electronics

Article

Single-Stage Pose Estimation and Joint Angle Extraction Method for Moving Human Body

Shuxian Wang [1], Xiaoxun Zhang [1,*], Fang Ma [2], Jiaming Li [1] and Yuanyou Huang [1]

[1] School of Materials Science and Engineering, Shanghai University of Engineering Science, Shanghai 201620, China
[2] School of Mechanical and Automotive Engineering, Shanghai University of Engineering Science, Shanghai 201620, China
* Correspondence: xx.zhang@sues.edu.cn

Abstract: Detecting posture changes of athletes in sports is an important task in teaching and training competitions, but its detection remains challenging due to the diversity and complexity of sports postures. This paper introduces a single-stage pose estimation algorithm named yolov8-sp. This algorithm enhances the original yolov8 architecture by incorporating the concept of multi-dimensional feature fusion and the attention mechanism for automatically capturing feature importance. Furthermore, in this paper, angle extraction is conducted for three crucial motion joints in the motion scene, with polynomial corrections applied across successive frames. In comparison with the baseline yolov8, the improved model significantly outperforms it in AP^{50} (average precision) aspects. Specifically, the model's performance improves from 84.5 AP to 87.1 AP, and the performance of $AP^{50–95}$, AP^{M}, and AP^{L} aspects also shows varying degrees of improvement; the joint angle detection accuracy under different sports scenarios is tested, and the overall accuracy is improved from 73.2% to 89.0%, which proves the feasibility of the method for posture estimation of the human body in sports and provides a reliable tool for the analysis of athletes' joint angles.

Keywords: pose estimation; yolov8-sp; joint angle extraction; feature extraction; multi-dimensional feature fusion

Citation: Wang, S.; Zhang, X.; Ma, F.; Li, J.; Huang, Y. Single-Stage Pose Estimation and Joint Angle Extraction Method for Moving Human Body. *Electronics* **2023**, *12*, 4644. https://doi.org/10.3390/electronics12224644

Academic Editors: George A. Papakostas and George A. Tsihrintzis

Received: 27 September 2023
Revised: 25 October 2023
Accepted: 8 November 2023
Published: 14 November 2023

1. Introduction

In real-world scenarios, precise and real-time posture estimation is pivotal for comprehending an athlete's present condition. Gathering data pertinent to an athlete's performance during training sessions or competitions allows players and coaches to delve deeper into specific movement details, such as shifts in elbow angles, muscle dynamics, and the trajectory of arm swings. This depth of understanding aids in mastering the nuances of movements and postures, proactively addressing potential injuries, and ensuring timely interventions for any movement discrepancies. For instance, in tennis, the angular dynamics between the right and left arms during a serve can shed light on the intricacies of the double-reverse technique. Similarly, in sports like field hockey and ice hockey, monitoring variations in hand positioning, movement, and flexion can offer crucial insights during gameplay. Moreover, spotting asymmetries in activities like running or race walking can serve as a preemptive measure against potential sports-related injuries. Furthermore, in motion tracking, abnormal shifts in joint angles can be indicative of mishaps such as falls.

This paper focuses on the recovery of human pose data in sports scenarios. Current research predominantly targets general motion detection, leaving a notable gap for data extraction and processing in specialized sports environments. Human pose estimation stands as the foundation for achieving this. There are two principal paradigms for pose estimation: top-down and bottom-up. The primary distinction between them involves the employment of manual detection followed by subsequent processes. The top-down [1–6] approach first utilizes a human detector to segment and crop each individual into distinct image patches.

Sequentially, pose estimation is systematically applied to each segmented image. Over time, numerous strategies have been developed to heighten pose estimation accuracy through the top-down method. Noteworthy strategies encompass the multitasking framework [1], feature pyramid [2], and high-resolution feature map [4]. By normalizing each image segment, the per-person scale is significantly reduced, fitting for convolutional neural network (CNN) training. The Simple Baseline [5] introduces a streamlined architecture combined with a deep backbone and several deconvolutional layers, amplifying the resolution of the output features.

In contrast, the bottom-up methods [7–13] offer consistent run times and process the entire image in one go, estimating keypoints for each individual and subsequently classifying these keypoints. They typically depend on heatmaps for intricate post-processing steps. With the bottom-up strategy, a single forward pass can capture all potential keypoints from the input image. A subsequent association phase groups these keypoints into distinct skeletal structures, making it a more streamlined process than the segmented sequential processing seen in the top-down approach. Nonetheless, the post-processing stage of this method can encompass tasks like pixel-level Non-Maximum Suppression (NMS), line integration, fine-tuning, and clustering. Adjusting and refining coordinates serve to diminish the quantization errors associated with downsampled heatmaps. Meanwhile, NMS aims to pinpoint local peaks within the heatmap. However, even with post-processing, the heatmap might still not provide a clear enough distinction between two closely situated joints of the same category. Another limitation is that bottom-up methods cannot be trained from end-to-end, as the post-processing phase is non-differentiable and takes place outside the convolutional network's domain. Oct-MobileNet [14] employs octave convolution and attention mechanisms to improve the extraction of high-frequency features from the human body contour, resulting in enhanced accuracy with reduced model computational burden. MSTPose [15] learns texture features through CNN, captures spatial features of the images through the MST module, and employs one-dimensional vector regression to preserve the position-sensitive spatial sequential mapping structure of the transformer output.

To address the aforementioned challenges, this paper presents a method for human pose estimation in dynamic scenarios without relying on heatmaps. We employ real-time detection of the moving human figure, ensuring promptness, and provide continuous updates on joint angle variations. Our approach incorporates pose predictions at various scales to compensate for scale discrepancies. Furthermore, we introduce the modified c2f (mc2f) module, inspired by attention mechanisms, to accentuate crucial features while diminishing irrelevant ones. Significant advancements in object detection are effortlessly adaptable to our pose estimation framework. The pose estimation methodology proposed herein can be seamlessly integrated into any computer vision system for target detection with minimal computational overhead.

In this study, we present yolov8-sp, a novel extraction method tailored for motion scenes, aiming to precisely delineate pose alterations and angle variations. Three salient enhancements characterize our approach:

(1) The feature fusion module has been refined using a cutting-edge strategy, bolstering the network's capability to discern diminutive targets.
(2) We have integrated the mc2f module, which autonomously prioritizes various features based on their relevance.
(3) Leveraging pose estimation, we extract angular data within the motion setting and apply fitting adjustments to further refine feature information accuracy.

The structure of this paper is as follows: Section 2 provides a succinct overview of pertinent studies in the field. Section 3 delves comprehensively into our methodology, colloquially termed as "yolov8-sp". Section 4 showcases and critically assesses the efficacy of our technique. Section 5 wraps up our research and sketches the trajectory for prospective explorations.

2. Related Work

2.1. Human Body Posture Estimation

In the top-down strategy, Mask-RCNN [1] identifies keypoints using segmentation masks. The Simple Baseline [5] introduces an uncomplicated architecture, equipped with a profound backbone and multiple back-convolutions, aimed at enhancing the resolution of the resultant features. While these methodologies maintain scale invariance, processing all challenges uniformly at a single scale, they falter when it comes to addressing occlusion effectively.

