sensors

MDPI

Article

KSL-POSE: A Real-Time 2D Human Pose Estimation Method Based on Modified YOLOv8-Pose Framework

Tianyi Lu [†], **Ke Cheng** [*,†], **Xuecheng Hua** [†] **and Suning Qin** [†]

School of Computer, Jiangsu University of Science and Technology, Zhenjiang 212100, China; chnlty671@gmail.com (T.L.); huaxuecheng8888@gmail.com (X.H.); 18118814562@163.com (S.Q.)

[*] Correspondence: chengke1972@just.edu.cn

[†] These authors contributed equally to this work.

Abstract: Two-dimensional human pose estimation aims to equip computers with the ability to accurately recognize human keypoints and comprehend their spatial contexts within media content. However, the accuracy of real-time human pose estimation diminishes when processing images with occluded body parts or overlapped individuals. To address these issues, we propose a method based on the YOLO framework. We integrate the convolutional concepts of Kolmogorov–Arnold Networks (KANs) through introducing non-linear activation functions to enhance the feature extraction capabilities of the convolutional kernels. Moreover, to improve the detection of small target keypoints, we integrate the cross-stage partial (CSP) approach and utilize the small object enhance pyramid (SOEP) module for feature integration. We also innovatively incorporate a layered shared convolution with batch normalization detection head (LSCB), consisting of multiple shared convolutional layers and batch normalization layers, to enable cross-stage feature fusion and address the low utilization of model parameters. Given the structure and purpose of the proposed model, we name it KSL-POSE. Compared to the baseline model YOLOv8l-POSE, KSL-POSE achieves significant improvements, increasing the average detection accuracy by 1.5% on the public MS COCO 2017 data set. Furthermore, the model also demonstrates competitive performance on the CrowdPOSE data set, thus validating its generalization ability.

Keywords: human pose estimation; Kolmogorov–Arnold networks; YOLOv8-pose; multi-scale feature fusion

check for updates

Citation: Lu, T.; Cheng, K.; Hua, X.; Qin, S. KSL-POSE: A Real-Time 2D Human Pose Estimation Method Based on Modified YOLOv8-Pose Framework. *Sensors* **2024**, *24*, 6249. https://doi.org/10.3390/s24196249

Academic Editor: Loris Nanni

Received: 15 August 2024
Revised: 23 September 2024
Accepted: 25 September 2024
Published: 26 September 2024

1. Introduction

Human pose estimation is a vital research area in the computer vision field, which is focused on accurately identifying the spatial positions of key human joints in media content. Real-time human pose recognition is essential in various fields, including motion tracking, gesture recognition, motion analysis, and intelligent surveillance systems. It has extensive applications in virtual reality, behavior recognition, sports training, and the film and animation industries [1,2]. Achieving accurate 2D human pose estimation allows for optimization of model architectures and feature learning, thereby enhancing the precision and efficiency of pose estimation, ultimately leading to safer and more efficient services.

There are two types of tasks in the human pose recognition field: single-person pose estimation and multi-person pose estimation. In single-person pose estimation, the model focuses on identifying and regressing the keypoint information of the target person, which includes keypoint categories, the relative positions of keypoints, and confidence levels. In contrast, multi-person pose estimation involves simultaneously detecting and estimating the keypoint information of multiple individuals. In these tasks, various degrees of joint deformation occur during human activities (as illustrated in Figure 1a–c), and body parts may be occluded by other objects or the individuals themselves (as shown in Figure 1d–f). These factors can lead to the pose estimation algorithm either failing to obtain keypoint information or only capturing a small amount, ultimately resulting in poor pose estimation outcomes. Therefore, this research aims to better address the challenges of capturing subtle and complex features, regressing occluded keypoints, and detecting small targets in human pose estimation tasks.

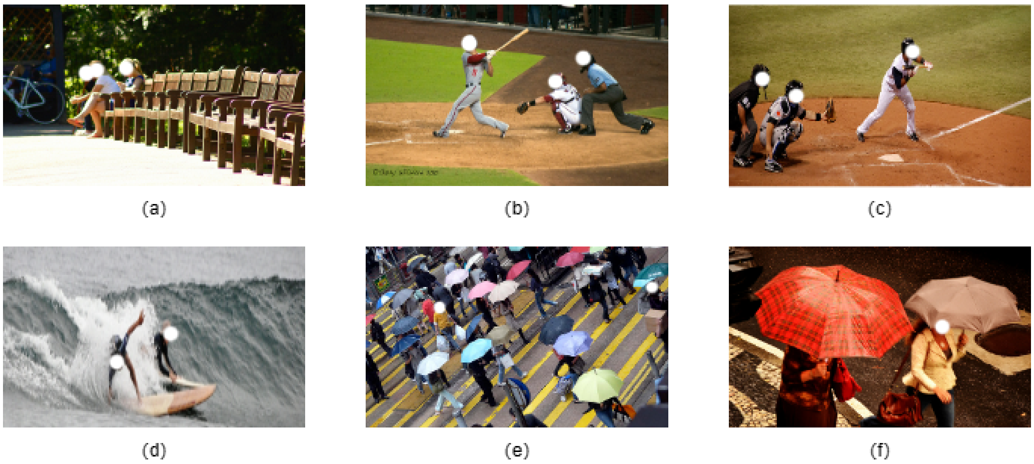

Figure 1. (**a**–**c**) demonstrate varying degrees of joint deformation that occur during human activities; (**d**–**f**) highlight instances where body parts are obscured either by objects or the individual.

In recent years, with the continuous development of Deep Neural Networks (DNNs) [3], Convolutional Neural Networks (CNNs) [4], Recurrent Neural Networks (RNNs) [5], attention mechanisms [6], Transformers [7], and Kolmogorov–Arnold Networks (KANs) [8], researchers have significantly enhanced the regression capability of models for keypoints. They have achieved this through leveraging well-trained network architectures, making structural modifications, and applying post-optimization processes. Moreover, various strategies have been adopted to reduce the inference cost of networks, such as lightweight network design [9,10], spatial pyramid pooling [11], and other techniques. Researchers have also endeavored to integrate human pose estimation with object detection to address a broader range of practical scenarios [12]. For example, Ghiasi et al. have proposed NAS-FPN [13], which uses neural network structure search to discover a new feature pyramid structure in a new scalable search space covering all cross-scale connections and can fuse the input features of any two layers into the output features of one layer.

The YOLO series models [14], which are popular in the field of computer vision, were designed to predict the locations and categories of objects in an image through single-stage propagation of the network, significantly improving their detection speed and efficiency. YOLOv8, an advanced version of this series, has made several improvements in network structure and training strategies, resulting in exceptional performance in both object detection and pose estimation tasks. YOLOv8 introduces new feature extraction modules and more efficient anchor box designs, greatly enhancing the model's detection accuracy and speed. It supports multi-scale fusion and employs spatial pyramid pooling and cross-stage partial networks for feature fusion, effectively capturing fine details in images. Furthermore, YOLOv8 integrates an efficient aggregation network and adopts more effective loss functions and optimization strategies, enabling the model to converge faster during training and achieve considerable detection accuracy. Despite numerous researchers having continuously improved or proposed highly effective human pose estimation models, the field still faces challenges such as partial joint loss due to arbitrary occlusion or complex backgrounds and difficulties in detecting small targets. Given the aforementioned challenges, we propose the KSL-POSE model based on YOLOv8l-POSE and inspired by the YOLO-POSE real-time framework. Through incorporating KAN convolutions, we enhance the model's feature extraction capabilities. Moreover, we introduce the SOEP module and a self-developed detection head, which help to improve its pose estimation performance for small target detection, significantly enhancing the model's performance in complex scenarios.

