This CVPR Findings paper is the Open Access version, provided by the Computer Vision Foundation. Except for this watermark, it is identical to the accepted version; the final published version of the proceedings is available on IEEE Xplore.

FSMC-Pose: Frequency and Spatial Fusion with Multiscale Self-calibration for Cattle Mounting Pose Estimation

Fangjing Li[1†] Zhihai Wang[1†] Xinxin Ding[1,2] Haiyang Liu[1*]
Ronghua Gao[2] Rong Wang[2] Yao Zhu[3*] Ming Jin[4]
[1]Beijing Jiaotong University [2]NERCITA [3]Tsinghua University [4]Griffith University

Abstract

*Mounting posture is an important visual indicator of estrus in dairy cattle. However, achieving reliable mounting pose estimation in real-world environments remains challenging due to cluttered backgrounds and frequent inter-animal occlusion. We present **FSMC-Pose**, a top-down framework that integrates a lightweight frequency–spatial fusion backbone, CattleMountNet, and a multiscale self-calibration head, SC2Head. Specifically, we design two algorithmic components for CattleMountNet: the Spatial Frequency Enhancement Block (SFEBlock) and the Receptive Aggregation Block (RABlock). SFEBlock separates cattle from cluttered backgrounds, while RABlock captures multiscale contextual information. The Spatial-Channel Self-Calibration Head (SC2Head) attends to spatial and channel dependencies and introduces a self-calibration branch to mitigate structural misalignment under inter-animal overlap. We construct a mounting dataset, MOUNT-Cattle, covering 1,176 mounting instances, which follows the COCO format and supports drop-in training across pose estimation models. Using a comprehensive dataset that combines MOUNT-Cattle with the public NWAFU-Cattle dataset, FSMC-Pose achieves higher accuracy than strong baselines, with markedly lower computational and parameter costs, while maintaining real-time inference on commodity GPUs. Extensive experiments and qualitative analyses show that FSMC-Pose effectively captures and estimates cattle mounting pose in complex and cluttered environments. Dataset and code are available at Github.*

1. introduction

Accurate estrus identification is pivotal to herd profitability and sustainability, influencing conception timing, days open, calving interval, labor cost, hormone use, and animal welfare [24]. Among behavioral indicators, mounting is the most intuitive and visually distinctive pose, characterized by forelimb lifting and hindlimb support, and it provides a critical behavioral cue for determining whether a cattle has entered estrus. If mounting pose can be measured automatically on a large scale, closed-loop decision-making in breeding, resource allocation, and health monitoring can be achieved. A reliable mounting pose estimation model would convert ubiquitous low-cost video into actionable signals, lowering dependence on skilled labor, improving reproductive efficiency, and reducing waste across diverse farm conditions. However, there is a lack of public cattle mounting datasets, resulting in the lack of research foundation for this agricultural production problem. Consequently, cattle mounting pose estimation research remains a blank.

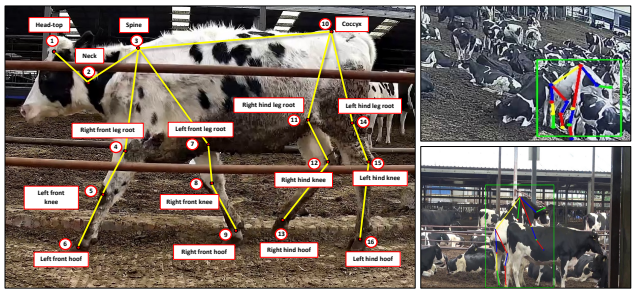


Figure 1. **Left**: keypoint annotation scheme for cattle. **Right**: real-world dense environments for cattle mounting pose estimation by FSMC-Pose, based on self-collected MOUNT-Cattle dataset.

Pose estimation [1, 9, 25, 26, 28] provides structured visual perception by extracting anatomical keypoints and spatial topology for reliable behavior recognition [18, 31, 32, 37]. Current animal pose estimation methods mainly follow bottom-up [2, 21, 33] or top-down [11, 13, 35] paradigms. Bottom-up methods such as DeepLabCut [17] effectively transfer human pose architectures to animals with limited annotations but fail under occlusion and inter-animal confusion. GANPose [30] introduces structural priors for occluded inference but is computationally expensive, while CMBN [5] reduces parameters by HRNet [29] optimiza-

*Correspondence to: Haiyang Liu <haiyangliu@bjtu.edu.cn> and Yao Zhu <ee_zhuy@zju.edu.cn>.

