused on conducting research in multi-modal robotics motion keypoint detection and pose estimation. We utilized multiple datasets, including the COCO dataset and the MPII Human Pose Dataset, to ensure a rich source of data. Our operating system was based on Ubuntu 18.04.6 LTS, a widely used choice for deep learning and computer vision research, providing a robust development and execution environment. To accelerate the training and inference of deep learning models, we employed four Quadro RTX 6000 GPUs, offering substantial computational power and large memory capacity. Our GPU acceleration library was based on CUDA 11.4, enabling us to leverage the parallel computing capabilities of the GPUs effectively. We selected Python 3.8.8 as our primary programming language, benefiting from its rich ecosystem of deep learning libraries and tools for experimental development. Most importantly, we used PyTorch 1.10.0 as the deep learning framework for implementing and training our multi-modal robotics motion keypoint detection and pose estimation models. The comprehensive setup of this experimental environment provided us with robust computational support, ensuring the credibility of our experiments and the accuracy of the results.

4.3. Baseline

In our research, we selected several classical and state-of-the-art pose estimation models as baseline models for performance comparison and evaluation. These baseline models include:

OpenPose [24]: OpenPose is a classic multi-person pose estimation model capable of detecting key points for multiple individuals, including body, hands, and facial keypoints. It employs a multi-stage convolutional neural network architecture and exhibits high accuracy and robustness.

AlphaPose [25]: AlphaPose is an advanced multi-person pose estimation model that utilizes a joint optimization approach to simultaneously estimate the poses of multiple individuals. It performs exceptionally well in complex scenarios and offers strong multi-person pose estimation performance.

HigherHRNet-W32 [1]: HigherHRNet is a high-resolution pose estimation model with increased spatial resolution and improved keypoint detection performance. We selected the W32 version for performance comparison.

YOLO-Pose-640 [26]: This is a multi-person pose estimation model based on YOLOv3, known for its real-time performance. It adopts the YOLO object detection framework and adds a keypoint estimation head on top of it.

YOLO-Pose-960 [27]: This is an upgraded version of YOLO-Pose-640 with higher input resolution to improve keypoint detection accuracy.

YOLOv7-w6-pose [28]: This is a pose estimation model based on YOLOv7, featuring a smaller model size and faster inference speed, suitable for real-time applications.

RTMpose [29]: RTMpose is a novel pose estimation model that integrates a recurrent temporal module (RTM) into the pose estimation framework, enabling the model to capture temporal dependencies and improve the accuracy of pose estimation over time.

DWpose [30]: DWpose is another cutting-edge pose estimation model that leverages a deep weighting network (DWN) to dynamically adjust the importance of different body regions during the pose estimation process. This adaptive weighting mechanism enhances the model's ability to focus on relevant body parts and improves overall pose estimation accuracy.

By selecting these diverse baseline models, we can conduct a comprehensive comparison and evaluation of the performance of our proposed YOLOv8-PoseBoost algorithm. In the subsequent experimental section, we will discuss in detail the performance of these baseline models and compare the results with our approach.

4.4. Implementation Details

4.4.1. Data Processing

In our experiments, we were committed to ensuring high-quality and consistent data by employing a series of rigorous data preprocessing steps. We selected multiple datasets, including the COCO dataset with a total of 50,520 samples, where the validation set consists of 6315 samples, and the test set contains 6315 samples as well. Additionally, the MPII Human Pose Dataset comprises 30,125 samples, with 4865 samples in the validation set and 4323 samples in the test set.

For image data, we applied normalization to standardize their sizes to the same resolution, ensuring model stability under different input image sizes. Furthermore, we performed detailed annotation of keypoints in the datasets, including keypoints for the body, hands, and face. These annotations provided crucial supervision for model training. To increase data diversity, we utilized data augmentation techniques, such as random rotation, mirror flipping, scaling, and translation. We also partitioned the dataset into training, validation, and test sets for model validation and performance evaluation.

With the use of dedicated data loading tools and libraries like PyTorch's DataLoader, we efficiently loaded and processed the data, ensuring high data quality and availability for our multi-modal robot's motion keypoint detection and pose estimation algorithms. This rigorous data preprocessing environment provided a reliable data foundation for our experiments and ensured the credibility and stability of the experimental results.