Overall, the contributions of KSL-POSE are as follows:

- First, KSL-POSE improves the backbone of the YOLOv8l-POSE baseline. We integrate the C2f feature extraction module with the KAN network, introducing non-linear activation functions to enhance the feature extraction capability of the convolutional kernels.
- Second, we introduce the small object enhance pyramid (SOEP) module, leveraging the cross-stage partial (CSP) [15] concept to further optimize and enhance the detail feature extraction performance on small targets.
- Finally, KSL-POSE combines the NAS-FPN concept and introduces a novel shared convolutional detection head. Unlike NAS-FPN, this detection module is composed of multiple shared convolutional layers and batch normalization (BN) layers, where BN is calculated independently, thus improving the utilization efficiency of model parameters and the cross-stage feature fusion capability.

2. Related Work

2.1. Top-Down Methods

Top-down human pose estimation methods involve locating instances through rectangular bounding boxes, then cropping these instances to obtain and detect keypoints within them. Representative algorithms of this type include CPN, HRNet, and Hourglass. Despite having specific applications in human pose estimation, top-down methods are limited in practical use due to their high computational complexity, poor real-time performance, strong dependency on bounding boxes, difficulty in handling multiple targets, and insufficient utilization of contextual information. For example, handling multi-person scenarios is challenging, especially when multiple bodies occlude or overlap each other. As the keypoint localization process within each bounding box is usually independent, the model cannot fully leverage global contextual information, leading to sub-optimal performance in both bounding box detection and keypoint localization.

2.2. Bottom-Up Methods

Bottom-up methods first detect all possible keypoints, then combine these keypoints into complete human poses through the use of clustering or matching techniques. The main characteristic of these methods is building the overall pose from local features (keypoints) incrementally. Representative methods include OpenPOSE, PersonLab, and HigherHRNet. While bottom-up methods are relatively effective in handling multi-person scenarios, their practical application is limited by their complex post-processing, difficulty in keypoint association, high computational burden, insufficient robustness, and reliance on global consistency.

Researchers have made various improvements and innovations in the field of human pose estimation. Zhang et al. [16] have proposed HF-HRNet, which enables enhanced multi-scale spatial feature fusion through the use of a cascaded depth (CAD) module and multi-scale context embedding (MCE) module, resulting in notable improvements in inference speed and model accuracy. Yang et al. [17] have introduced TransPOSE, which leverages Transformers and the self-attention mechanism to provide a similarity measure space between positional pairs, enabling the correct association of keypoints to corresponding human instances and effectively capturing long-term relationships. Qiu et al. [18] have proposed DiffusionPOSE, a model that significantly improves the accuracy of identifying human keypoints through utilization of noise heatmap generation, a progressive denoising process, conditional supervision based on human structural information, and an efficient architectural design. Liu et al. [19] have introduced LDCNet, which incorporates limb direction cues (LDCs) to suppress uncertainties in keypoint positions, thus preventing deep networks from overfitting uncertain keypoint locations. Moreover, researchers have experimented with incorporating attention mechanisms into networks, such as LSKA [20], DAttention [21], CA [22], and SA [23]. Notably, DAttention introduces a deformable self-attention module that selects the positions of keys and values in a data-dependent manner, allowing the model to focus more effectively on relevant areas and reduce the influence of irrelevant features.

The aforementioned algorithms have been shown to provide good inference efficiency and competitive detection accuracy in the field of human pose estimation. However, they still have shortcomings in addressing small target detection and arbitrary occlusion problems. Therefore, it is necessary to develop a human pose estimation method that can effectively address these issues.

3. Method

3.1. Overview

The KSL-POSE model builds upon the YOLOv8l-POSE real-time framework for human pose estimation. Specifically, we draw inspiration from the recent advancements in Kolmogorov–Arnold Networks (KANs) and incorporate non-linear activation functions based on the C2f module, which we refer to as the C2f_KAN module. Moreover, we introduce the small object enhance pyramid (SOEP) module, which processes feature layers through SPDConv to extract feature information enriched with small object details, and integrates this information into the corresponding feature layers. We also incorporate a custom layered shared convolution with batch normalization detection head (LSCB). This design improves the accuracy and efficiency of keypoint detection through multi-scale feature fusion and shared convolutional layers. Through end-to-end training, KSL-POSE can accurately localize human regions and regress keypoints in real-time for each body part in an image. The overall framework is illustrated in Figure 2.

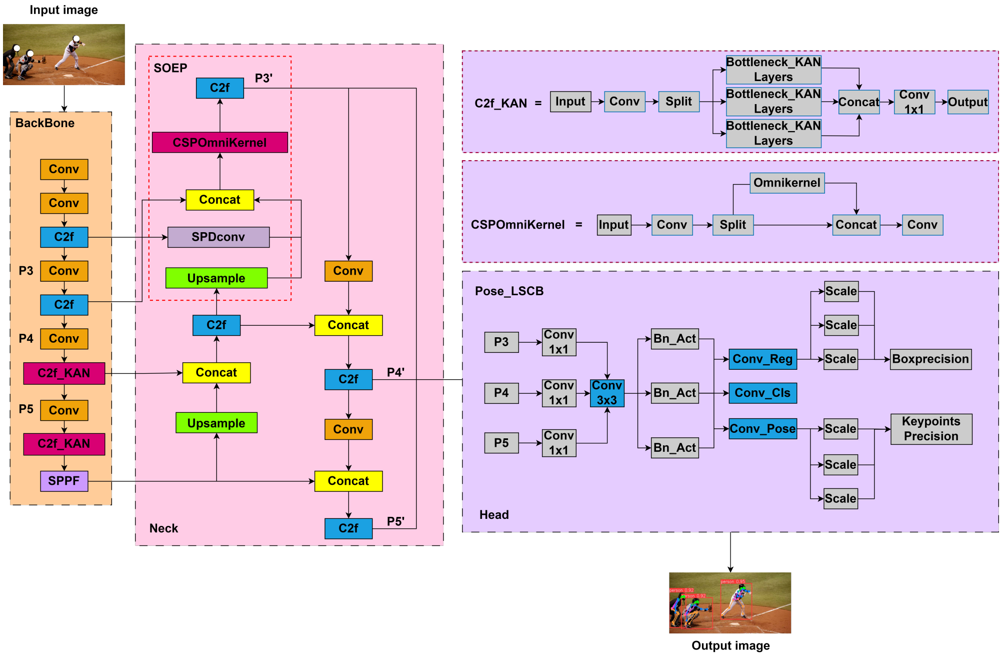

Figure 2. The overall framework of the KSL-POSE model.

3.2. C2f_KAN Module

In human pose estimation tasks, the use of standard convolution modules may lead to insufficient utilization of computational resources due to a large number of redundant features, as well as internal covariate shift issues during the training process, which can result in inadequate image feature extraction. To address these problems, we introduce the C2f_KAN module into the backbone part. This module incorporates the idea of instance normalization and has significant advantages in feature extraction and information fusion, making it particularly suitable for handling fine-grained features in human pose estimation. Its structure is shown in Figure 3.