†Fangjing Li and Zhihai Wang contributed equally to this work.

tion yet still struggles in dense herd scenes. More importantly, agricultural production requires real-time monitoring and feedback, but the high computational cost of bottom-up approaches further limits their adoption in real-time production scenarios. Top-down methods like GRMPose [4] and T-LEAP [22] improve accuracy through lightweight backbones or temporal modeling but suffer from keypoint obfuscation and high inference complexity.

Existing approaches largely transfer human pose estimation models to animals, but the complexity of agricultural production scenes makes these methods unsuitable for real-world deployment (Figure 1). Specifically, estrous cattle tend to aggregate, making mounting scenes denser than typical farm settings. Therefore, cluttered background interference and occlusion by other cattle blur or partially remove the mounting outline; in crowded views, it is difficult to fully distinguish the individuals involved in mounting, with intertwined limbs and joints causing identity confusion; moreover, overlapping coat patterns further increase the difficulty of keypoint recognition during pose estimation. So the question of "***how to improve mounting pose estimation in dense, cluttered real herd scenes while maintaining lightweight computation?***" remains a challenge.

To address these issues, we propose **FSMC-Pose**, a frequency and spatial fusion framework with multiscale self-calibration for cattle mounting pose estimation in real-world group-housed environments. Frequency and spatial fusion uses wavelet decomposition and fixed-Gaussian smoothing to suppress clutter, enhance separability between cattle and background, and preserve fine structural detail at low contrast. Multiscale self-calibration utilizes receptive field aggregation and spatial–channel co-calibration to aggregate context across scales, correct structural shifts under inter-animal overlap, and stabilize keypoint localization for small joints and large torso regions. Extensive experiments show that FSMC-Pose accurately captures mounting postures in complex scenes and provides an effective technological foundation for intelligent estrus detection systems. Our main contributions are summarized as follows:

- We propose FSMC-Pose, a lightweight top-down framework integrating a novel backbone CattleMountNet and SC2Head for robust mounting pose estimation in real-world, group-housed dairy cattle environments.
- We construct MOUNT-Cattle dataset and combine it with NWAFU-Cattle [5] to form a comprehensive benchmark for complex mounting environments. The annotations follow the COCO [14] format and support drop-in training across pose estimation models, covering 1,176 mounting instances (Figure 1).
- Extensive experiments show FSMC-Pose surpasses strong baselines on cattle mounting pose estimation tasks, while maintaining real-time inference on commodity GPUs. Specifically, FSMCPose improves AP, AP_{75}, AR, and AR_{75} by 1.4%, 3.0%, 0.9%, and 0.4%, reaching 89%, 92.5%, 89.9%, and 97.7%, respectively. Compared with RTMPose [10], its computational cost is only 4.4109 GFLOPS, and its parameter count is reduced by 80.01% to just 2.698M.

2. Related Work

Pose estimation [1, 9, 25, 26, 34] provides structured visual representations by localizing semantic keypoints and modeling their spatial relationships. Such representations have been widely used as an effective intermediate cue for behavior analysis [18, 31, 32, 37]. With recent advances in computer vision, animal pose estimation has also achieved substantial progress. Existing methods can generally be divided into bottom-up and top-down paradigms.

2.1. Bottom-up Methods

Bottom-up methods detect all keypoints first and then group them into individual skeletons. DeepLabCut [17] shows that transfer learning can reduce annotation cost for animal pose estimation. GANPose [30] introduces adversarial learning with structural priors to improve grouped-animal pose estimation under occlusion. CMBN [5], built on HRNet [29], further improves efficiency for cattle pose estimation. However, bottom-up methods still suffer from keypoint association errors in scenes with heavy overlap and frequent occlusion.

2.2. Top-down Methods

Top-down methods first detect each animal instance and then estimate its keypoints, often achieving strong localization accuracy. GRMPose [4] combines a lightweight CSPNext backbone [16] with a graph-based coordinate classification module to balance efficiency and accuracy. T-LEAP [22] extends LEAP [20] by introducing temporal information from consecutive frames, improving robustness to occlusion and motion ambiguity. Nevertheless, top-down methods depend on detector quality, and temporal models are less suitable for single-frame inference.

Overall, although existing methods have improved animal pose estimation in complex farm environments, few are specifically designed for mounting scenarios characterized by severe inter-animal overlap and ambiguous appearance.