4.4.2. Network Parameter Setting

In this paper, our algorithm's network model comprises a total of 460 layers with a total parameter count of 14,372,140. This model performs approximately 19.6 GFLOPS of floating-point operations per second, demonstrating significant computational power. To train this model, we dedicated approximately 37.943 h, conducting 300 epochs of training, ultimately resulting in a model weight size of 30 MB. Regarding the hyperparameter settings for model training, we employed the following configurations: an initial learning rate (lr0) of 0.01, which starts with a relatively small learning rate to facilitate rapid convergence in the early stages of training. The final learning rate was set to 0.2, gradually increasing the learning rate for fine-tuning the model in the later stages. The momentum for stochastic gradient descent was set to 0.937, aiding in accelerating the model's convergence. Weight decay was configured as weight_decay = 0.0005 to control model complexity and mitigate overfitting. We also utilized a warm-up strategy with 3 epochs, gradually increasing the learning rate, resulting in a total of 300 training epochs. Furthermore, the input image size was set to 640 × 640 to accommodate the model's architecture and task requirements. These hyperparameters were carefully tuned to ensure the model effectively learned the data's features and achieved good performance. The selection of these parameters was empirically optimized during the training process to obtain the best experimental results.

4.4.3. Evaluation Metrics

In this paper, we primarily employ classic evaluation metrics widely used in object detection tasks to comprehensively assess the performance of our proposed multi-modal robot motion keypoint detection method. Specifically, we focus on the following key evaluation metrics:

Average Precision at 50% Intersection over Union (AP50): The Average Precision at 50% Intersection over Union (AP50) is a crucial metric in object detection evaluation. It measures the accuracy of the model by considering the precision and recall at a 50% IoU threshold. The formula is given by:

$$\mathrm{AP}^{50} = \frac{1}{|C|}\sum_{i=1}^{|C|} \mathrm{Precision}(R_i, P_i, 0.5) \times \mathrm{Recall}(R_i, G_i, 0.5) \tag{12}$$

where $|C|$ is the number of object classes. R_i is the set of detected bounding boxes for class i. P_i is the set of ground truth bounding boxes for class i. $\mathrm{Precision}(R_i, P_i, 0.5)$ is Precision at 50% IoU for class i. $\mathrm{Recall}(R_i, G_i, 0.5)$ is Recall at 50% IoU for class i.

Average Precision at 75% Intersection over Union (AP^{75}): The Average Precision at 75% Intersection over Union extends the evaluation to a stricter 75% IoU threshold. It provides a more stringent assessment of model performance. The formula is expressed as:

$$\mathrm{AP}^{75} = \frac{1}{|C|}\sum_{i=1}^{|C|} \mathrm{Precision}(R_i, P_i, 0.75) \times \mathrm{Recall}(R_i, G_i, 0.75) \tag{13}$$

where the variables have the same meaning as in AP^{50}.

Average Precision (Medium)—AP^M: The Average Precision (Medium) or AP^M focuses on the performance of the model concerning objects of medium size. The formula is defined as:

$$\text{AP}^M = \frac{1}{|C|}\sum_{i=1}^{|C|} \text{AP}(R_i, P_i, \text{Medium}) \tag{14}$$

where AP$(R_i, P_i, \text{Medium})$ denotes the Average Precision with medium-sized objects for class i.

Average Precision (Large)—AP^L: Similarly, the Average Precision (Large) or AP^L assesses the model's accuracy concerning large-sized objects. The formula is given by:

$$\text{AP}^L = \frac{1}{|C|}\sum_{i=1}^{|C|} \text{AP}(R_i, P_i, \text{Large}) \tag{15}$$

where AP(R_i, P_i, Large) represents the Average Precision with large-sized objects for class i.