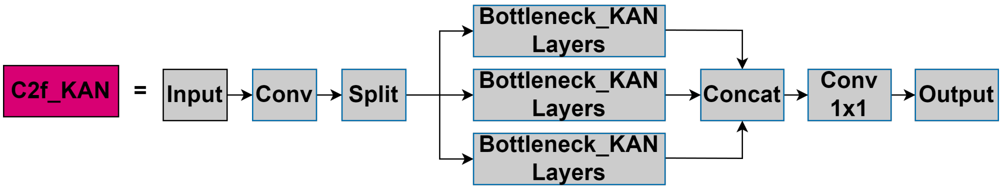

Figure 3. C2f_KAN module diagram.

The introduction of the KAN caused a sensation. In contrast to traditional MLPs, which use fixed activation functions at the nodes (“neurons”), a KAN implements learnable activation functions along the edges (“weights”). Specifically, the weight parameters in a KAN are replaced by univariate functions parameterized as spline curves, thereby eliminating linear weights. This adjustment allows KANs to excel in small-scale AI and scientific tasks, obtaining advances in both accuracy and interpretability. The underlying mathematical principle can be written as follows:

$$f(x) = f(x_1, \cdots, x_n) = \sum_{q=1}^{2n+1} \Phi_q\Big(\sum_{p=1}^{n} \phi_{q,p}(x_p)\Big), \tag{1}$$

where $\Phi_{q,p} \colon [0,1] \to R$ and $\Phi_q \colon R \to R$.

To optimize computation, KAN convolution incorporates a basis function, $b(x)$, in the design of the activation function. Therefore, the activation function $\phi(x)$ is the sum of the basis function $b(x)$ and the spline function $spline(x)$. Specifically, their mathematical formulas can be written as follows:

$$\phi(x) = w(b(x) + spline(x)), \tag{2}$$

$$b(x) = silu(x) = x/(1 + e^{-x}), \tag{3}$$

$$spline(x) = \sum_i c_i B_i(x), \tag{4}$$

where ω is a weight that can better control the overall size of the activation function, c_i denotes the coefficients that are optimized during training, $B_i(x)$ is a B-spline basis function defined on the grid, and the grid points define the interval where each basis function $B_i(x)$ is active, significantly affecting the shape and smoothness. During training, the parameters c_i of these splines (i.e., the coefficients of the basis function $B_i(x)$) are optimized to minimize the loss function, thereby adjusting the shape of the spline to best fit the training data.

The C2f_KAN module, as shown in Algorithms 1–3, inspired by the KAN, retains the efficient feature extraction capabilities of the original C2f module while significantly enhancing its feature extraction and adaptability. Through integrating KAN convolution and non-linear activation functions, this module strengthens the convolutional kernels, enabling them to capture more intricate and subtle features. The C2f_KAN module is utilized in the P4 and P5 layers of the backbone network.

To balance parameter reduction with feature extraction performance, multiple Bottleneck_KAN modules are used to progressively extract and refine features, ensuring that the final output feature map is both rich and high in quality. Moreover, during the concatenation of features from different layers, 1 × 1 convolution is applied for dimensionality reduction and fusion, ensuring that the feature map’s dimensions and information content are optimal for further processing. The convolution function in the C2f_KAN module can be written as follows:

$$y_{i,j,k} = \sum_{m=1}^{M} \sum_{n=1}^{N} x_{i+m-1,j+n-1,l} \cdot w_{m,n,l,k} + b_k, \tag{5}$$

where x represents the input feature map with dimensions $H \times W \times C$ (H is the height, W is the width, and C is the number of channels), ω is the convolution kernel with dimensions

$M \times N \times C \times K$ (M is the kernel height, N is the kernel width, C is the number of input channels, and K is the number of output channels), b_k (corresponding to the output channels K) denotes the bias term, and $y_{i,j,k}$ is the output feature map (where i, j, and k are the indices for the height, width, and channel dimensions of the output feature map, respectively) with dimensions $H' \times W' \times K'$ (H' and W' are the height and width of the output feature map, and K' is the number of output channels)

Algorithm 1 PyTorch-like Code for C2f_KAN Algorithm

```
**Input:** $c1, c2, n, kan\_method, shortcut, g, e$
**Output:** Output tensor after processing through all Bottleneck_KAN layers
Initialize: super()
Create a list of Bottleneck_KAN modules: self.m $\leftarrow$ nn.ModuleList(Bottleneck_KAN($c1, c2, kan\_method, shortcut, g, k = (3,3), e = 1.0$) for $_{\text{in range}}(n)$))
**for** each layer in self.m **do**
    $x \leftarrow$ layer($x$)
**end for**
**Return** $x$
```

Algorithm 2 PyTorch-like Code for Bottleneck_KAN Module

```
**Input:** $c1, c2, kan\_method, shortcut, g, k, e$
**Output:** Output tensor after applying bottleneck operations
Initialize: super()
Compute: $c_{\leftarrow}$int($c2 \times e$)
Define: self.cv1 $\leftarrow$ choose_kan($kan\_method, c1, c_{,}k[0]$)
Define: self.cv2 $\leftarrow$ choose_kan($kan\_method, c_{,}c2, k[1]$)
Compute: self.add $\leftarrow$ shortcut and $c1 == c2$
**if** not self.add **then**
    **Return** self.cv2(self.cv1($x$))
**else**
    **Return** $x +$ self.cv2(self.cv1($x$))
**end if**
```

Algorithm 3 PyTorch-like Code for choose_kan Function

```
**Input:** $name, c1, c2, k$
**Output:** Selected KAN convolutional layer
**if** $name ==$ 'FastKANConv2DLayer' **then**
    kan $\leftarrow$ FastKANConv2DLayer($c1, c2$, kernel_size $= k$, padding $= k//2$)
**else if** $name ==$ 'KANConv2DLayer' **then**
    kan $\leftarrow$ KANConv2DLayer($c1, c2$, kernel_size $= k$, padding $= k//2$)
**else if** $name ==$ 'KALNConv2DLayer' **then**
    kan $\leftarrow$ KALNConv2DLayer($c1, c2$, kernel_size $= k$, padding $= k//2$)
**else if** $name ==$ 'KACNConv2DLayer' **then**
    kan $\leftarrow$ KACNConv2DLayer($c1, c2$, kernel_size $= k$, padding $= k//2$)
**else if** $name ==$ 'KAGNConv2DLayer' **then**
    kan $\leftarrow$ KAGNConv2DLayer($c1, c2$, kernel_size $= k$, padding $= k//2$)
**end if**
**Return** kan
```

3.3. SOEP Module

Detecting small targets during human pose estimation on the P3, P4, and P5 layers can be challenging. Traditional methods often add an extra detection layer before the P3 layer to enhance the detection of small targets. However, this can lead to increased computational complexity and longer post-processing times. To address these challenges, we improved upon the CSP approach and OmniKernel [24] through creating the CSPOmniKernel, which we refer to as the small object enhance pyramid (SOEP). We apply depthwise convolutions with kernel size of $K \times K$ to pursue a large receptive field and use $1 \times K$ and $K \times 1$ depth convolutions to obtain context information. To avoid introducing a large amount of computational overhead, we place the module in the neck part. Then, we explore the possibility of using maximal convolutions for image restoration by gradually increasing k, and reasonably expand the convolution kernel (receptive field) to the feature size while avoiding a large number of parameters. Instead of adding a detection layer, we process the data through SPDConv, which extracts features enriched with small target information. These features are then integrated into the P3 layer. This approach effectively learns feature representations from global to local scales, ultimately improving the detection performance for small targets. The structure of this method is illustrated in Figure 4.