3. Dataset

We constructed a mounting dataset called MOUNT-Cattle, which is a mounting-centric dairy cattle pose dataset collected from real-world farms. Specifically, MOUNT-Cattle was recorded at a large commercial dairy farm in Yanqing District, Beijing, during July to August 2024, using an infrared network camera (Hikvision DS-2CD3T46WDV3-I3)

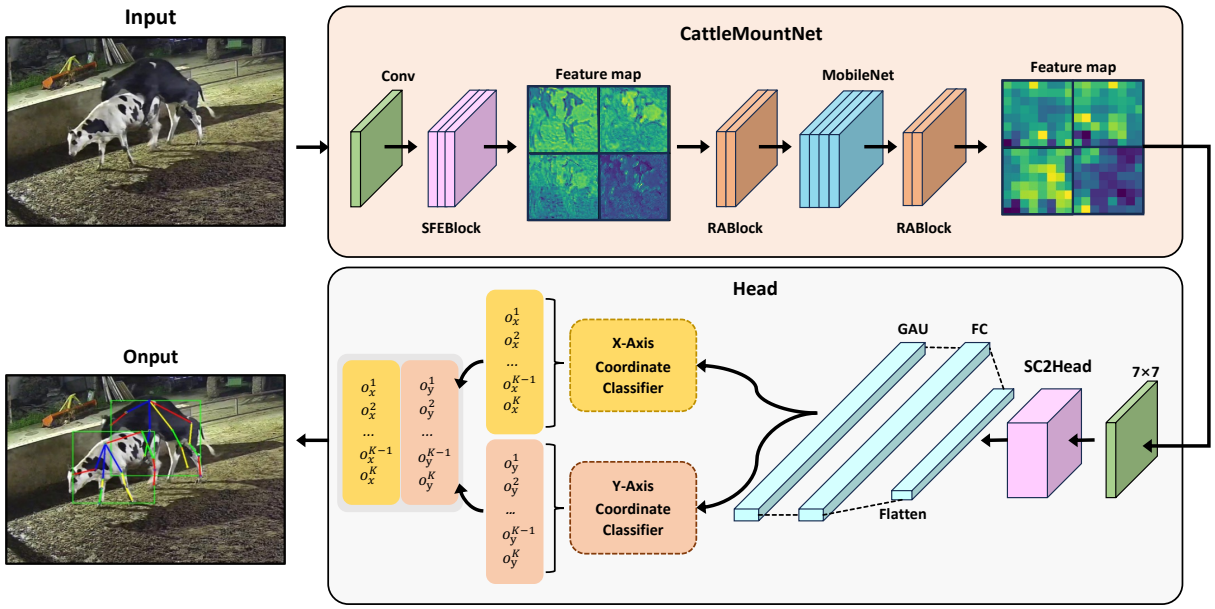


Figure 2. Architecture of FSMC-Pose framework, including the proposed lightweight backbone CattleMountNet (SFEBlock and RABlock) (Figure 3) and self-calibration SC2Head (Figure 4). Following the top-down design of RTMPose [10], and employing MobileNet [23].

and a Sony FDR-AX60 4K camcorder. The dataset focuses on mounting behavior under dense herd conditions, deliberately covering severe background clutter, similar coat patterns, and mutual occlusion, and preserving the full mounting process from initiation to termination. After manual filtering, MOUNT-Cattle contains 1,176 high-quality annotated mounting instances. We then combine our self-collected MOUNT-Cattle dataset with the public NWAFU-Cattle dataset [5], which does not include mounting behavior, to construct a comprehensive benchmark dataset.

Dataset Split. Each cattle instance in this combined dataset is labeled in COCO format [14] with a bounding box and 16 keypoints following Animal-Pose [3] and AP-10K [36], while omitting eye and mouth keypoints and adding head-top and neck to focus on whole-body pose. Keypoint visibility is categorized as invisible, partially visible, or visible, and the data are split into train/validation/test sets with an $8:1:1$ ratio. More details of MOUNT-Cattle and the combined benchmark are provided in appendix A.

Table 1. Statistics of keypoint visibility categories across dataset splits. Values are counts with proportions in parentheses.

Split	Invisible	Partial	Visible	Total
Train	7,493 (11.51%)	4,646 (7.14%)	52,965 (81.35%)	65,104
Val	929 (11.14%)	564 (6.77%)	6,843 (82.09%)	8,336
Test	1,066 (12.81%)	572 (6.88%)	6,682 (80.31%)	8,320
Total	9,488 (11.60%)	5,782 (7.07%)	66,490 (81.32%)	81,760

4. Methodology

4.1. Overview

FSMC-Pose comprises a lightweight backbone network, CattleMountNet, and an improved pose estimation head, SC2Head. As shown in Figure 2, the input image is normalized and then processed by the backbone network to extract multi-level features. For CattleMountNet, we integrate depthwise separable convolutions, residual connections, and inverted residual structures in a modular fashion to enhance the model's capability for foreground–background discrimination and scale representation.

To further feature extraction, we design two modules, SFEBlock and RABlock, from a complementary perspective. They fuse frequency and spatial domain information and by modeling multiscale receptive fields, respectively. Following feature extraction, we employ a spatial–channel self-calibration mechanism to focus the attention on critical body regions, and adopt a coordinate regression strategy for keypoint prediction. Furthermore, we introduce the SC2Head to enhance feature representations during keypoint localization based on RTMPose [10].