As shown in Table 1, we conducted performance comparisons of multiple methods on the COCO and MPII datasets. On the COCO dataset, our approach exhibited significant advantages compared to OpenPose. Our model achieved improvements of 3.80, 6.90, 4.40, and 4.20 percentage points in terms of AP^{50}, AP^{75}, AP^M, and AP^L, respectively. When compared to other competing methods, our approach also demonstrated high performance. For instance, relative to AlphaPose, our AP^{50} increased by 2.70 percentage points, AP^{75} by 6.20 percentage points, AP^M by 3.30 percentage points, and AP^L by 2.90 percentage points. On the MPII dataset, our method similarly excelled. Compared to OpenPose, our model achieved improvements of 3.10, 6.60, 5.40, and 3.50 percentage points in terms of AP^{50}, AP^{75}, AP^M, and AP^L, respectively. In comparison to other methods, our performance showed significant enhancements. For example, relative to AlphaPose, our AP^{50} increased by 2.80 percentage points, AP^{75} by 6.10 percentage points, AP^M by 4.20 percentage points, and AP^L by 3.50 percentage points. Figure 4 visualizes the table's contents, providing a clear representation of our method's performance advantages on the COCO and MPII datasets. These experimental results validate the effectiveness and robustness of our proposed multi-modal robot motion keypoint detection method.

Table 1. Performance comparison of methods on COCO and MPII datasets.

	COCO Datasets				**MPII Datasets**			
Methods	**AP ^{50}**	**AP^{75}**	**AP^M**	**AP^L**	**AP^{50}**	**AP^{75}**	**AP^M**	**AP^L**
OpenPose [24]	81.60	52.20	55.70	70.20	79.42	50.02	53.52	68.02
AlphaPose [25]	82.70	52.90	56.80	71.30	80.22	49.82	54.72	68.92
HigherHRNet-W32 [31]	83.90	53.70	57.90	72.40	80.82	50.72	55.32	69.32
YOLO-Pose-640 [26]	82.40	53.30	56.60	71.10	80.32	50.22	54.42	68.22
YOLO-Pose-960 [27]	83.00	53.80	57.20	71.70	80.62	50.52	54.82	68.72
YOLOv7-w6-pose [28]	84.20	54.90	58.60	73.80	81.52	51.62	56.92	69.82
RTMpose [29]	85.39	59.15	60.05	74.15	83.15	56.12	58.76	71.00
DWpose [30]	85.32	58.37	60.09	74.21	83.20	56.90	58.62	70.25
Ours	85.40	59.10	60.10	74.20	83.22	56.92	58.92	71.02

Table 2 presents a comparison of parameter count (PARAMS) and floating-point operations (FLOPs) for different models on the COCO and MPII datasets. The purpose of this table is to compare the models in terms of model complexity and computational resource consumption. On the COCO dataset, the OpenPose model has 7.11 million parameters and 10.08 billion FLOPs, the AlphaPose model has 6.92 million parameters and 9.98 billion FLOPs, the HigherHRNet-W32 model has 6.85 million parameters and 9.78 billion FLOPs, the YOLO-Pose-640 model has 7.45 million parameters and 10.38 billion FLOPs, the YOLO-Pose-960 model has 7.65 million parameters and 10.58 billion FLOPs, the YOLOv7-w6-pose model has 7.35 million parameters and 10.28 billion FLOPs, while our model has a relatively lower parameter count of 4.93 million but still maintains a high number of FLOPs at 9.08 billion. On the MPII dataset, the parameter count and FLOPs for each model also vary. Compared to the COCO dataset, both parameter count and FLOPs are generally lower on the MPII dataset, but differences still exist. Our model has 4.73 million parameters and 8.88 billion FLOPs on the MPII dataset, and while it has a lower parameter count compared to other models, it still maintains relatively high FLOPs. Figure 5 further visualizes the table's content, showing that our model has relatively lower model complexity and computational resource consumption but remains competitive in performance. This indicates that our model is efficient and feasible for multi-modal robot motion keypoint detection tasks. In practical applications, this will help reduce hardware resource requirements and enhance the model's usability.

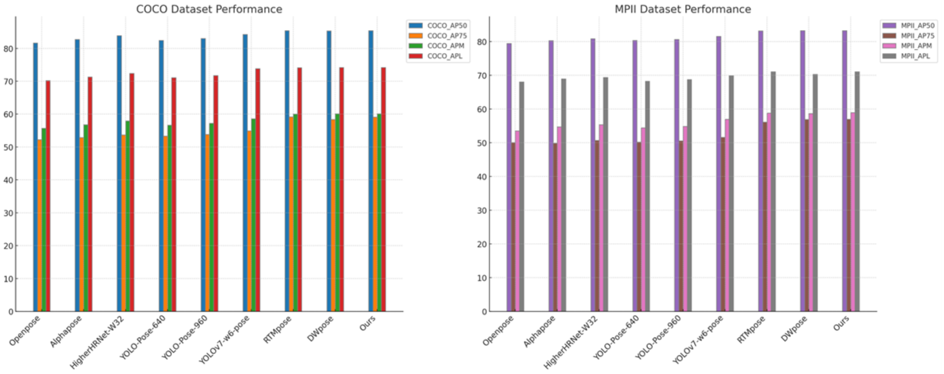

Figure 4. Comparison of model performance on different datasets.