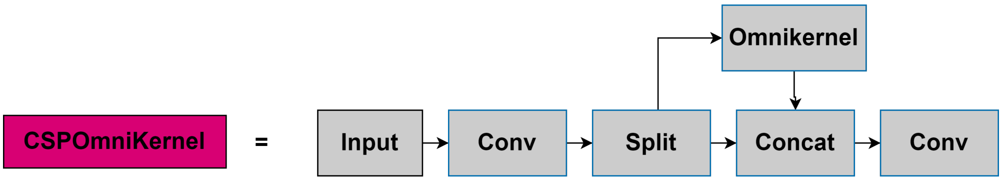

Figure 4. CSP-OmniKernel module diagram.

3.4. LSCB Module

The layered shared convolution with batch normalization (LSCB) detection head is designed to receive feature maps from different scales (P3, P4, P5) using a multi-scale feature fusion method. By merging multi-scale features, it processes the feature maps from different scales to include information of varying granularity, thus capturing more contextual information. This helps the model to more accurately locate keypoints. Moreover, the LSCB module achieves the goal of reducing the number of parameters and improving computational efficiency through the use of shared convolutional layers, ensuring the real-time performance of the model. The resulting diagram is shown in Figure 5.

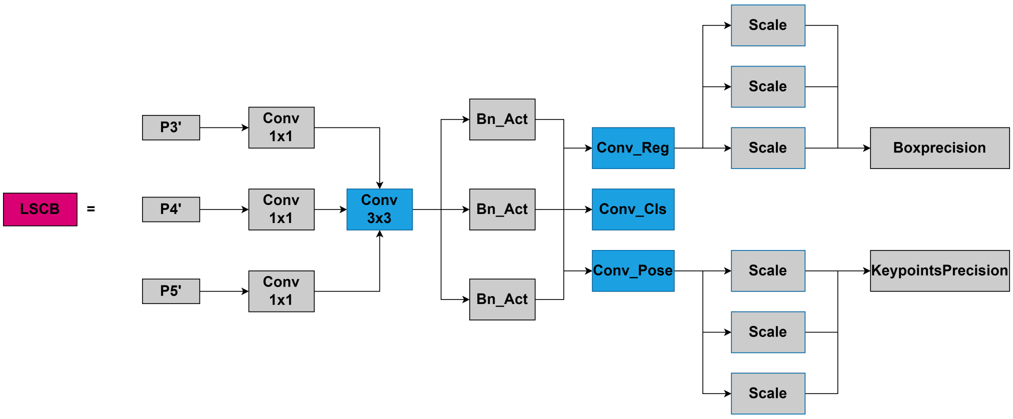

Figure 5. LSCB module diagram. Shared convolutional layers are filled in blue.

High-precision keypoint detection is essential for human pose estimation tasks. In the LSCB module, the keypoint feature processing layers consist of a 1×1 convolution layer, a 3×3 convolution layer, and another 1×1 convolution layer. These layers are designed to handle feature maps at different scales and to extract keypoint features effectively. The LSCB module processes multi-scale (P3′, P4′, P5′) features utilizing shared convolution layers, thereby capturing more contextual information from various scales of feature maps. Given the statistical differences between features at different levels, normalization is necessary. Directly incorporating batch normalization (BN) in the shared parameter detection head can cause errors due to moving average discrepancies. While group normalization (GN) can mitigate this issue, it increases the computational overhead during inference. Therefore, we independently calculate BN for each shared convolution layer, avoiding the moving average errors associated with shared BN. This approach significantly reduces the number of parameters. Moreover, to handle the inconsistent target scales detected by each detection head, we employ a Scale layer to adjust the features, ensuring that each detection head can adapt to different target scales. Specifically, after the three feature layers output from the neck enter the detection head, each branch first passes through a 1 × 1 convolution layer to adjust the number of channels, unifying the number of channels of the three input feature layers to the number of channels in the middle layer. Then, all feature layers are gathered into a shared convolution module for feature extraction. The convolution kernel size is 3 × 3. The use of shared convolution can reduce the number of parameters and calculations of the model. Finally, the regression branch and the classification branch are separated. In the regression branch, a 1 × 1 convolution layer is used to predict the coordinate offset of the bounding box. However, when there are objects with significant differences in size and shape within a scene, shared convolution may not fully adapt to targets of varying scales and shapes, potentially leading to insufficient representational capacity of the feature map. To compensate for its limitations in integrating multi-scale features, architectures like Feature Pyramid Networks (FPNs) or Path Aggregation Networks (PANs) can be introduced on top of shared convolution. These techniques help the model capture features of different scales without significantly increasing the number of parameters. Specifically, in LSCB, to address the issue of inconsistent target scales detected by each detection head, the output of the regression branch utilizes the Scale layer to scale the features and adjust the target scale, thereby enabling the localization of targets of different sizes. A similar mechanism can be observed in architectures like FPN, where multi-scale feature fusion improves detection performance for objects of various sizes. The introduction of the Scale layer is a further optimization of existing multi-scale fusion methods, adding flexibility and controllability. This enhances the applicability of shared convolution in multi-scale target detection and mitigates its negative impact on the quality of feature maps.

In the classification branch, a 1 × 1 convolution layer is used to predict the probability of each category. The convolution layer weights of the two branches are independent, such that the model can learn the positioning and classification tasks separately. Finally, the predicted keypoint features are decoded into actual keypoint coordinates, ensuring the accuracy of keypoint locations and improving detection precision.

Overall, the shared weight parameter design can effectively reduce the number of parameters and calculations, thus improving the running speed of the model, and allows for the processing of features of different scales at the same time, capturing information of various sizes in the image, which helps the model to better understand the relationships between objects in the image, thus improving the recognition accuracy.

4. Experiment

We evaluated the proposed model on both the COCO-POSE 2017 data set and the CrowdPOSE data set. In addition, we conducted ablation experiments to analyze the effects of various modules. These experiments were conducted on the Ubuntu 22.04.4 LTS operating system, utilizing an NVIDIA RTX 3090 GPU for acceleration. The KSL-YOLO model was constructed based on Pytorch 2.3.1 and Cuda 11.5.

4.1. Data Sets

The COCO-POSE 2017 data set [14] is an invaluable and widely utilized resource that has significantly advanced human pose estimation technology. It comprises 250,000 images annotated with keypoints, where each human instance in an image is marked with 17 keypoints. These keypoints provide the coordinates of major human joints, with each keypoint accompanied by a visibility flag indicating its visibility in the image. This data set supports researchers and developers in creating and evaluating advanced computer vision algorithms, thereby facilitating various application scenarios.