4.2. Lightweight Backbone: CattleMountNet

We build CattleMountNet on inverted residual structures [8, 19]: a feature map of size $H \times W \times C$ is first expanded by a 1×1 pointwise convolution, processed by a 3×3 depthwise convolution, then projected back to low dimension with another 1×1 pointwise convolution and fused

with the input. This bottleneck design preserves key information while keeping computation low. To better handle dense, cluttered group-housed cattle scenes, we introduce two modules on top of this structure: the Spatial-Frequency Enhancement Block (SFEBlock) and the Receptive Aggregation Block (RABlock). SFEBlock enhances separation between cattle and background via frequency–spatial modeling, and RABlock aggregates multiscale context to handle strong keypoint scale variation.

Spatial-Frequency Enhancement Block (SFEBlock). In real barns, mud, shadows, and lighting often make cow textures similar to the background, causing low contrast and blurred keypoints that degrade with depth. SFEBlock is designed to strengthen target–background separation while remaining lightweight, as illustrated in Figure 3.

SFEBlock combines Wavelet Transform Convolution (WTConv) [6] and Gaussian filtering. Wavelets provide multiscale frequency-domain modeling with enlarged receptive fields, while the Gaussian kernel smooths responses and suppresses background noise. Given input $F_{in} \in \mathbb{R}^{H\times W\times C}$, we first decompose it with WT and convolve each sub-band:

$$F_{WTconv} = \mathrm{IWT}\left(\mathrm{Conv}\left(W, \mathrm{WT}\left(F_{in}\right)\right)\right), \tag{1}$$

where W is the depthwise kernel for wavelet sub-bands. WT is downsampled low- and high-frequency components; small kernels on each band capture context while preserving local structure, and IWT reconstructs spatial features.

Pixels near the center receive higher weights, emphasizing salient structure and suppressing noise. We use a fixed 5×5 kernel, $F_{\text{gauss}} = G_{1.0}^{5\times5}\left(F_{WTconv}\right)$, to smooth each channel, then fuse wavelet and Gaussian features and compress them via a 1×1 convolution; element-wise multiplication and a 3×3 convolution further refine spatial responses, and a residual connection preserves the input, yielding:

$$F_{\text{temp}} = \mathrm{Conv}_{2D}^{1\times1}\left(F_{WTconv} + F_{\text{gauss}}\right), \tag{2}$$

$$F_{\text{out}} = \mathrm{Conv}_{2D}^{3\times3}\left(F_{WTconv} \otimes F_{\text{temp}}\right) + F_{in}. \tag{3}$$

This improves contrast on cattle contours while keeping computation modest.

Receptive Aggregation Block (RABlock). Cattle keyparts span small hooves and large torso or spine regions. Single-scale features cannot simultaneously capture such variation in cluttered scenes. RABlock addresses this via parallel depthwise convolutions with different dilation rates plus residual aggregation, as shown in Figure 3.

On top of the inverted residual unit, we add learnable channel-wise biases for lightweight distribution adjustment. The main branch contains three parallel 3×3 depthwise

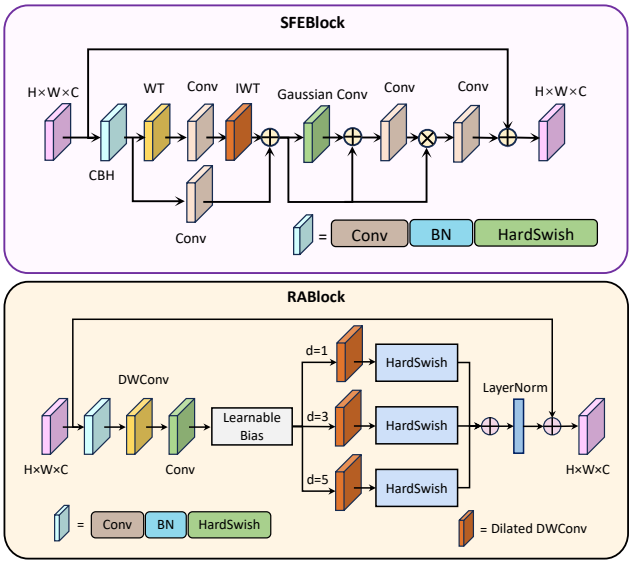


Figure 3. Architectures of the proposed CattleMountNet components: SFEBlock and RABlock.