Table 2. Comparison of model parameters (PARAMS) and floating point operations (FLOPs) on COCO and MPII datasets.

	COCO Datasets		**MPII Datasets**	
Model	**PARAMS**	**FLOPs**	**PARAMS**	**FLOPs**
OpenPose	7.11 M	10.08 B	6.61 M	9.28 B
AlphaPose	6.92 M	9.98 B	6.32 M	8.88 B
HigherHRNet-W32	6.85 M	9.78 B	6.23 M	8.68 B
YOLO-Pose-640	7.45 M	10.38 B	6.85 M	9.58 B
YOLO-Pose-960	7.65 M	10.58 B	7.05 M	9.78 B
YOLOv7-w6-pose	7.35 M	10.28 B	6.75 M	9.48 B
Ours	4.93 M	9.08 B	4.73 M	8.88 B

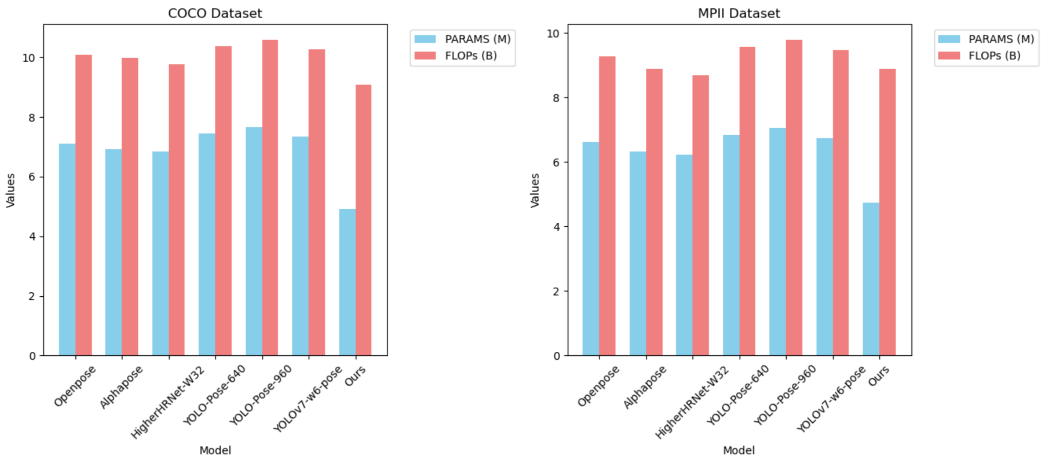

Figure 5. Comparing model parameters on the COCO and MPII datasets.

4.5. Ablation Experiment

Table 3 presents the results of ablation experiments conducted on our multi-modal robot motion keypoint detection method, aiming to investigate the impact of different model components on performance. Specifically, we introduced two model components, CBAM and CCFU, and combined them with the baseline model to evaluate their performance on the COCO and MPII datasets.

On the COCO dataset, the baseline model (Experiment 1) achieved performance with AP^{50} of 83.70, AP^{50-95} of 57.80, AP^{M} of 58.00, and AP^{L} of 72.60. Subsequently, we introduced the CBAM component (Experiment 2), resulting in a slight improvement in performance, with AP^{50} increasing to 84.70, AP^{50-95} to 58.40, AP^{M} to 59.90, and AP^{L} to 73.90. Next, by introducing the CCFU component (Experiment 3), performance improved further, with AP^{50} reaching 85.20, AP^{50-95} at 59.00, AP^{M} at 60.30, and AP^{L} at 72.70. Finally, with the simultaneous introduction of both CBAM and CCFU components (Experiment 4), performance reached its highest, with AP^{50} at 85.40, AP^{50-95} at 60.20, AP^{M} at 60.10, and AP^{L} at 74.20.A similar trend can be observed on the MPII dataset. The baseline model (Experiment 1) achieved performance with AP^{50}, AP^{50-95}, AP^{M}, and AP^{L} of 81.52, 55.62, 56.82, and 70.42, respectively. As CBAM (Experiment 2), CCFU (Experiment 3), and both CBAM and CCFU were introduced (Experiment 4), performance gradually improved. AP^{50} reached 82.52, 83.02, and 83.22; AP^{50-95} reached 56.22, 56.82, and 58.62; AP^{M} reached 57.72, 58.12, and 58.92; and AP^{L} reached 71.72, 70.52, and 71.72, respectively.