The bottom-up strategy utilizes a probabilistic map anchored on a heatmap, aiming to determine if a particular pixel hosts a specific keypoint. Leveraging Non-Maximum Suppression (NMS), it identifies the precise location of keypoints within the heatmap by pinpointing the local maxima. For instance, the Openpose model [12] integrates two branches, one predicting the keypoint heatmap and the other defining the part affinity field that depicts the 2D vector relationships between keypoints. The more recent DEKR [11] employs a dual-head framework: one head dedicated to keypoint heatmap estimation and the other managing human body bounding box centers and offsets, forgoing the conventional grouping functions. Still, it is dependent on multi-scale evaluations to amplify its precision. CGNet [16] innovates with an attention mechanism that aligns each keypoint to its inherent center, beneficial in tackling occlusion scenarios, but falls short with smaller figures. PINet's methodology [13] segments the human bounding box into three sections, estimating each in isolation, thus bolstering resilience against occlusions. Methods anchored on the human bounding box's center [11,13] employ an offset regression predicated on that center for keypoints. This approach intertwines the keypoints' heatmap with the central heatmap, necessitating multiple NMS iterations in post-processing, thereby augmenting the operational intricacy.

YOLOv8, delineated by Glenn [17], represents an advanced iteration of the YOLO lineage, originally pioneered by Redmon et al. in 2016 [18]. This architecture, dedicated to predicting object bounding boxes and their corresponding classes within images, relies on a singular neural network structure, encompassing a backbone, neck, and detection head components. Taking cues from YOLOv7's ELAN design, YOLOv8 refines the YOLOv5's C3 structure, superseding it with the gradient-enhanced C2f configuration. Concurrently, the channel distribution undergoes optimization, adjusting in accordance with varied model scales. On the loss computation front, YOLOv8 employs the positive sample distribution technique of the TaskAlignedAssigner and integrates the Distribution Focal Loss for added efficacy. Venturing into pose estimation, Yolopose [19] offers a groundbreaking solution. Built upon the YOLO foundation, it negates the need for heatmaps. This innovation facilitates a streamlined, end-to-end methodology, sidestepping the elaborate post-processing inherent to bottom-up strategies. Contrasting with the top-down paradigm, Yolopose excels in localizing all subjects and their corresponding poses in a solitary inference, obviating the need for iterative forward passes. A visual representation of the comprehensive YOLOv8 framework, augmented with a pose estimation header, is showcased in Figure 1.

This approach eliminates the need for the post-processing steps typical of the bottom-up method, such as aggregating detected keypoints into a skeletal structure. This is because each detected frame inherently associates with a pose, thereby naturally grouping the joint points. Moreover, unlike the top-down method, there is no need for multiple forward passes, as all individuals are concurrently localized with their respective poses through a singular inference process [11].

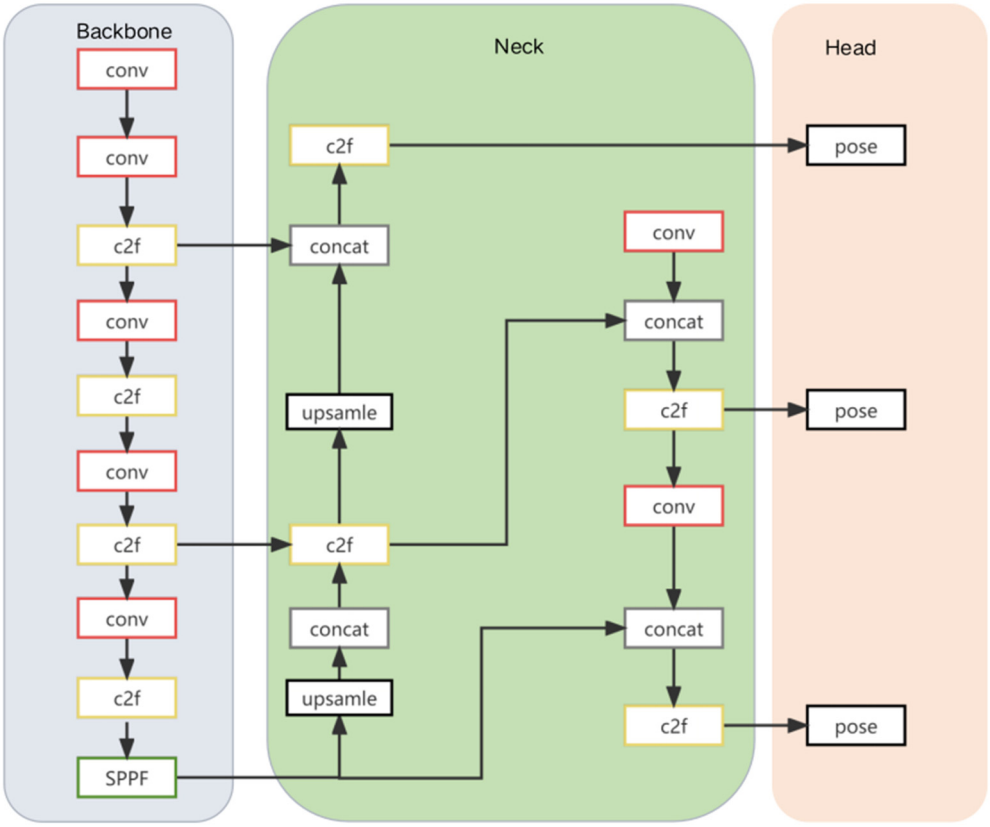

Figure 1. Detailed architecture of Yolov8.

2.2. Human Pose Loss Function Formulation and Human Pose Estimation Loss

Object keypoint similarity (OKS) is a widely recognized metric for evaluating keypoint accuracy. While bottom-up methods reliant on heatmaps traditionally employ the L1 loss for keypoint detection, this loss does not consider the object's size or the specific nature of the keypoint. Given that heatmaps function as probabilistic maps, directly using OKS as a loss function in a purely heatmap-based methodology is not feasible. OKS can only be employed as a loss function during the regression of keypoint locations. Geng [20] and his team pioneered the use of normalized L1 loss for keypoint regression, representing an initial move towards incorporating OKS loss.

In our study, keypoints are directly mapped back to the center of the anchors to optimize the evaluation metrics themselves, eschewing the need for an alternative loss function. We encapsulate the complete pose information within each bounding box. Therefore, if the ground truth bounding box aligns with an anchor point in terms of position and scale ss, we can project the keypoint relative to the anchor point's center. For every individual keypoint, the OKS is calculated distinctly. These individual values are then aggregated to derive the final OKS loss, also referred to as the keypoint underscore loss [11].

$$\text{OKS} = \frac{\sum exp\left(d_i^2/2S^2K_i^2\right)\delta(V_i > 0)}{\sum \delta(V_i > 0)} \tag{1}$$

$$L_{kpts}(\text{s, i, j, k}) = 1 - \sum_{n=1}^{N_{kpts}} OKS = 1 - \frac{\sum_{n=1}^{N_{kpts}} exp(\frac{d_n^2}{2S^2k_n^2})\delta(V_n > 0)}{\sum_{n=1}^{N_{kpts}} \delta(V_n > 0)} \tag{2}$$

d_n = Eucledian distance between and ground truth location for n^{th} keypoint.
k_n = Keypoint specific weights.
s = Scale of an object.
$\delta(v_n)$ = visibilty flag for each keypoint.