convolutions with dilation rates 1, 3, and 5, capturing local, mid-range, and long-range context. For input $\mathrm{F}_{l-1} \in \mathbb{R}^{H\times W\times C}$ at layer L, RABlock is defined as:

$$\mathbf{H}_{l-1}^{n} = \mathrm{HardSwish}\left(\mathrm{Conv}_{3\times3}^{\mathrm{dil}=2n-1}\left(\mathrm{F}_{l-1}\right)\right), \tag{4}$$

$$\mathbf{H}_{l-1} = \mathrm{LayerNorm}\left(\mathbf{H}_{l-1}^{1} + \mathbf{H}_{l-1}^{2} + \mathbf{H}_{l-1}^{3}\right). \tag{5}$$

Depthwise convolutions keep parameters low, while HardSwish [8] provides efficient nonlinearity for mobile settings. Summing and normalizing the three paths yields a multiscale feature map that better responds to both small joints and large body structures. Residual connections in each path help preserve original structure and stabilize training under strong scale and background variation.

4.3. Multiscale Self-calibration Head: SC2Head

In group-housed mounting scenes, similar coat patterns and strong inter-cow overlap make keypoints of the same body parts spatially close and semantically ambiguous, causing structural confusion and mis-association between individuals. Our backbone with SFEBlock and RABlock improves cow–background separation and multiscale representation, but these effects mainly act in early feature extraction, and the prediction head still struggles to maintain structural consistency. To address this, we introduce SC2Head on top of RTMPose [10], as shown in Figure 4, which couples spatial and channel attention with a self-calibration branch to correct structural shifts and enhance keypoint localization.

The SC2Head consists of three branches: the Spatial Attention Branch (SAB), the Channel Attention Branch (CAB), and the Self-Calibration Branch (SCB). Given an

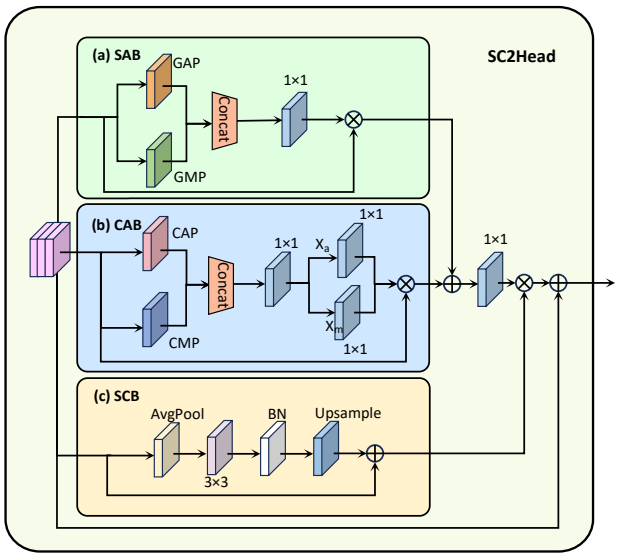


Figure 4. Architecture of the proposed SC2Head module for spatial–channel co-calibrated keypoint prediction. The improved visualization is shown in Figure 6.

input feature $\mathrm{X} \in \mathbb{R}^{H\times W\times C}$, SC2Head is defined as:

$$\mathbf{C}_o = f_{1\times1}([\mathrm{SAB}(\mathbf{X}), \mathrm{CAB}(\mathbf{X})] \odot \mathrm{SCB}(\mathbf{X})) + \mathbf{X} \quad (6)$$
$$= f_{1\times1}([\mathrm{SA}, \mathrm{CA}]) \odot \mathrm{SC} + \mathbf{X}, \quad (7)$$

where $\mathbf{C}_o \in \mathbb{R}^{H\times W\times C}$ denotes the SC2Head output, $f_{1\times1}$ represents a 1×1 convolution, $\odot$ denotes the broadcasted Hadamard product, and SA, CA, and SC are the outputs of the SAB, CAB, and SCB branches, respectively.

Spatial Attention Branch (SAB). As shown in Figure 4(a), for a given input feature, global average pooling and global max pooling are performed along the channel dimension to capture different semantic responses. The two responses are concatenated and aggregated to generate spatial attention weights, which are then multiplied with the input feature $\mathbf{X}$ to produce the output feature, expressed as:

$$\mathbf{SA} = f^{s}_{3\times3}\left(\left[f^{\mathrm{Avg}}_{\mathrm{spatial}}(\mathbf{X}), f^{\mathrm{Max}}_{\mathrm{spatial}}(\mathbf{X})\right] \odot \mathbf{X}\right), \quad (8)$$

where $f^{s}_{3\times3}$ represents a 3×3 convolution with a sigmoid activation function, and $f^{\mathrm{Avg}}_{\mathrm{spatial}}(\cdot)$ and $f^{\mathrm{Max}}_{\mathrm{spatial}}(\cdot)$ denote spatial average pooling and spatial max pooling, respectively.