The CrowdPOSE data set [25] was specifically designed for multi-person pose estimation tasks, addressing the unique challenges of pose estimation in densely populated scenes. Developed by researchers from the University of Science and Technology of China, Peking University, and SenseTime, this data set serves as a benchmark for high-quality pose estimation in crowded environments. The CrowdPOSE data set consists of 20,000 images, with each human instance annotated with 14 keypoints. It encompasses complex scenarios such as overlapping human instances, partial or complete occlusion by other individuals or objects, and varying degrees of joint deformation. This data set provides a comprehensive and challenging benchmark for pose estimation in dense crowds.

4.2. Evaluation Metrics and Training Configuration

To evaluate the model's performance, this study primarily compares the average precision (AP) and average recall (AR) based on different Object Keypoint Similarity (OKS) thresholds; for example, the average precision at an OKS threshold of 50% (AP50), average recall at an OKS threshold of 50% (AR50), and mean average precision (mAP50-95) across all OKS thresholds from 50% to 95% (in 5% increments). These metrics allowed for a comprehensive evaluation of the model's accuracy and robustness in detecting keypoints across varying degrees of precision and recall.

KSL-POSE is based on the YOLOv8l-POSE framework, and does not use Automatic Mixed Precision (AMP) training. The model uses the Adam optimizer, with a batch size of 16, an input image size of 640 × 640, an initial learning rate of 0.03, and is trained for 350 epochs.

4.3. Evaluation Results

We compared KSL-POSE with other models on the COCO-POSE 2017 data set, as detailed in Table 1.

Table 1. Comparison of various real-time pose estimation methods on the COCO-POSE 2017 data set.

Method	AP	AP50	AP75	APM	APL	AR
Hourglass [26]	63.0	85.7	68.9	58.0	70.4	-
HGG [27]	67.6	85.1	73.7	62.7	74.6	-
OpenPOSE [28]	61.8	84.9	67.5	57.1	68.2	66.5
HRNet	64.5	86.2	71.7	59.2	73.5	-
HigherHRNet [29]	70.5	89.6	76.9	65.1	76.9	-
YOLOv5m6-rlepose [30]	67.6	89.2	74.5	-	73.8	74.0
YOLOv8-PoseBoost [31]	-	85.4	59.1	60.1	74.2	-
YOLOv8l-POSE	67.6	90.0	74.6	63.3	75.7	74.9
SDPose-T [32]	69.2	90.2	76.8	65.7	75.2	-
YOLOv8x-POSE	71.7	91.5	78.4	68.2	78.3	77.9
Diffusion RegPose [33]	72.5	89.9	79.5	66.8	80.5	-
Ours	69.1	90.9	75.5	65.2	76.6	76.7

The human pose estimation performance of KSL-POSE on the COCO-POSE 2017-val data set is shown in Figure 6.

Moreover, we also conducted comparisons on the CrowdPOSE data set, and the results are detailed in Table 2.

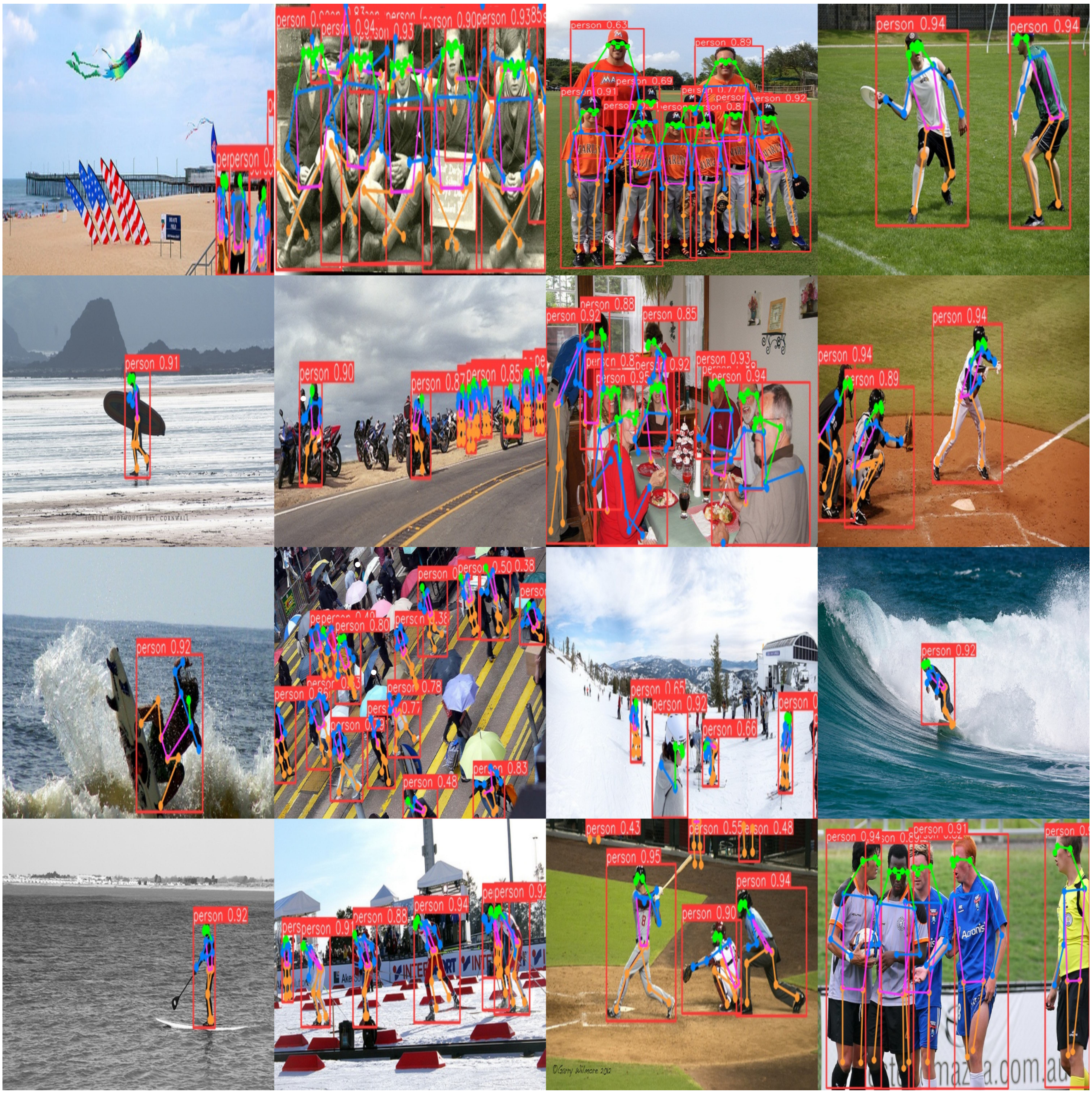

Figure 6. Results on the COCO-POSE 2017-val data set. The red boxes depict pedestrians of varying scales and the content in red boxes represent the accuracy of recognition, we have marked the keypoints with different colors.

Table 2. Comparison of various real-time pose estimation methods on the CrowdPose data set.

Method	AP	AP50	AP75	APE	APM	APH
OpenPOSE	48.0	61.5	53.7	62.7	48.7	32.3
Mask-RCNN [34]	57.2	83.5	60.3	69.4	57.9	45.8
SDPose-S-V2 [32]	64.5	-	-	-	-	-
HrHRNet-W48 [29]	67.6	87.4	72.6	75.8	68.1	58.9
OpenPifPaf [35]	70.5	89.1	76.1	63.8	78.4	72.1
Diffusion RegPose [33]	72.7	91.1	79.3	79.3	73.3	64.9
ViTPOSE-B [36]	73.4	91.8	77.9	78.8	75.1	64.5
Ours	68.3	89.6	74.9	77.2	70.5	60.8

The human pose estimation performance of KSL-POSE on the CrowdPOSE-val data set is shown in Figure 7.