For each keypoint, we learn a confidence parameter that indicates its visibility. Through learning this parameter, the model can appropriately weight its predicted keypoints. The visibility status of a keypoint serves as the ground truth, As shown in Equation (4). Binary Cross Entropy (BCE) is a loss function used for binary classification tasks. Binary Cross Entropy evaluates the difference between the predicted and actual values, imposing a more significant penalty on models that predict incorrectly. Specifically, the definition of Binary Cross Entropy is as shown in Equation (3).

$$BCE(p,q) = -(ylog(p) + (1-y)log(1-p)) \quad (3)$$

p = Probability that the model predicts as a positive instance.
y = Ground truth label.

$$L_{kpts\_conf}(s,i,j,k) = \sum_{n=1}^{N_{kpts}} BCE\left(\delta(V_n > 0), P_{kpts}^n\right) \quad (4)$$

P_{kpts}^n = predicted confidence for n^{th} keypoint.

2.3. Human Reconstruction

SMPL [21] operates as a vertex-dependent linear model that breaks down human modeling into distinct shape and pose components, as shown in Figure 2. The pose parameter, denoted by θ, is characterized by a standard skeleton comprising K = 23 labeled points. These points specify the rotation angle of part k, with its orientation in the kinematic tree being determined by the parent node of part k. Conversely, the shape parameter β is extracted using principal component analysis (PCA), representing the m principal components within a reduced-dimensional space. Herein, M(β,θ) is the final three-dimensional human body mesh, which is a function of shape and pose. T(β,θ) is the human body template defined by shape and pose parameters. The function M(β,θ) accepts both θ and β as input parameters, subsequently producing a triangular mesh comprising N = 6890 vertices. This model focuses on the vertices and showcases additive characteristics. Succinctly, a human-like model can be articulated through the subsequent equation [21]:

$$M(\beta,\theta) = W(T(\beta,\theta), J(\beta), \theta; W) \quad (5)$$

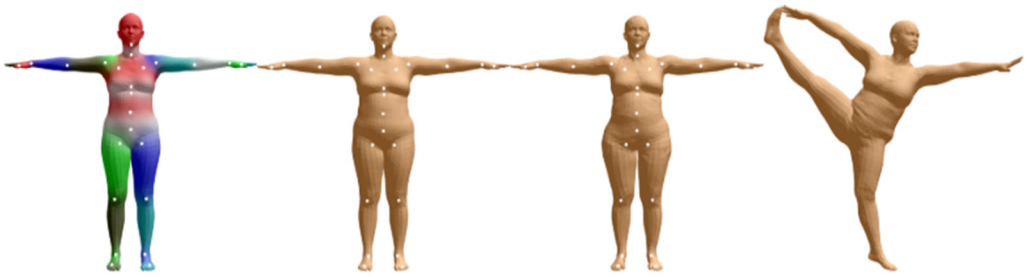

Figure 2. Parameterized representation of the model, adapted with permission from Loper, M.; Mahmood, N.; Romero, J.; Pons-Moll, G.; Black, M.J.'s study 'SMPL: A Skinned Multi-Person Linear Model' [21].

β = Shape parameter, which defines the overall contour of the body.
θ = Pose parameter, which defines the rotation of the body's joints.
J(β) = Human body template defined by shape and pose parameters.
W = Linear blend skinning function.

$$T(\beta,\theta) = \overline{T} + B_s(\beta) + B_p(\theta) \quad (6)$$

$\overline{T}$ = Mean human body template.
$B_s(\beta)$ = Shape blend term, which describes the template variations resulting from shape β.

$B_p(\theta)$ = Pose blend term, which describes the template variations resulting from pose θ.

3. Proposed Method

3.1. Yolov8-sp

In YOLOv8, the fusion of feature information within merely two dimensions frequently results in the loss of vital data, particularly when dealing with high-dimensional datasets. Consequently, this gives rise to a model that inadequately represents the intricacies of the underlying data. Moreover, such a model lacks the flexibility to effectively manage extended sequences, intricate details in imagery, pivotal information, and the like.

In response, we have enhanced YOLOv8, resulting in our proposed YOLOv8-SP—a human detection methodology optimized for motion scenarios. When juxtaposed with the original YOLOv8, our YOLOv8-SP introduces the following key modifications:

(1) Incorporation of a three-dimensional cross-scale feature information fusion MultiCat module.
(2) Substitution of the original C2F module with the advanced MC2F within the backbone.

Figure 3 below provides a detailed description of the proposed YOLOv8-SP.

3.1.1. Multicat Module

One of the challenges in human keypoint detection arises from the diminutive size of small-target samples. Given the substantial downsampling factor of YOLOv8, the deeper feature maps struggle to capture the intricacies of small-target features. To address this, we have incorporated a module that augments the small-target detection layer on shallower feature maps and fuses it with the deeper feature maps. This ensures that the network prioritizes the detection of small targets, leading to enhanced detection outcomes.

As depicted in Figure 4, the module conducts multi-scale feature amalgamation [22] via pooling and interpolation techniques, culminating in three feature maps cohesively merged along the channel dimension. Specifically, for larger scale features, both adaptive max pooling and average pooling (as per Equation (7)) are employed. Adaptive max pooling differs from the conventional max pooling layer in that it does not require a predefined specific window size or stride. Instead, it automatically calculates the necessary pooling window size and stride to produce an output of a specified size. This provides a flexible approach to perform pooling operations, making the network architecture more adaptable and suitable for various application scenarios. These methods capture not only the prominent features (peak intensities) but also the holistic average content across the feature maps. Conversely, for the finer scale features, the inherent high-frequency details are retained through the employment of proximal upsampling (according to Equation (8)), thereby circumventing the introduction of extraneous information or potential ambiguities. Ultimately, these varied scale features are integrated (as described in Equation (9)) across the channel dimension, resulting in an output boasting a depth of L + M + S and dimensions of (L + M + S, H_m, W_m), wherein L represents the large-scale feature maps, M denotes the medium-scale feature maps, and S stands for the small-scale feature maps. This fusion methodology ensures that the model concurrently processes information across all scales, facilitating a more resilient and comprehensive feature representation.

$$L' = AdaptiveMaxPool(L, (H_m, W_m)) + AdaptiveAvgPool(L, (H_m, W_m)) \quad (7)$$

where the dimensions of (L, the H_m, W_m)

$$S' = Upsample(s, mode = 'nearest') \quad (8)$$

where the dimensions of (S, H_m, W_m)