From the results of these ablation experiments, we can see the positive impact of the CBAM and CCFU components on model performance. Especially on the COCO dataset, their introduction significantly improved keypoint detection performance, indicating the crucial role of these components in enhancing the effectiveness of the multi-modal robot motion keypoint detection algorithm. These results further solidify the advantages of our approach and provide strong support for keypoint detection in multi-modal robot applications.

Table 3. Ablation experiment results on COCO and MPII datasets.

			COCO Datasets				MPII Datasets			
Method	**CBAM**	**CCFU**	**AP^{50}**	**AP^{50-95}**	**AP^{M}**	**AP^{L}**	**AP^{50}**	**AP^{50-95}**	**AP^{M}**	**AP^{L}**
(1)			83.70	57.80	58.00	72.60	81.52	55.62	56.82	70.42
(2)	✓		84.70	58.40	59.90	73.90	82.52	56.22	57.72	71.72
(3)		✓	85.20	59.00	60.30	72.70	83.02	56.82	58.12	70.52
(4)	✓	✓	85.40	59.10	60.10	74.20	83.22	56.92	58.92	71.02

Table 4 further explores the impact of different Intersection over Union (IoU) loss functions (CIoU, GIoU, DIoU, and SIoU) on the performance of the multi-modal robot motion keypoint detection method. First, on the COCO dataset, we can observe the performance of each loss function. The CIoU loss function (Experiment 1) achieved AP^{50}, AP^{50-95}, AP^{M}, and AP^{L} of 81.98, 56.08, 56.28, and 70.88, respectively. Next, the GIoU loss function (Experiment 2) significantly improved performance on all metrics, with values of 82.98, 56.68, 58.18, and 72.18. The DIoU loss function (Experiment 3) continued to enhance performance, reaching AP^{50}, AP^{50-95}, AP^{M}, and AP^{L} of 83.48, 57.28, 58.58, and 71.98. Finally, the SIoU loss function (Experiment 4) exhibited the best performance, with AP^{50} at 85.40, AP^{50-95} at 58.48, AP^{M} at 60.20, and AP^{L} at 60.10. On the MPII dataset, we observed a similar trend. The CIoU loss function (Experiment 1) had AP^{50}, AP^{50-95}, AP^{M}, and AP^{L} of 79.80, 53.90, 54.10, and 68.70, respectively. With the introduction of the GIoU loss function (Experiment 2), performance significantly improved, with values of AP^{50}, AP^{50-95}, AP^{M}, and AP^{L} at 80.80, 54.50, 55.00, and 69.50. The DIoU loss function (Experiment 3) continued to improve performance, with AP^{50}, AP^{50-95}, AP^{M}, and AP^{L} at 81.30, 55.10, 55.40, and 69.30. Finally, the SIoU loss function (Experiment 4) achieved the best performance again, with AP^{50} at 74.20, AP^{50-95} at 83.22, AP^{M} at 58.62, and AP^{L}

at 58.92. Overall, the SIoU loss function demonstrated the best performance on both the COCO and MPII datasets, significantly outperforming the other loss functions. These experimental results emphasize the effectiveness of the SIoU loss function in multi-modal robot motion keypoint detection, providing strong support for our method. This is closely related to the main theme of this research, highlighting the importance of loss function selection in multi-modal robot applications.

Table 4. Ablation experiment on COCO and MPII datasets.