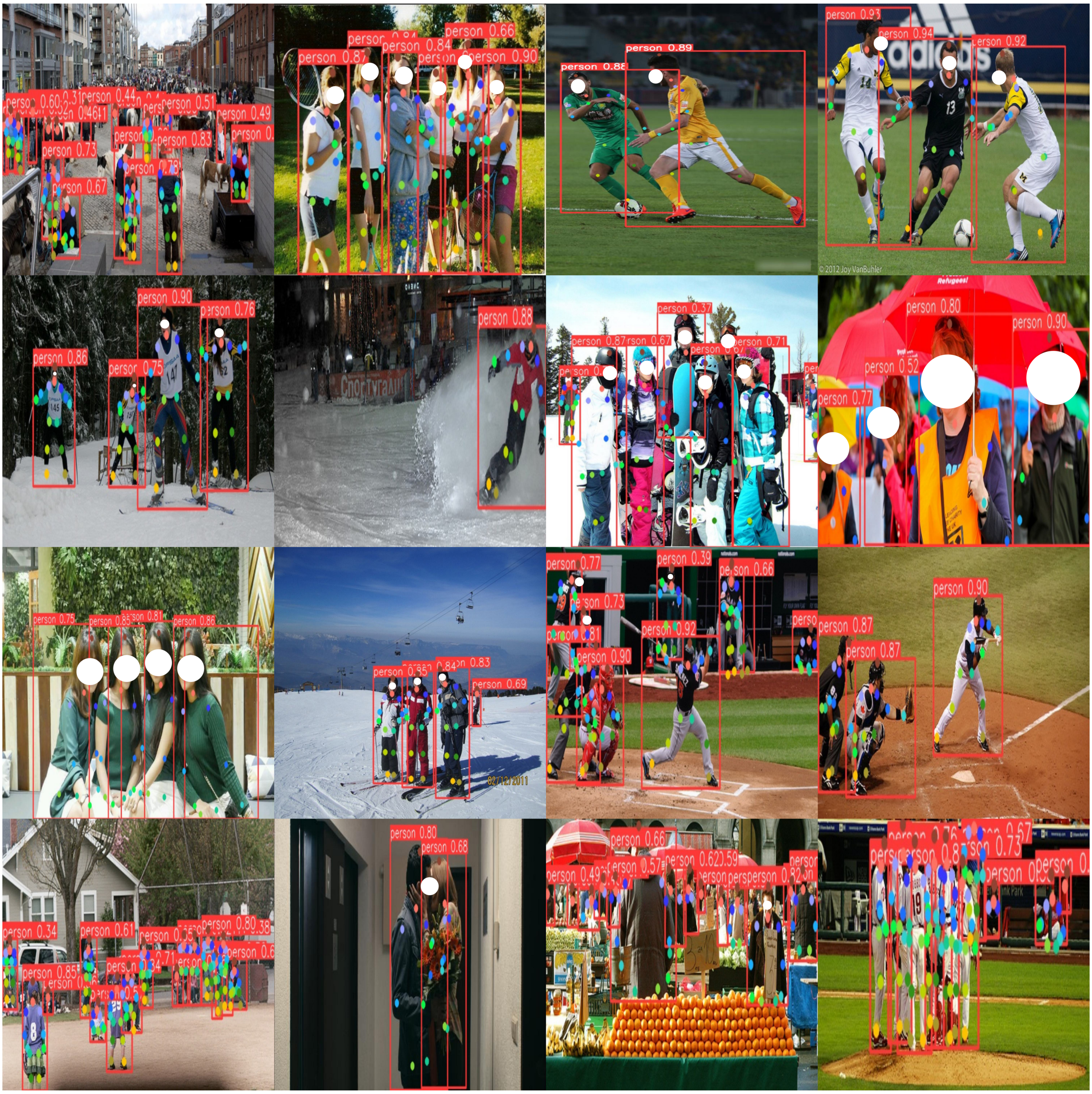

Figure 7. Results on the CrowdPOSE-val data set. The red boxes depict pedestrians of varying scales and the content in red boxes represent the accuracy of recognition; we have marked the keypoints with different colors.

The experimental results show that KSL-POSE improved the average precision on the COCO-POSE 2017-val set by 1.5% and AR by 2.2%, compared to the baseline YOLOV8l-POSE. The overall performance was not significantly different from that of YOLOv8l-POSE. Meanwhile, it also demonstrated competitive performance on the CrowdPOSE data set.

KSL-POSE performed well on both the COCO-POSE 2017-val and CrowdPOSE-val data sets, with its human pose estimation results shown in Figures 6 and 7, respectively.

When dealing with scenarios such as dense crowds, occluded body parts, and individual intersections, KSL-POSE maintained the accuracy of pose estimation. Moreover, it accurately identified and estimated the poses of most small targets, regardless of their relative positions in the image.

In summary, KSL-POSE presented good adaptability, robustness, and generalization in various practical application scenarios.

4.4. Ablation Experiments

To further evaluate the contribution of each module within the overall network structure, we conducted ablation experiments to assess the impact of each module on the entire network. We evaluated the effectiveness of each module through comparing the AP, AP50, and AP75. The ablation experiments were conducted on the COCO-POSE 2017-val set, and the specific experimental results are detailed in Table 3.

Table 3. Ablation study of the proposed real-time pose estimation method on the CrowdPose data set.

Method	AP	AP50	AP75	FLOPs
YOLOv8l-POSE	67.6	90.0	74.6	168.6
+KAN+SOEP+LSCB	69.1	90.9	75.5	198.9
+KAN+SOEP	69.8	91.1	76.1	236.4
+SOEP+LSCB	68.8	89.7	73.4	187.7
+KAN+LSCB	67.2	88.8	72.3	130.1
+KAN	68.4	89.6	73.8	168.6
+SOEP	70.2	91.2	75.9	225.4
+LSCB	68.7	89.4	71.6	117.3

The ablation experiments indicated that KSL-POSE, compared to the baseline model, improved the average detection accuracy while only increasing the processing time by 30.3 ms. Specifically, AP50 and AP75 were improved by 0.9% and 1.1%, respectively. With the addition of the LSCB detection head, we effectively reduced the Floating Point Operations (FLOPs) and improved the model's computational speed. Although incorporating this module slightly decreases recognition accuracy, as shown in Table 3, a comparison between the methods "+KAN+SOEP+LSCB" and "+KAN+SOEP" reveals that we can reduce the FLOPs by 37.5B while only sacrificing 0.7% in accuracy. This enhances the model's computational efficiency and speed. On the other hand, when the LSCB module is not included, compared to the baseline model YOLOv8l-POSE, although the recognition accuracy increases, the FLOPs increase by 67.8B, significantly reducing the model's computational efficiency. Overall, KSL-POSE gained significantly enhanced human pose estimation performance through innovatively combining the efficient feature extraction module C2f_KAN, the detail feature enhancement module SOEP, and the refined keypoint detection head LSCB.