$$\text{Output} = \text{Concatenate}(L', M, S') \quad (9)$$

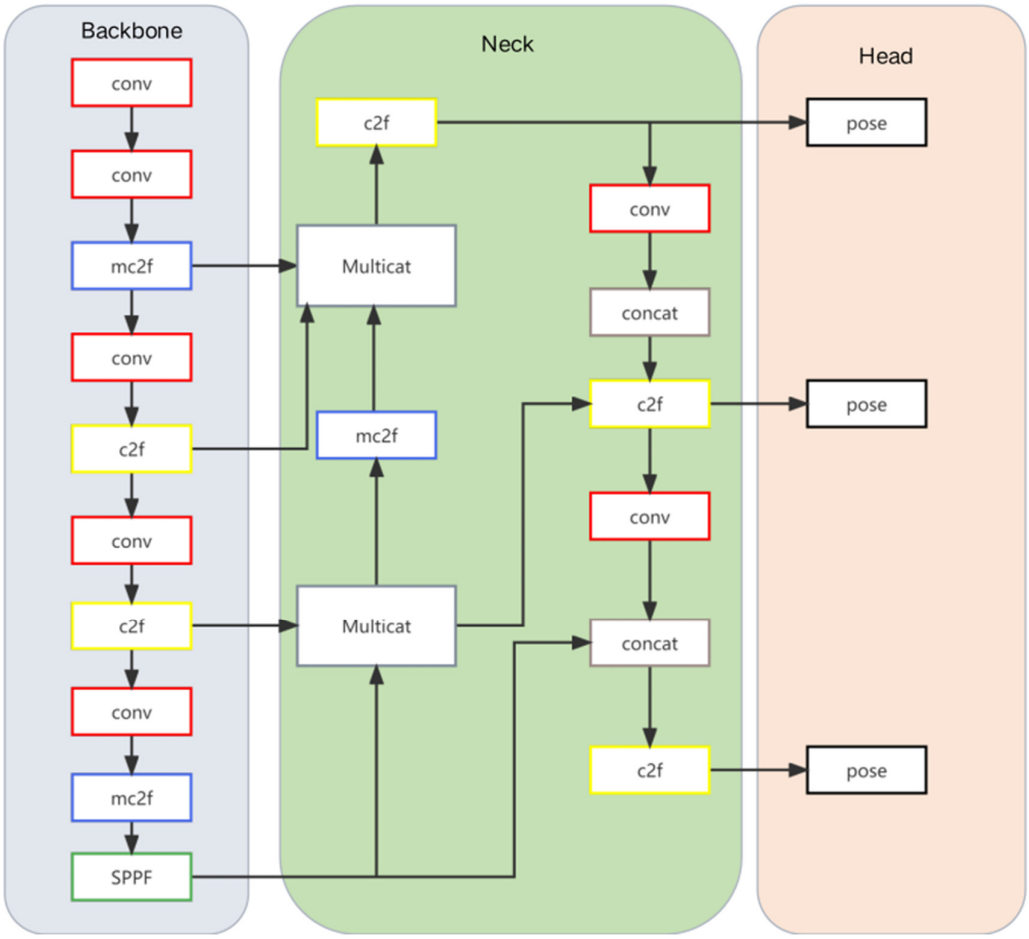

Figure 3. Detailed architecture of yolov8-sp.

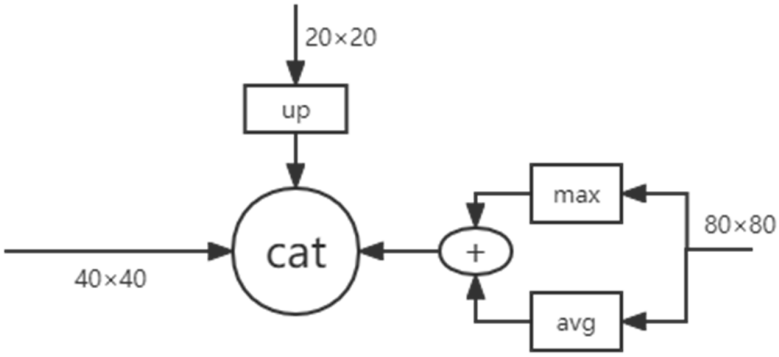

Figure 4. Multicat module.

3.1.2. Mc2f Module

In the realm of deep learning, feature interdependencies across channels are common. The c2f module processes each channel uniformly, without allocating specific weights to individual channels. This approach risks overlooking pivotal information from dominant channels.

In this module, we aim to autonomously discern the significance of each feature channel through learning. By leveraging the importance ascertained, we amplify valuable features while diminishing irrelevant ones. This not only ensures comprehensive capture of global contextual data but also adeptly marries local and global attributes, yielding a more intricate feature representation.

As depicted in Figure 5, we draw inspiration from the Squeeze-and-Excitation (SE) attention mechanism. Initially, the module undergoes global average pooling (Equation (10))

across spatial dimensions, resulting in a 1 × 1 × C vector for a given H × M × C feature map (where H is the height, M is the width, and C is the channel count). This process essentially compresses spatial information into a single representative value, endowing it with a broad, global perspective. A significant benefit of the GAP is its drastic reduction of the model's parameters, effectively curbing overfitting.

$$Z_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{w} X_{cij} \tag{10}$$

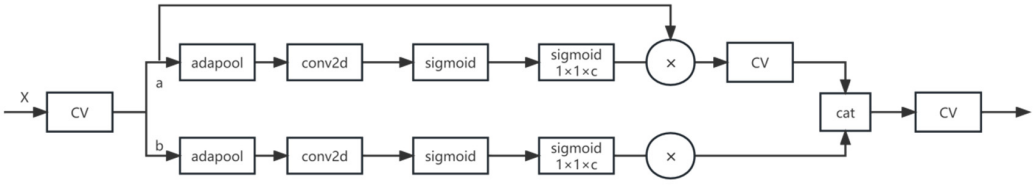

Figure 5. Mc2f module.

The output, a 1 × 1 × C feature map, leverages the weights w to discern correlations among the C channels. Opting for this methodology over the original SE's fully connected layer, we significantly decrease both the parameter count and computational load. This choice is especially advantageous for larger feature maps. Notably, the convolutional layer operation accounts for the spatial structure of the feature map—a dimension neglected by the fully connected layer—ensuring that our streamlined module retains vital spatial details.

To encapsulate, our module bifurcates the input features into two distinct channels (a and b), processing each uniquely. This differential treatment aids in amalgamating diverse channel information into a cohesive feature representation. Consequently, the model's capacity to encapsulate richer and more intricate features is amplified, making it adept at accommodating varied data distributions and intricate feature interrelations.

For a specified feature map with dimensions H × M × C (where H represents the height, M denotes the width, and C signifies the channel count), the output is a vector of dimensions 1 × 1 × C.

In the CV block, as shown in Figure 6, we draw inspiration from the bottleneck block highlighted in [23]. The module initiates with a deep convolutional layer, which serves to capture spatial attributes without adding computational complexity. This is succeeded by layer normalization, an operation that treats each feature map distinctly, aligning them to a mean of zero and a standard deviation of one. Such normalization accelerates convergence, addresses gradient vanishing issues, and bolsters model robustness. Through the application of a 1 × 1 convolution, the dimension of each feature is augmented to 4 × dim, enhancing the representational capacity of the model.

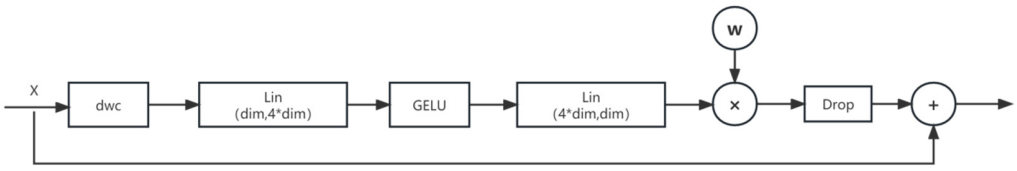

Figure 6. Cv module.

Additionally, we incorporated a non-linear layer into our architecture to bolster the model's performance, specifically opting for GELU over RELU. The GELU [16] function can be perceived as a more refined version of RELU [20], serving to amplify the network's nonlinear characteristics. It has been employed in cutting-edge transformers like Google's BERT [24] and OpenAI's GPT-2 [25]. This block primarily focuses on the nonlinear transformation and adjustment of features. It incorporates pointwise convolution and dimensionality reduction in the deep network, along with linear transformations and

activation functions in the feature dimensions. Moreover, it sporadically omits features as a strategy to mitigate overfitting. A notable capability of the module is its learnable scaling for every channel, endowing it with enhanced tuning abilities, which in turn enables more deliberate emphasis or suppression of particular features. Furthermore, inspired by the self-attention mechanism in transformer architectures [26], we introduced feature weighting to augment the learnable modulation capability of each channel's importance. Subsequently, we employed drop path [27] to randomly discard certain features, mitigating the risk of overfitting. The module finally adds the output from the drop path to the original output as per ResNet [28], aiding the network in learning the identity mapping, thereby enhancing the training stability and accelerating convergence.