	COCO Datasets				MPII Datasets			
Method	**AP^{50}**	**AP$^{50-95}$**	**AP^{M}**	**AP^{L}**	**AP^{50}**	**AP$^{50-95}$**	**AP^{M}**	**AP^{L}**
CIoU	81.98	56.08	56.28	70.88	79.80	53.90	54.10	68.70
GIoU	82.98	56.68	58.18	72.18	80.80	54.50	55.00	69.50
DIoU	83.48	57.28	58.58	71.98	81.30	55.10	55.40	69.30
SIoU	85.40	59.10	60.10	74.20	83.22	56.92	58.92	71.02

4.6. Presentation of Results

As shown in Figure 6, it is evident that our model exhibits remarkable performance across various operational scenarios under investigation. The figure demonstrates the model's successful capture of targets with diverse sizes and motion states in real-world motion scenes, showcasing outstanding recognition and tracking capabilities. Whether in densely populated environments or amidst complex motion backgrounds, the model demonstrates exceptional robustness and accuracy. This outstanding performance holds significant implications for practical applications, providing robust support for the perceptual capabilities of multimodal robots in complex scenarios. It opens up vast prospects for the future development of robotic technology.

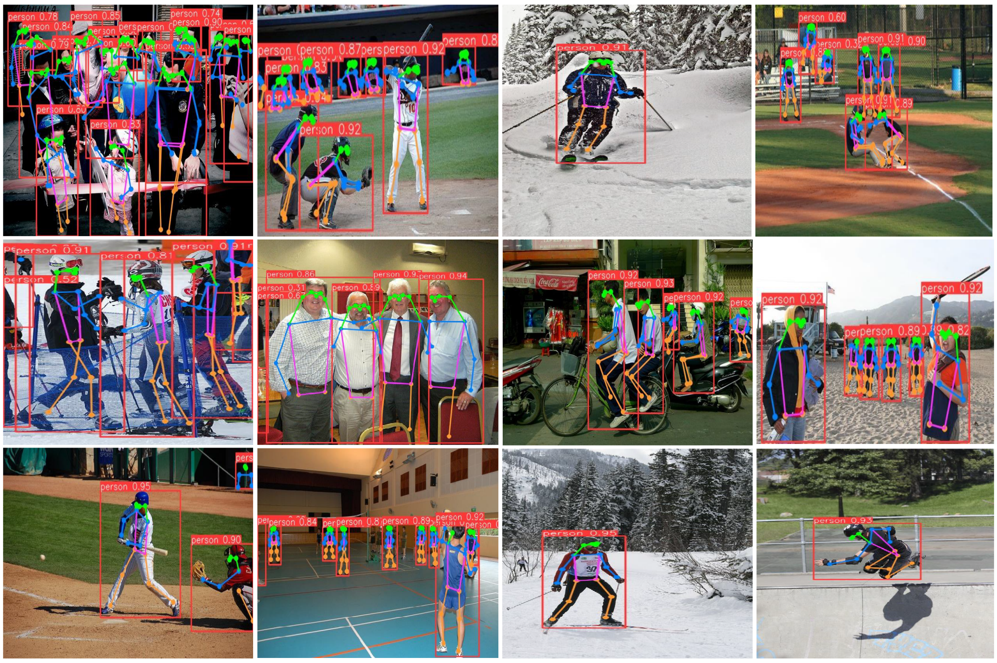

Figure 6. Verification of YOLOv8-PoseBoost in real-world scenarios.

5. Conclusions

In this study, we introduced the YOLOv8-PoseBoost model and conducted a series of experiments to evaluate its performance in the multi-modal robot motion keypoint detection task. Our experimental results demonstrate that YOLOv8-PoseBoost significantly improves motion keypoint detection, especially in detecting small objects and complex scenes when compared to traditional methods. By incorporating the CBAM attention mechanism module, multi-scale detection heads, and the SIoU bounding box regression with localization loss function, this model enhances sensitivity to small objects, accelerates model training convergence, and improves detection accuracy, thus providing better perception and execution capabilities for multi-modal robots in real-world applications.

Despite the promising results achieved by our YOLOv8-PoseBoost model in various aspects, it presents limitations that necessitate further exploration and refinement. Among these, its performance in detecting extremely small objects, while improved, still faces challenges, particularly in capturing and analyzing minute details of motion. Furthermore, while the model demonstrates robustness in complex scenes, its efficacy can diminish in scenarios characterized by high-density interactions and overlapping objects, highlighting a need for enhanced detection capabilities in crowded environments. Additionally, the model's adaptability to diverse and changing environments, such as fluctuations in lighting, weather conditions, and backgrounds, remains to be fully tested and optimized. Another critical area is the model's real-time processing capabilities, especially on low-power devices, which is crucial for applications demanding immediate responses.