Moreover, we visualize the results of the YOLOv8l-POSE and KSL-POSE methods in Figure 8, highlighting the areas where the former failed to accurately recognize keypoints with green circles, while the red boxes depict pedestrians of varying scales and the content in red boxes represent the accuracy of recognition. YOLOv8l-POSE could not accurately identify keypoints or small targets when processing images with occluded body parts and overlapped individuals. However, through small target monitoring and shared convolutional layer methods, KSL-POSE can regress human pose positioning and corresponding skeletons in real-time and achieve precise and accurate recognition. The obtained results demonstrate that KSL-POSE provides an effective solution for human pose estimation.

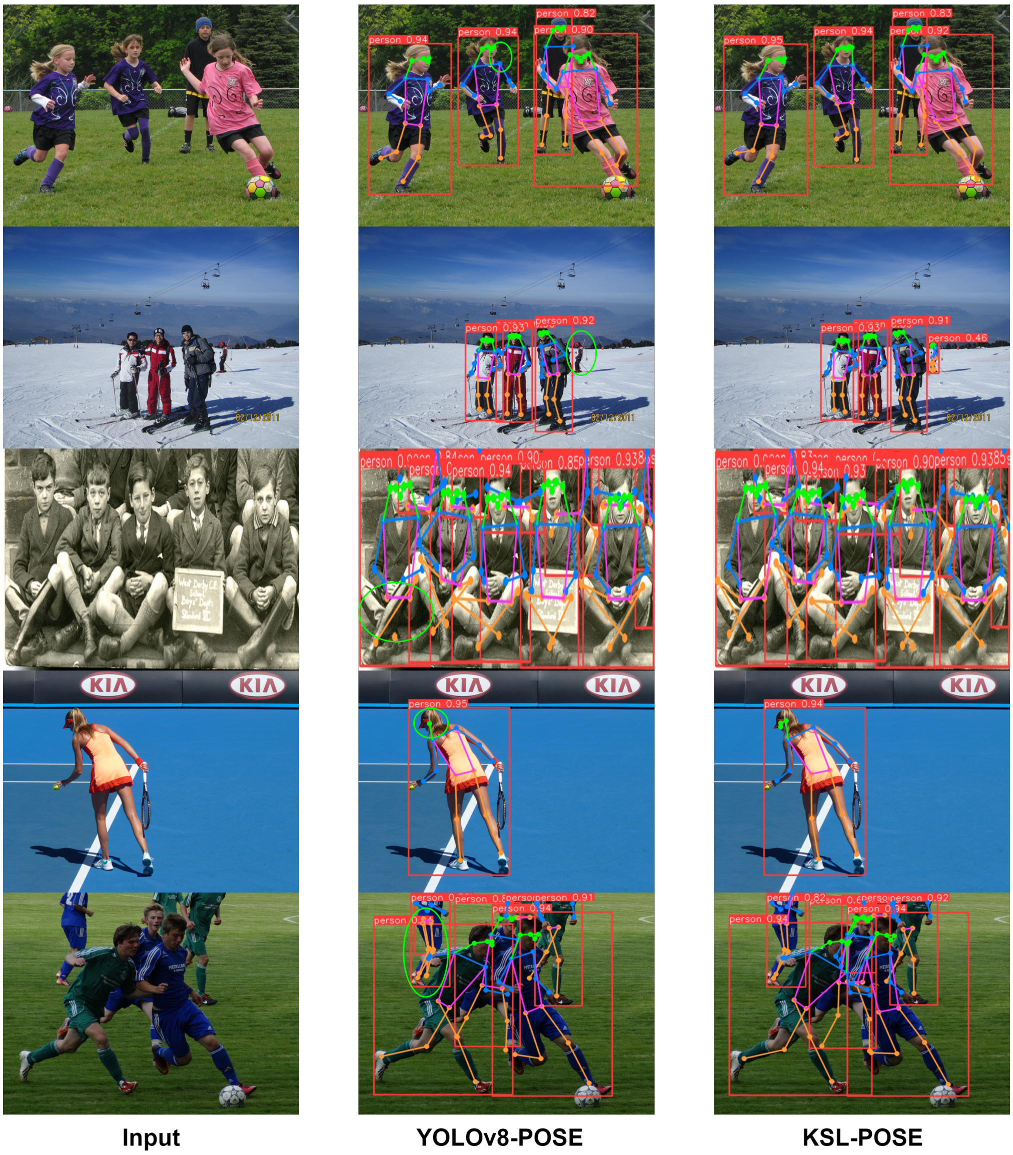

Figure 8. A visual comparison of YOLOv8l-POSE and KSL-POSE, with differences highlighted in green circles.

5. Conclusions

In this study, we proposed an enhanced human pose estimation model, called KSL-POSE, based on YOLOv8l-POSE. Compared with other pose estimation methods, we are more concerned with solving the problems that occur when keypoints are not prominent or occluded. Therefore, we drew inspiration from KAN and introduced non-linear activation functions and instance normalization ideas to handle fine-grained features in human pose estimation. As shown in Table 3, KSL-POSE increased the AP value of the model by 0.8%. In addition, we also introduced SOEP and LSCB modules to optimize the quality of the corresponding generated keypoints through small target detection, multi-scale feature fusion, and shared convolutional layer methods, in order to better adapt to local occlusion in the human pose recognition task and improve the model's ability to determine human joint points. With these modifications, Table 3 demonstrates a 1.5% improvement in the model's AP. Through introducing the C2f_KAN module, SOEP module, and POSE_LSCB detection head, KSL-POSE achieves significant improvements in both keypoint detection accuracy and computational efficiency.

KSL-POSE was thoroughly evaluated on the COCO-POSE 2017 and CrowdPOSE human pose estimation data sets. The results indicate that this model surpasses the existing YOLOv8l-POSE model, in terms of average precision (AP) and average recall (AR). Although KSL-POSE showed a slight increase in FLOPs, and increased model complexity, it can capture more feature information, thus performing better in handling occluded targets and small object detection. Overall, KSL-POSE can efficiently perform real-time human pose estimation tasks. Future research will focus on further optimizing the KSL-POSE model to make it more lightweight, aiming to reduce the number of parameters and computational cost while maintaining its accuracy, thereby enhancing its adaptability to various environments and tasks.

Author Contributions: Conceptualization, T.L. and K.C.; methodology, K.C.; validation, T.L., S.Q. and X.H.; writing—original draft preparation, T.L.; writing—review and editing, K.C. and X.H.; supervision, K.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Science Foundation Program of China (NSFC) (grant number: 61976241), Postgraduate Research and Practice Innovation Program of Jiangsu Province (grant number: KYCX24_4129), and the International Science and technology cooperationplan project of Zhenjiang (grant number: GJ2021008).

Institutional Review Board Statement: Ethical review and approval were waived for this study, due to the fact that we have implemented privacy protection for the volunteers in the images.

Informed Consent Statement: Patient consent was waived due to the fact that we have implemented privacy protection for the volunteers in the images.

Data Availability Statement: The data sets analyzed in the study are openly available in [CrowdPose] at https://github.com/Jeff-sjtu/CrowdPose.git (accessed on 15 March 2024) and [MS COCO] at https://cocodataset.org/ (accessed on 20 December 2023).