In summary, this CV block is primarily a combination of deep convolution and point-wise convolution with regularization and learnable channel weights, aiming to enhance the quality of features and the training stability of the model. The detailed composition of the module is illustrated in the subsequent figure:

$$\text{GELU}\ (x) = 0.5\ \times\ (1 + \tanh(\sqrt{\frac{2}{\pi}}(x + 0.044715x^3))) \tag{11}$$

3.2. Joint Angle Calculation and Pcf Fitting

3.2.1. Joint Angle Calculation

Yolov8-sp detects 17 key points on the human body for angle estimation and pose restoration. Figure 7 shows the specific locations of the detected human keypoints. Specifically, the labels 0–17 represent the human keypoints in the following order: nose, neck, right shoulder, right elbow, right wrist, left shoulder, left elbow, left wrist, right hip, right knee, right ankle, left hip, left knee, left ankle, right eye, left eye, right ear, left ear.

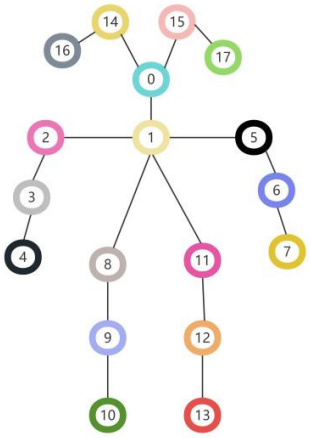

Figure 7. Seventeen keypoints of the human body.

As movements unfold, the angles formed by human limb segments vary based on the type of motion. In this study, we employ Equation (12) to extract the angular features of these movements:

$$\cos\ \theta = \frac{A \times B}{|A| \times |B|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}} \tag{12}$$

where A and B represent the lines connecting the relevant joints.

The subsequent diagram illustrates the angular relationships between each feature point, as described in Table 1. We employ a three-term calculation for comprehensive analysis:

Table 1. Key angle calculations.

Joints	Left	Right
Elbow joint	5–6–7	2–3–4
Knee joint	11–12–13	8–9–10
Shoulder joint	1–5–6	1–2–3

3.2.2. Polynomial Fitting Correction

While the YOLOv8-SP algorithm boasts impressive accuracy, challenges persist in complex scenarios, such as instances of keypoint occlusion or detection inaccuracies. Recognizing the nuances in an athlete's posture changes in sports settings, this study introduces a joint angle fitting algorithm tailored for angle correction. This algorithm focuses primarily on pivotal joints like the elbow, knee, and shoulder. This enhancement ensures a more accurate analysis of joint angle variations and facilitates a superior 3D reconstruction of the athlete's posture.

Using an iterative fitting algorithm, the module first extracts the keypoints detected by yolov8-sp and computes the desired joint angles to obtain a set of continuously varying data ø (x_i, y_i) of joint angles, where x_i is the number of corresponding key angles in the intercepted video, and y_i is the computed value of the corresponding joint angle. The true joint angle value is Y_i $(1 \leq m \leq 8)$, P_m $(1 \leq m \leq 8)$ is the number of terms of the polynomial fit, and ∂ is the fitting error of the joint angle, which is selected as the maximum distance from the true joint angle value Y_i to the fitted curve y_i. The error is calculated for different values of the joint angle. The error is calculated for different values of m, and the minimum sum of squares of the error is obtained, thus the corresponding fitted curve is not a fitted curve. ∂_{min} is obtained, thus the corresponding fitted degree P is obtained. The relevant fitting equations are as follows.

$$\partial = \sum_{i=1}^{n} (y_i - Y_i)_i^2 \tag{13}$$

$$y = \beta_0 + \beta_1 \chi + \beta_2 \chi^2 + \ldots + \beta_p \chi^p \tag{14}$$

For the angle correction values obtained after polynomial fitting, we use trigonometric functions to determine the precise position of the relevant points, as shown in Figure 8.

3.2.3. Evaluation Metrics for Fitting Corrections to Joint Angles

We classify the accuracy of joint angle detection into two metrics: the missed detection rate and the misdetection rate.

(1) Leakage rate

We employ this metric to assess the model's precision in identifying joint angles, ensuring no vital feature information is overlooked.

$$\mathrm{MR} = \frac{N_m}{N_m + N_d} \tag{15}$$

where N_d = detected node, N_m = missed node.

(2) Error detection rate

We use the average error as a metric for evaluating the critical angle and include a tolerance range τ considering that small errors are acceptable in practical critical point detection. The JAM is calculated under tolerance error:

$$\text{JAM} = 1 - \frac{\sum_{i=1}^{n} max(0, |y_i - Y_i| - \tau)}{\sum_{i=1}^{n} y_i} \quad (16)$$

where y_i is the computed joint angle, Y_i is the groundtruth, τ is the tolerance range, i is the i-th joint angle predicted, and n is the number of all joint angles (sample size).

We assessed the joint angles identified by yolov8-sp using the aforementioned metrics. The efficacy of the approach was substantiated through qualitative findings.

```
FUNCTION: EstimateRelevantPoints(x1, y1, L1, ratio,θ) -> x2, y2
    // Calculate the length of the second part based on the known length and ratio
    L2 = ratio × L1

    // Use trigonometric functions to calculate the position offset of the second joint
    Δx = L2 × COS(θ)
    Δy = L2 × SIN(θ)

    // Determine the coordinates of the second joint based on the first joint as the reference
point
    x2 = x1 + Δx    // Offset on the x-axis
    y2 = y1 + Δy    // Offset on the y-axis

    RETURN x2, y2
END FUNCTION
```

Figure 8. Estimation of relevant points.

4. Experiments and Results

In this section, experiments were conducted on a Linux server powered by an NVIDIA GTX 3090 GPU (Manufacturer: ZOTAC; City: Hong Kong; Country: China). We detail the dataset collection process, training specifics, and the performance evaluation standards. Additionally, we juxtapose yolov8-sp with contemporary cutting-edge methods using evaluation metrics to substantiate the proficiency of our proposed model. For delineating specific joint angles, we employ two metrics, MR and JAM, underscoring the efficacy of polynomial fitting in joint angle rectification.

4.1. Datasets

We gathered data on athletes' posture variations from two renowned public datasets: COCO and MPII Human Pose, as well as from videos encompassing a variety of sports activities, which included motions such as throwing, running, jumping, and hitting. All images within the dataset were annotated using Label and subsequently converted to the YOLO format for storage, as described by Glenn [17]. The dataset comprises a total of 9210 images. Recognizing the significance of variations in the elbow, knee, and shoulder joints within sporting contexts, our data collection emphasized sports postures exhibiting pronounced changes in these specific joints.

4.2. Yolov8-sp Training Details

We partitioned the SportDataset into distinct subsets: 80% (7368 images) for training, 10% (921 images) for testing, and the remaining 10% (921 images) for validation. Firstly, we adjusted the longer side of the input images to the targeted dimension, ensuring the aspect ratio remained consistent. Then, padding was added to the shorter side to form a square shape. Through this method, we guaranteed that all input images had a uniform size of 640 × 640 pixels. To enhance the network's robustness, we employed various data augmentation techniques: horizontal flipping, multi-scale adjustment (20%), random translation (2%), and random rotation (35%). These augmentations were disabled during the final 10 epochs of training, enabling us to develop a highly accurate and resilient human pose estimation model. The specific training parameters are as shown in Figure 9.