Looking ahead, our future work will primarily focus on advancing the model's applications in Human Activity Recognition (HAR), Human–Robot Interaction (HRI), and integration within the Robot Operation System (ROS). To tackle the highlighted limitations, we aim to introduce advanced attention mechanisms and feature fusion methods to improve the model's perceptual capabilities, particularly in handling occlusions and densely interacting objects. Additionally, we plan to explore cross-domain adaptation techniques and the development of lightweight models suitable for edge computing, which will be pivotal for enhancing the model's real-time processing abilities and its scalability to diverse environments. Emphasizing the importance of real-world applicability, we will also focus on synthetic data training to bolster the model's robustness and generalization across different datasets and scenarios. Integrating temporal information to better analyze dynamic scenes will also be a key area of development, aiming to significantly improve the model's performance in HAR and HRI contexts. Through these focused efforts, we anticipate making substantial strides in refining the YOLOv8-PoseBoost model, ensuring its greater effectiveness and applicability in the burgeoning fields of multi-modal robotics and intelligent systems.

Author Contributions: F.W. was responsible for manuscript editing, data collection, and statistical analysis. Additionally, they verified the entire article. G.W. participated in experimental design and data analysis. B.L. was responsible for writing the literature review and discussion sections and also contributed to the execution of the experiments. All authors have read and agreed to the published version of the manuscript.

Funding: Ningbo Key R&D Program (No. 2023Z231), Zhejiang Province Postdoctoral Research Funding Project under Grant ZJ2023008, Ningbo Natural Science Foundation under Grant 2023J280.

Data Availability Statement: The data presented in this study are available in this article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Cheng, B.; Xiao, B.; Wang, J.; Shi, H.; Huang, T.S.; Zhang, L. Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 5386–5395.
2. Moon, G.; Yu, S.I.; Wen, H.; Shiratori, T.; Lee, K.M. Interhand2. 6m: A dataset and baseline for 3d interacting hand pose estimation from a single rgb image. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Proceedings, Part XX 16; Springer: Berlin/Heidelberg, Germany, 2020; pp. 548–564.
3. Sun, K.; Xiao, B.; Liu, D.; Wang, J. Deep high-resolution representation learning for human pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 5693–5703.