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

Abbreviations

The following abbreviations are used in this manuscript:

MDPI	Multidisciplinary Digital Publishing Institute
DOAJ	Directory of open access journals
TLA	Three letter acronym
LD	Linear dichroism

References

1. Li, Y.; Jia, S.; Li, Q. BalanceHRNet: An effective network for bottom-up human pose estimation. *Neural Netw.* **2023**, *161*, 297–305. [CrossRef] [PubMed]
2. Zheng, C.; Wu, W.; Chen, C.; Yang, T.; Zhu, S.; Shen, J.; Kehtarnavaz, N.; Shah, M. Deep learning-based human pose estimation: A survey. *ACM Comput. Surv.* **2023**, *56*, 11. [CrossRef]
3. Zhang, J.; Zheng, Y.; Qi, D.; Li, R.; Yi, X. DNN-based prediction model for spatio-temporal data. In Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, Burlingame, CA, USA, 31 October–3 November 2016; pp. 1–4.
4. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
5. Donahue, J.; Anne Hendricks, L.; Guadarrama, S.; Rohrbach, M.; Venugopalan, S.; Saenko, K.; Darrell, T. Long-term recurrent convolutional networks for visual recognition and description. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 2625–2634.
6. Park, J.; Woo, S.; Lee, J.Y.; Kweon, I.S. Bam: Bottleneck attention module. *arXiv* **2018**, arXiv:1807.06514.
7. Ren, Z.; Zhou, Y.; Chen, Y.; Zhou, R.; Gao, Y. Efficient human pose estimation by maximizing fusion and high-level spatial attention. In Proceedings of the 2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021), Jodhpur, India, 15–18 December 2021; pp. 1–6.
8. Liu, Z.; Wang, Y.; Vaidya, S.; Ruehle, F.; Halverson, J.; Soljačić, M.; Hou, T.Y.; Tegmark, M. Kan: Kolmogorov-arnold networks. *arXiv* **2024**, arXiv:2404.19756.
9. Han, D.; Yun, S.; Heo, B.; Yoo, Y. Rexnet: Diminishing representational bottleneck on convolutional neural network. *arXiv* **2020**, arXiv:2007.00992.

10. Qian, S.; Ning, C.; Hu, Y. MobileNetV3 for image classification. In Proceedings of the 2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE), Nanchang, China, 26–28 March 2021; pp. 490–497.
11. Huang, Z.; Wang, J.; Fu, X.; Yu, T.; Guo, Y.; Wang, R. DC-SPP-YOLO: Dense connection and spatial pyramid pooling based YOLO for object detection. *Inf. Sci.* **2020**, *522*, 241–258. [CrossRef]
12. Dubey, A.; Santra, A.; Fuchs, J.; Lübke, M.; Weigel, R.; Lurz, F. HARadNet: Anchor-free target detection for radar point clouds using hierarchical attention and multi-task learning. *Mach. Learn. Appl.* **2022**, *8*, 100275. [CrossRef]
13. Ghiasi, G.; Lin, T.Y.; Le, Q.V. Nas-fpn: Learning scalable feature pyramid architecture for object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7036–7045.
14. 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.
15. Hu, M.; Feng, J.; Hua, J.; Lai, B.; Huang, J.; Gong, X.; Hua, X.S. Online convolutional re-parameterization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 568–577.
16. Zhang, H.; Dun, Y.; Pei, Y.; Lai, S.; Liu, C.; Zhang, K.; Qian, X. HF-HRNet: A simple hardware friendly high-resolution network. *IEEE Trans. Circuits Syst. Video Technol.* **2024**, *34*, 7699–7711. [CrossRef]
17. Yang, S.; Quan, Z.; Nie, M.; Yang, W. Transpose: Keypoint localization via transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 11802–11812.
18. Qiu, Z.; Yang, Q.; Wang, J.; Wang, X.; Xu, C.; Fu, D.; Yao, K.; Han, J.; Ding, E.; Wang, J. Learning structure-guided diffusion model for 2d human pose estimation. *arXiv* **2023**, arXiv:2306.17074.
19. Liu, T.; Liu, H.; Yang, B.; Zhang, Z. LDCNet: Limb direction cues-aware network for flexible human pose estimation in industrial behavioral biometrics systems. *IEEE Trans. Ind. Inform.* **2023**, *20*, 8068–8078. [CrossRef]
20. Lau, K.W.; Po, L.M.; Rehman, Y.A.U. Large separable kernel attention: Rethinking the large kernel attention design in cnn. *Expert Syst. Appl.* **2024**, *236*, 121352. [CrossRef]
21. Xia, Z.; Pan, X.; Song, S.; Li, L.E.; Huang, G. Vision transformer with deformable attention. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 4794–4803.
22. Hou, Q.; Zhou, D.; Feng, J. Coordinate attention for efficient mobile network design. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 13713–13722.
23. Zhang, Q.L.; Yang, Y.B. Sa-net: Shuffle attention for deep convolutional neural networks. In Proceedings of the ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 2235–2239.
24. Cui, Y.; Ren, W.; Knoll, A. Omni-Kernel Network for Image Restoration. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 20–27 February 2024 ; Volume 38, pp. 1426–1434.
25. Li, J.; Wang, C.; Zhu, H.; Mao, Y.; Fang, H.S.; Lu, C. Crowdpose: Efficient crowded scenes pose estimation and a new benchmark. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 10863–10872.
26. Newell, A.; Yang, K.; Deng, J. Stacked Hourglass Networks for Human Pose Estimation. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016.
27. Hua, G.; Li, L.; Liu, S. Multipath affinage stacked—Hourglass networks for human pose estimation. *Front. Comput. Sci.* **2020**, *14*, 144701. [CrossRef]
28. Liu, D.; Zhao, Z.; Wang, X.; Hu, Y.; Zhang, L.; Huang, T. Improving 3D human pose estimation via 3D part affinity fields. In Proceedings of the 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa Village, HI, USA, 7–11 January 2019; pp. 1004–1013.
29. 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.
30. Jiang, Y.; Yang, K.; Zhu, J.; Qin, L. YOLO-Rlepose: Improved YOLO Based on Swin Transformer and Rle-Oks Loss for Multi-Person Pose Estimation. *Electronics* **2024**, *13*, 563. [CrossRef]
31. Wang, F.; Wang, G.; Lu, B. YOLOv8-PoseBoost: Advancements in Multimodal Robot Pose Keypoint Detection. *Electronics* **2024**, *13*, 1046. [CrossRef]
32. Chen, S.; Zhang, Y.; Huang, S.; Yi, R.; Fan, K.; Zhang, R.; Chen, P.; Wang, J.; Ding, S.; Ma, L. SDPose: Tokenized Pose Estimation via Circulation-Guide Self-Distillation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024; pp. 1082–1090.
33. Tan, D.; Chen, H.; Tian, W.; Xiong, L. DiffusionRegPose: Enhancing Multi-Person Pose Estimation using a Diffusion-Based End-to-End Regression Approach. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024; pp. 2230–2239.
34. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.

35. Kreiss, S.; Bertoni, L.; Alahi, A. Openpifpaf: Composite fields for semantic keypoint detection and spatio-temporal association. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 13498–13511. [CrossRef]
36. Xu, Y.; Zhang, J.; Zhang, Q.; Tao, D. Vitpose: Simple vision transformer baselines for human pose estimation. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 38571–38584.

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.