Parameter	Settings
Optimizer	SGD
Learning rate	0.01
Batch size	32
Epoch	200
Input size	640×640

Figure 9. Training parameters.

The assessment criteria followed the guidelines based on OKS. We calculated the following metrics for different threshold ranges: the AP50 (AP at oks = 0.5), the AP^{50-95} (AP at oks = 0.5 to oks = 0.95), the AP^{75} (AP at oks = 0.75). AP^{M} (AP for medium-sized individuals). AP^{L} (AP for large body size).

4.3. Comparison of Models

4.3.1. Comparison of Baseline Models

This section compares the performance of the baseline model, yolov8, with the yolov8-sp model. The baseline yolov8 model is trained using the same dataset but does not include the multicat module and the mc2f module.

Table 2 shows the comparison of the performance of the two models on the dataset. It can be seen that with the improved model, theAP^{50} aspect is significantly better than that of the baseline model.

Table 2. Performance comparison with baseline model. (The '√' symbol shows a module is active in the experiment.)

Method	Multicat	mc2f	AP^{50}	AP^{50-95}	AP^{M}	AP^{L}
(1)			85.4	59.5	60.7	74.3
(2)	√		86.4	60.1	61.6	75.6
(3)		√	86.9	60.7	62.0	74.4
(4)	√	√	87.1	61.9	61.8	75.9

Comparing with (1) and (2), we find that adding the multicat module results in an AP^{50} increase of 1.0%, an AP^{50-95} increase of 0.6%, an AP^{M} increase of 0.9% and an AP^{L} increase of 1.3%. The results validate that the multicat module extracts more effective features in the pose estimation task, which significantly improves the performance of the network and demonstrates the importance of multiscale feature fusion.

Comparing with (1) and (3), we find that adding the mc2f module results in an AP^{50} increase of 1.5% and, an AP^{50-95} increase of 1.2%, an AP^{M} increase of 1.3%, and an AP^{L}

increase of 0.1%. The results verify that the mc2f module acquires richer feature representations, better preserves spatial information, and significantly improves the performance of the network in the pose estimation task, demonstrating the importance of the attention idea in the pose estimation task.

4.3.2. Comparison with Advanced Models

We tested the results of other models against our model on SportDataset, as shown in Table 3.

Table 3. Results of comparison with other models.

Methods	Backbone	AP^{50}	AP^{75}	AP^{M}	AP^{L}
HRnet [4]	HRnet-W32	84.0	52.6	58.3	72.7
HigherHRNnet [29]	HRnet-W48	84.6	53.2	58.4	73.1
Yolov5pose [30]	Darknet-csp-d53-s	85.2	58.4	58.7	73.5
Openpose [8]	----------	83.3	53.9	57.4	71.9
Hourglass [31]	Hourglass	82.9	52.6	56.1	69.2
Lightopenpose [32]	----------	78.5	51.3	55.8	67.7
Ours	Darknet-53	87.1	60.8	61.8	75.9

4.3.3. Comparison with Openpose

In this section, we conducted a comparison with Openpose, as shown in Figure 10. The comparison was performed on four categories (occlusion, blur, multiplayer and extreme viewpoints) of challenging images and focused on the elbow, shoulder and knee joints.

It can be seen that for simple samples such as (f), similar high-quality results were obtained in this paper compared to the traditional bottom-up method Openpose. For fuzzy and multi-person samples such as (a), (c), (d), (e), the model still works reliably and significantly outperforms Openpose at the elbow, knee, and shoulder joints. This is due to the fact that we collected enough motion postures so that the network learned the relevant information about the elbow, knee, and shoulder joints to a greater extent. The multicat and mc2f modules could focus more on the detection of different dimensional features, automatically obtain the importance level of each feature channel, and use the obtained importance level to enhance the useful features and suppress the unimportant features for the current task. For complex samples such as extreme angles (b), both Openpose and our method failed in general, but our model could still detect the important parts more accurately in some extreme cases, while Openpose suffered from obvious joint misalignment.

4.4. Polynomial Fitting of Joint Angles

4.4.1. Quantitative Evaluation

We selected a tennis instruction video, randomly captured a segment of forehand and backhand serve tutorials as a test sample, and calculated the real value of the corresponding joint angle change ($0 < x < \$, $0 < y < 180$) by manually labeling the key points. The yolov8-sp was used to detect the key points of the human body and calculate the angles of left and right shoulder joints, left and right elbow joints and left and right knee joints for analysis. For example, the change in the left elbow joint of this coach in the teaching situation of forehand and backhand serve is shown in Figure 11. In this graph, the horizontal axis represents the temporal progression, ranging from 0 to 10, which precisely corresponds to a time interval of 0 to 1 s. Each unit increment on the x-axis signifies 0.1 s. The vertical axis, on the other hand, denotes the angle of the left elbow joint, spanning from 20 degrees to 180 degrees. In this manner, we can clearly observe the variation trend of the left elbow joint angle within a span of 1 s, as well as its relationship with the fitted values.