Channel Attention Branch (CAB). Each channel in an image feature map typically carries distinct semantic information, and assigning different weights to channels helps the network focus on the most informative ones. As shown in Figure 4(b), for an input feature $\mathrm{X} \in \mathbb{R}^{C\times H\times W}$, channel-wise average pooling $f^{\mathrm{Avg}}_{\mathrm{channel}}(\cdot)$ and max pooling $f^{\mathrm{Max}}_{\mathrm{channel}}(\cdot)$ are applied using kernels of size (H, W). The two pooled responses, corresponding to the same feature columns, are concatenated along the channel dimension to form a shared representation:

$$\mathrm{M_c} = \left[f^{\mathrm{Avg}}_{\mathrm{channel}}(\mathbf{X}), f^{\mathrm{Max}}_{\mathrm{channel}}(\mathbf{X})\right]. \quad (9)$$

The shared feature $\mathbf{M}_\mathrm{c} \in \mathbb{R}^{2C\times H\times W}$ is processed to enable feature interaction and split equally into two branches:

$$\mathbf{X}_a, \mathbf{X}_m = \mathrm{Chunk}_2\left(\mathrm{CBL}\left(\mathbf{M}_c\right)\right), \quad (10)$$

where $\mathrm{Chunk}_2(\cdot)$ divides the tensor into two equal parts along the channel dimension, and $\mathrm{CBL}(\cdot)$ denotes a subnetwork composed of a 1×1 convolution, batch normalization (BN), and LeakyReLU activation. The channel attention map $\mathrm{CA} \in \mathbb{R}^{C\times H\times W}$ is then computed as:

$$\mathrm{CA} = \mathbf{X} \odot \mathrm{F}^{\mathrm{Avg}}_{\mathrm{channel}}\left(\mathbf{X}_a\right) \odot \mathrm{F}^{\mathrm{Max}}_{\mathrm{channel}}\left(\mathbf{X}_m\right), \quad (11)$$

where $\mathrm{F}^{\mathrm{Avg}}_{\mathrm{channel}}(\cdot)$ and $\mathrm{F}^{\mathrm{Max}}_{\mathrm{channel}}(\cdot)$ denote submodules within the channel attention mechanism, each consisting of convolution, BN, ReLU, another convolution, and Sigmoid.

Self-Calibration Branch. As illustrated in Figure 4(c), the SCB branch is designed to model contextual information effectively and establish long-range dependencies across spatial positions. For an input feature $\mathrm{X} \in \mathbb{R}^{C\times H\times W}$, the SCB computation is described as:

$$\mathbf{SC} = \delta_s\left(\mathbf{X} + \mathrm{B}_2\left(\mathrm{conv}\left(\mathrm{A}_2(\mathbf{X})\right)\right)\right), \quad (12)$$

where $\mathrm{A}_2(\cdot)$ denotes bilinear interpolation with an upsampling factor of 2, and B_2 represents average pooling with a kernel size of 2×2 and stride 2. The resulting self-calibrated feature is then concatenated with the spatial and channel attention outputs to form the final fused feature representation for keypoint prediction.

5. Experiments

5.1. Experimental Setup

Evaluation Setting. The dataset in this study follows the COCO annotation format. To evaluate the similarity between predicted and ground-truth keypoints, we adopted the Object Keypoint Similarity (OKS) metric from the COCO dataset to calculate Average Precision (AP) and Average Recall (AR). The OKS is defined as follows:

$$OKS = \frac{\sum_i \exp\left(-\frac{d_i^2}{2s^2k_i^2}\right)\delta(v_i > 0)}{\sum_i \delta(v_i > 0)} \quad (13)$$

where d_i denotes the Euclidean distance between the ground-truth and predicted keypoints, v_i indicates the visibility flag of keypoint i, s^2 represents the object scale, and

Table 2. Quantitative results across different pose estimation baselines. HigherAssociativeEmbeddingHead is abbreviated as HAEHead. AssociativeEmbeddingHead is abbreviated as AEHead.