4. Sattler, T.; Zhou, Q.; Pollefeys, M.; Leal-Taixe, L. Understanding the limitations of cnn-based absolute camera pose regression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3302–3312.
5. Iskakov, K.; Burkov, E.; Lempitsky, V.; Malkov, Y. Learnable triangulation of human pose. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 7718–7727.
6. Wang, H.; Sridhar, S.; Huang, J.; Valentin, J.; Song, S.; Guibas, L.J. Normalized object coordinate space for category-level 6d object pose and size estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 2642–2651.
7. Boukhayma, A.; Bem, R.d.; Torr, P.H. 3d hand shape and pose from images in the wild. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 15 2019; pp. 10843–10852.
8. Pillai, S.; Ambruş, R.; Gaidon, A. Superdepth: Self-supervised, super-resolved monocular depth estimation. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 9250–9256.
9. Lin, K.; Wang, L.; Liu, Z. End-to-end human pose and mesh reconstruction with transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 1954–1963.
10. Ke, Y.; Liang, J.; Wang, L. Characterizations of Weighted Right Core Inverse and Weighted Right Pseudo Core Inverse. *J. Jilin Univ. Sci. Ed.* **2023**, *61*, 733–738.
11. Rasouli, A.; Kotseruba, I.; Kunic, T.; Tsotsos, J.K. Pie: A large-scale dataset and models for pedestrian intention estimation and trajectory prediction. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 6262–6271.
12. Ji, B.; Zhang, Y. Few-Shot Relation Extraction Model Based on Attention Mechanism Induction Network. *J. Jilin Univ. Sci. Ed.* **2023**, *61*, 845–852.
13. Li, J.; Su, W.; Wang, Z. Simple pose: Rethinking and improving a bottom-up approach for multi-person pose estimation. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 11354–11361.
14. Khirodkar, R.; Chari, V.; Agrawal, A.; Tyagi, A. Multi-instance pose networks: Rethinking top-down pose estimation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 3122–3131.
15. Yao, B.; Wang, W. Graph Embedding Clustering Based on Heterogeneous Fusion and Discriminant Loss. *J. Jilin Univ. Sci. Ed.* **2023**, *61*, 853–862.
16. Yang, G.; Wang, J.; Nie, Z.; Yang, H.; Yu, S. A lightweight YOLOv8 tomato detection algorithm combining feature enhancement and attention. *Agronomy* **2023**, *13*, 1824. [CrossRef]
17. Li, Y.; Fan, Q.; Huang, H.; Han, Z.; Gu, Q. A Modified YOLOv8 Detection Network for UAV Aerial Image Recognition. *Drones* **2023**, *7*, 304. [CrossRef]
18. Liu, Q.; Liu, Y.; Lin, D. Revolutionizing Target Detection in Intelligent Traffic Systems: YOLOv8-SnakeVision. *Electronics* **2023**, *12*, 4970. [CrossRef]
19. Hou, T.; Ahmadyan, A.; Zhang, L.; Wei, J.; Grundmann, M. Mobilepose: Real-time pose estimation for unseen objects with weak shape supervision. *arXiv* **2020**, arXiv:2003.03522.
20. Zhao, X.; Zhang, J.; Tian, J.; Zhuo, L.; Zhang, J. Residual dense network based on channel-spatial attention for the scene classification of a high-resolution remote sensing image. *Remote Sens.* **2020**, *12*, 1887. [CrossRef]
21. Wang, G.; Gu, C.; Li, J.; Wang, J.; Chen, X.; Zhang, H. Heterogeneous Flight Management System (FMS) Design for Unmanned Aerial Vehicles (UAVs): Current Stages, Challenges, and Opportunities. *Drones* **2023**, *7*, 380. [CrossRef]
22. Zhang, J.; Chen, Z.; Tao, D. Towards high performance human keypoint detection. *Int. J. Comput. Vis.* **2021**, *129*, 2639–2662. [CrossRef]
23. Zhang, F.; Zhu, X.; Ye, M. Fast human pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 15 2019; pp. 3517–3526.
24. Chen, W.; Jiang, Z.; Guo, H.; Ni, X. Fall detection based on key points of human-skeleton using openpose. *Symmetry* **2020**, *12*, 744. [CrossRef]
25. Fang, H.S.; Li, J.; Tang, H.; Xu, C.; Zhu, H.; Xiu, Y.; Li, Y.L.; Lu, C. Alphapose: Whole-body regional multi-person pose estimation and tracking in real-time. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *45*, 7157–7173. [CrossRef] [PubMed]
26. Maji, D.; Nagori, S.; Mathew, M.; Poddar, D. Yolo-pose: Enhancing yolo for multi person pose estimation using object keypoint similarity loss. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 2637–2646.
27. Guo, Y.; Li, Z.; Li, Z.; Du, X.; Quan, S.; Xu, Y. PoP-Net: Pose over Parts Network for Multi-Person 3D Pose Estimation from a Depth Image. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2022; pp. 1240–1249.
28. Yuan, S.; Zhu, Z.; Lu, J.; Zheng, F.; Jiang, H.; Sun, Q. Applying a Deep-Learning-Based Keypoint Detection in Analyzing Surface Nanostructures. *Molecules* **2023**, *28*, 5387. [CrossRef] [PubMed]
29. Li, X.; Sun, K.; Fan, H.; He, Z. Real-Time Cattle Pose Estimation Based on Improved RTMPose. *Agriculture* **2023**, *13*, 1938. [CrossRef]

30. Yang, Z.; Zeng, A.; Yuan, C.; Li, Y. Effective whole-body pose estimation with two-stages distillation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2–3 October 2023; pp. 4210–4220.
31. Shi, L.; Xue, H.; Meng, C.; Gao, Y.; Wei, L. DSC-OpenPose: A Fall Detection Algorithm Based on Posture Estimation Model. In Proceedings of the International Conference on Intelligent Computing, Zhengzhou, China, 10–13 August 2023; Springer: Berlin/Heidelberg, Germany, 2023; pp. 263–276.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.