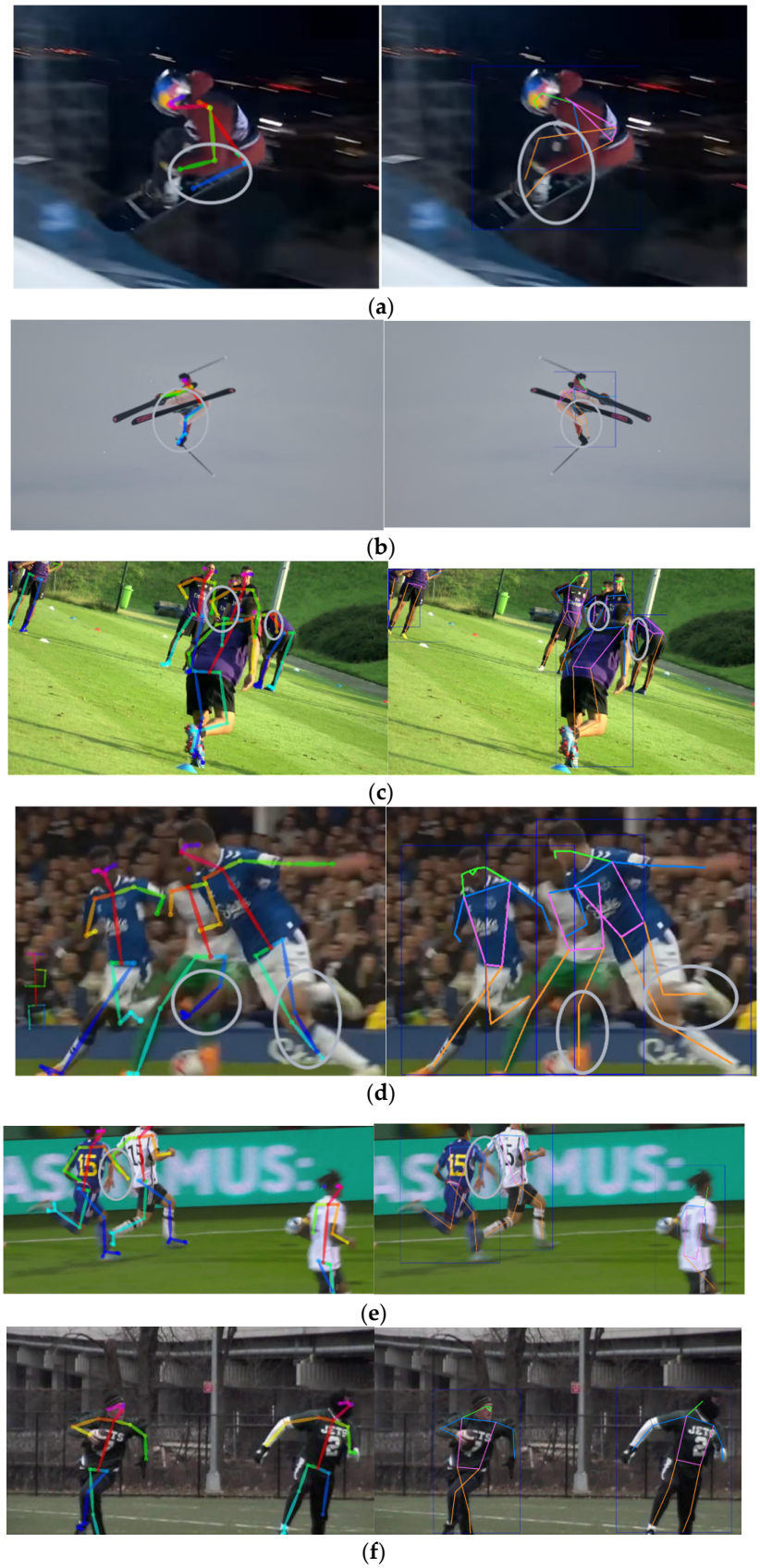

Figure 10. Qualitative results. Different performances of this paper's method and the Openpose method in various cases such as fuzzy (**a**), extreme viewpoint (**b**), multiplayer (**c**–**f**). The left column is the Openpose method, and the right column is the method of this paper.

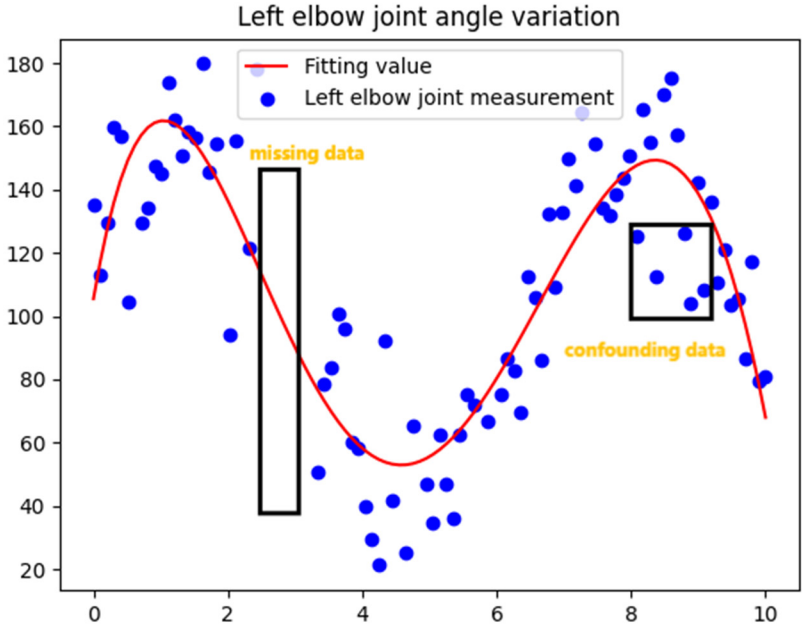

Figure 11. Fitting of the left elbow joint in tennis instruction.

In this section, we evaluated three important joint accuracies before and after fitting for five types of motion scenarios, as shown in Table 4.

Table 4. Motion scene accuracy by type.

Type	JAM^e	JAM^s	JAM^k	MR^e	MR^s	MR^k
Tennis	90.5	88.2	87.0	10.4	11.6	14.6
Football	86.1	84.4	80.2	16.7	15.5	17.3
Skiing	88.4	88.3	89.5	16.3	14.1	13.8
Gymnastics	95.5	94.0	93.6	11.3	10.2	13.4
Running	92.3	88.7	88.9	12.9	13.9	18.6

JAM is the fitted accuracy, and MR is the original missed detection rate.

The overall accuracy changes before and after fitting the three types of joints for all motion scenarios are as follows, as shown in Figure 12:

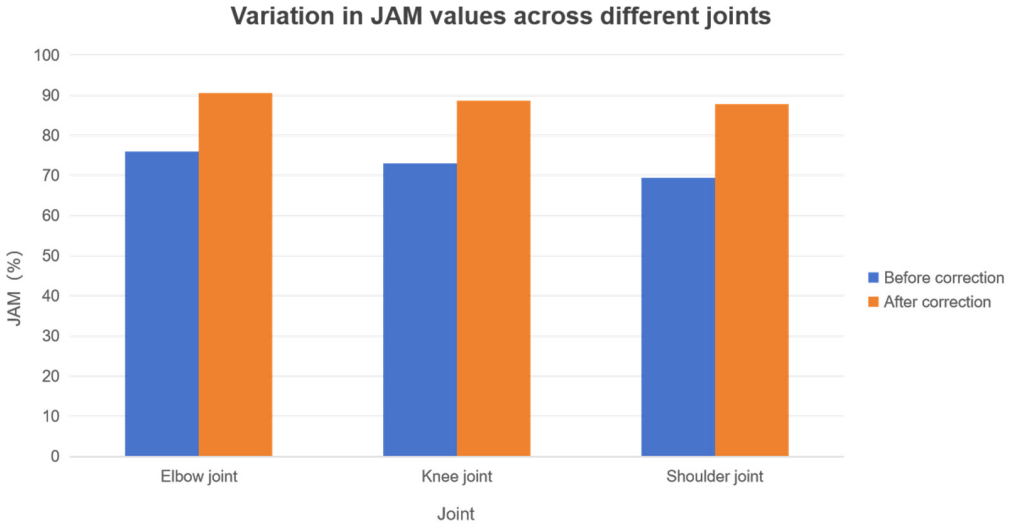

Figure 12. Precision change.

4.4.2. Qualitative Assessment of Joint Angles

We also compared the effect of corrections before and after using curve fitting in Smpl human reconstruction, as shown in Figure 13:

Figure 13. Fitting corrections in smpl human 3D reconstruction.

In this section, we further applied angle correction to a skiing video to validate the efficacy of our method with regard to knee joint angles. Specifically, in skiing, the dynamic changes of the knee joint follow certain patterns. However, during actual detection, the ski athlete's gear and certain movements might obscure the knee joint. To overcome this challenge, we took advantage of the continuous nature of joint angle variations and applied polynomial fitting to correct the continuous changes in knee joint angles throughout the process. Furthermore, we reconstructed a three-dimensional human posture model to provide a more intuitive visual representation of the research results. Upon comparison, there were notable differences in the angle distribution of the right knee before and after the fitting correction. This proves that even when faced with obstructions from ski equipment or interference from other external factors, our method can still accurately recover and estimate the actual angles of the knee joint.

The introduction of this technology holds revolutionary significance in the fields of sports biomechanics, education, and athlete technique analysis. It not only offers researchers a tool to precisely capture and analyze joint dynamics but also provides sports coaches, trainers, and relevant institutions with a practical, efficient, and precise tool for movement assessment and feedback.

5. Conclusions

In this study, we introduce a comprehensive single-stage pose estimation approach combined with an angle fitting correction technique, tailored for a wide range of motion scenarios. The effectiveness of our method is validated using a custom-built dataset specifically designed to capture targeted motion dynamics. Our improved model, yolov8-sp, builds upon the foundational architecture of yolov8. It enhances the core and feature fusion elements of the original structure and seamlessly integrates a polynomial fitting methodology to correct potential inaccuracies in joint angle estimations. This ensures precise extraction of crucial motion-related information from varied scenes.