Methods	Backbone	Head	AP/%	AP_{50}/%	AP_{75}/%	AR/%	AR_{50}/%	AR_{75}/%	FLOPs/G	Params/M
DEKR [7]	HRNet	DEKRHead	87.2	95.8	90.3	89.0	96.7	91.9	44.416	29.548
CID [25]	HRNet	CIDHead	88.0	96.4	90.8	89.0	97.1	91.7	44.160	29.363
CoupledEmbedding [27]	HRNet	AEHead	72.2	90.5	75.4	78.0	95.0	77.2	41.100	28.641
CoupledEmbedding [27]	HRNet	HAEHead	73.9	90.1	74.0	80.4	96.6	82.5	40.500	28.541
SimCC [12]	ResNet50	SimCCHead	87.4	96.0	91.0	89.9	96.7	92.9	5.493	36.753
SimCC [12]	ResNet101	SimCCHead	87.4	97.0	91.6	89.8	97.5	91.7	9.140	55.745
RTMPose [10]	CSPNext	RTMCCHead	88.6	97.0	90.6	89.0	97.5	92.7	1.926	13.550
DWPose [34]	-	-	88.3	97.0	91.5	89.8	97.3	92.1	2.200	-
RTMO [15]	CSPDarknet	RTMOHead	87.8	96.8	89.6	88.7	97.1	91.0	31.656	22.475
Ours (FSMC-Pose)	CowMountNet	SC2Head	**89.0**	**97.0**	**92.5**	**89.9**	**97.7**	**93.1**	**0.354**	**2.698**

k_i is the standard deviation corresponding to each keypoint, which varies depending on the annotation. $\delta(v_i > 0)$ equals 1 when the keypoint is visible and 0 otherwise, meaning that only visible keypoints are considered in the computation; predictions for unannotated keypoints do not affect the final results. In addition, we also evaluate the model's computational efficiency using parameters such as the total number of parameters and floating-point operations (GFLOPs).

Implementation Details. All models in this study were trained and tested on an Ubuntu 18.04 operating system using the PyTorch deep learning framework. The experiments were conducted on an NVIDIA Tesla P100 PCIe GPU with 16 GB of memory and an Intel Xeon E5-2680 v4 CPU running at 2.40 GHz. The PyTorch version used was 1.10.1 with CUDA 10.2 support. The Adam optimizer with a warm-up strategy was adopted. Considering both model complexity and dataset scale, the initial learning rate was set to 0.001, and the input image resolution was 256×192. Several data augmentation strategies were applied, including random scaling within a specified range, rotation, random horizontal shifting, and random occlusion of image regions. These augmentations introduce noise and variability into the data to prevent overfitting to specific image features or convergence to the local minima.

5.2. Experimental Results

Quantitative Results on Strong Baselines. We conducted a comprehensive evaluation of FSMC-Pose against several representative methods on the constructed dataset, including both top-down and bottom-up paradigms, to fully compare performance across different pose estimation approaches. The results are presented in Table 2, with the best results highlighted in bold. FSMC-Pose achieved the highest 89.0% AP on this dataset, surpassing representative methods such as SimCC and RTMO, and outperforming other comparison models. This demonstrates the strong accuracy advantage of FSMC-Pose in the pose estimation task. Moreover, FSMC-Pose also achieved excellent results in AP_{75}, AR_{50}, and AR_{75}, exceeding state-of-the-art methods such as SimCC and RTMPose, and significantly outperforming DEKR, CID, and RTMO. The results show that FSMC-Pose consistently extracts stable keypoint features across varying poses and scales, demonstrating strong generalization and robustness. Notably, FSMC-Pose also excels in model compactness and computational efficiency, with only 0.354M parameters and 2.698 GFLOPs. Compared with lightweight methods such as RTMPose and SimCC, it substantially reduces computational cost while maintaining high accuracy, achieving a balance between lightweight design and efficient inference.

Qualitative Results and Visualization. We qualitatively compare FSMC-Pose with representative top-down baselines (CID, SimCC, RTMPose) on both challenging test images and additional dense herd scenes, as illustrated in Figure 5. In the four groups of examples, cattle are densely packed, only a few keypoints of the mounting individual remain visible, and cloudy low-light conditions plus low camera resolution further increase difficulty. All compared models show obvious failures, including missing keypoints in heavily occluded regions and confused skeletons between overlapping individuals. In these settings, CID, SimCC, and RTMPose often exhibit missing or misplaced keypoints and confused skeleton assembly, especially when limbs of different cows overlap or when contrast is low. In contrast, FSMC-Pose consistently produces more coherent skeletons and more accurate keypoint localization. The frequency–spatial enhancement of SFEBlock and the multi-scale modeling of RABlock help separate cattle from cluttered backgrounds and capture both small joints and large torso structures, while SC2Head stabilizes predictions under heavy overlap and partial occlusion. Overall, the visual results indicate that FSMC-Pose generalizes better to complex, real-world herd environments.