Specifically, we begin by utilizing the single-stage pose estimation model yolov8-sp to detect 17 keypoints on the human body. Subsequently, we extract the relevant joint angle data for three critical joints during motion: the elbow joint, knee joint, and shoulder joint. To address potential inaccuracies in joint angle extraction due to occasional detection errors by yolov8-sp in extreme situations, we introduce a polynomial fitting correction. The underlying principle of this correction method is rooted in the continuous variation of joint angles during motion. By combining these two steps, we can accurately extract data from dynamic motion scenarios. These data can be employed for in-depth analysis of athletes' movements and research in various related fields. This process ensures the precision and utility of the data, providing a reliable foundation for sports research and analysis.

Significantly, our methodology offers an edge over traditional approaches by eliminating the need for multiple high-speed cameras, complex multi-angle camera arrays, or

intensive computational resources. This democratizes the application of our tool, making it a valuable asset not just for professional athletes but also for amateurs. It provides insightful feedback, paving the way for posture refinement and effective sports training using just a singular, streamlined pose estimation step, along with precise joint angle extraction and correction.

In future work, we plan to expand our motion dataset, delving deeper into capturing nuanced movements and optimizing our model's performance accordingly. Furthermore, we aim to explore the versatility of yolov8-sp by deploying it in diverse application scenarios. This research not only advances the field of pose estimation but also holds potential for transformative impacts across multiple domains, from sports to healthcare and beyond.

Author Contributions: Conceptualization, S.W., J.L. and Y.H.; Formal analysis, S.W.; Funding acquisition, X.Z. and F.M.; Investigation, S.W.; Methodology, S.W., X.Z., J.L. and Y.H.; Project administration, X.Z.; Resources, X.Z.; Supervision, X.Z., F.M., J.L. and Y.H.; Writing—original draft, S.W.; Writing—review and editing, S.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Class III Peak Discipline of Shanghai—Materials Science and Engineering (High-Energy Beam Intelligent Processing and Green Manufacturing) and National Key R&D Program of China under Grant 2020AAA0109300.

Data Availability Statement: Data are contained within the article.

Acknowledgments: We express our gratitude to the Editors and Reviewers for their valuable comments.

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

References

1. 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.
2. Chen, Y.; Wang, Z.; Peng, Y.; Zhang, Z.; Yu, G.; Sun, J. Cascaded Pyramid Network for Multi-Person Pose Estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7103–7112.
3. Tian, Z.; Chen, H.; Shen, C. DirectPose: Direct End-to-End Multi-Person Pose Estimation. *arXiv* **2019**, arXiv:1911.07451.
4. 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.
5. Xiao, B.; Wu, H.; Wei, Y. Simple Baselines for Human Pose Estimation and Tracking. In *ECCV*; Springer: Berlin/Heidelberg, Germany, 2018.
6. Ke, L.; Chang, M.-C.; Qi, H.; Lyu, S. DetposeNet: Improving Multi-Person Pose Estimation via Coarse-Pose Filtering. *IEEE Trans. Image Process* **2022**, *31*, 2782–2795. [CrossRef] [PubMed]
7. Lee, J.; Kim, T.-y.; Beak, S.; Moon, Y.; Jeong, J. Real-Time Pose Estimation Based on ResNet-50 for Rapid Safety Prevention and Accident Detection for Field Workers. *Electronics* **2023**, *12*, 3513. [CrossRef]
8. Cao, Z.; Simon, T.; Wei, S.-E.; Sheikh, Y. Realtime multi-person 2D pose estimation using part affinity fields. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7291–7299.
9. Wei, S.-E.; Ramakrishna, V.; Kanade, T.; Sheikh, Y. Convolutional Pose Machines. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 4724–4732.
10. Kocabas, M.; Karagoz, S.; Akbas, E. MultiPoseNet: Fast Multi-Person Pose Estimation Using Pose Residual Network. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 417–433.
11. Geng, Z.; Sun, K.; Xiao, B.; Zhang, Z.; Wang, J. Bottom-Up Human Pose Estimation via Disentangled Keypoint Regression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 14676–14686.
12. Brasó, G.; Kister, N.; Leal-Taixé, L. The Center of Attention: Center-Keypoint Grouping via Attention for Multi-Person Pose Estimation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 10–17 October 2021; pp. 11853–11863.
13. Wang, D.; Zhang, S.; Hua, G. Robust Pose Estimation in Crowded Scenes with Direct Pose-Level Inference. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 6278–6289.
14. Wang, L.; Su, B.; Liu, Q.; Gao, R.; Zhang, J.; Wang, G. Human Action Recognition Based on Skeleton Information and Multi-Feature Fusion. *Electronics* **2023**, *12*, 3702. [CrossRef]
15. Wu, C.; Wei, X.; Li, S.; Zhan, A. MSTPose: Learning-Enriched Visual Information with Multi-Scale Transformers for Human Pose Estimation. *Electronics* **2023**, *12*, 3244. [CrossRef]

16. Hendrycks, D.; Gimpel, K. Gaussian Error Linear Units (GELUs). *arXiv* **2016**, arXiv:1606.08415.
17. Glenn, J. Ultralytics YOLOv8. 2023. Available online: https://github.com/ultralytics/ultralytics (accessed on 2 March 2023).
18. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
19. Maji, D.; Nagori, S.; Mathew, M.; Poddar, D. YOLO-Pose: Enhancing YOLO for Multi-Person Pose Estimation Using Object Keypoint Similarity Loss. *arXiv* **2022**, arXiv:2204.06806. [CrossRef]
20. Nair, V.; Hinton, G.E. Rectified Linear Units Improve Restricted Boltzmann Machines. In Proceedings of the 27th International Conference on International Conference on Machine Learning, Haifa, Israel, 21–24 June 2010.
21. Loper, M.; Mahmood, N.; Romero, J.; Pons-Moll, G.; Black, M.J. SMPL: A Skinned Multi-Person Linear Model. *ACM Trans. Graph.* **2015**, *34*, 248. [CrossRef]
22. Lin, T.Y.; Dollár Piotr Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature Pyramid Networks for Object Detection. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
23. Liu, Z.; Mao, H.; Wu, C.Y.; Feichtenhofer, C.; Darrell, T.; Xie, S. A ConvNet for the 2020s. *arXiv* **2022**, arXiv:2201.03545.
24. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 3–5 June 2019.
25. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language Models Are Unsupervised Multitask Learners. *J. OpenAI* **2019**, *1*, 8.
26. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5998–6008.
27. Hu, H.; Zhang, Z.; Xie, Z.; Lin, S. Local relation networks for image recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October 2019–2 November 2019; pp. 3464–3473.
28. Veit, A.; Wilber, M.J.; Belongie, S. Residual networks behave like ensembles of relatively shallow networks. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016.
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 (CVPR), Seattle, WA, USA, 14–19 June 2020.
30. Redmon, J.; Divvala, S.; Girshick, R. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
31. Newell, A.; Yang, K.; Deng, J. Stacked Hourglass Networks for Human Pose Estimation. *arXiv* **2016**, arXiv:1603.06937.
32. Osokin, D. Real-time 2D Multi-Person Pose Estimation on CPU: Lightweight OpenPose. *arXiv* **2018**, arXiv:1811.12004.