Qualitative analysis of keypoint heatmaps. We further compare heatmaps across models (Figure 6). CID retains

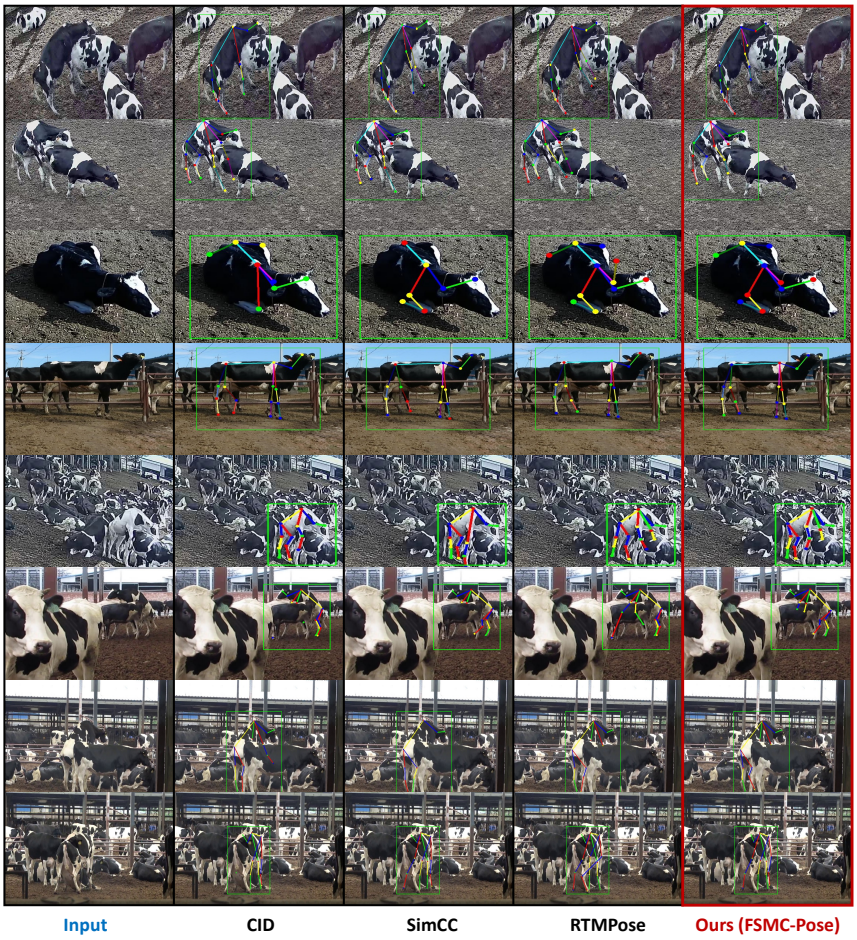


Figure 5. Qualitative comparison of mounting pose estimation results on challenging real-world scenes. FSMC-Pose produces more accurate pose than CID [25], SimCC [12], and RTMPose [10], especially under occlusion, cluttered backgrounds, and dense herd scenarios.

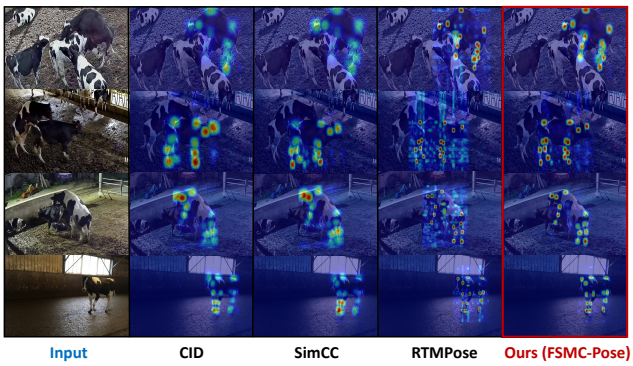


Figure 6. Qualitative comparison of predicted keypoint heatmaps under complex herd scenes. FSMC-Pose produces more concentrated and well-localized responses, especially around limbs and joints, compared with other top-down methods [10, 12, 25].

some localization ability but shows diffuse or blurred responses in cluttered areas, while SimCC struggles to separate individuals in crowded scenes, leading to displaced keypoints. FSMC-Pose generates more compact and well-aligned heatmaps, especially around limbs and joints.

5.3. Ablation Study

Effect of each module. We evaluate the effectiveness of the proposed modules using RTMPose [10] as the baseline, results are summarized in Table 3. The baseline has 13.55M parameters and strong accuracy but high deployment cost. Replacing the backbone with a lightweight MobileNet [23] reduces parameters to 1.609M (Table 3) at the expense of accuracy, suggesting the need for auxiliary modules to recover representation capacity. On top of this lightweight design, we introduce three modules (SFEBlock, RABlock, and SC2Head) for trade-off accuracy and efficiency. Individually, SFEBlock enhances edge and texture cues; RABlock enlarges the receptive field to strengthen global structure modeling; SC2Head applies spatial–channel attention that improves recall but may slightly reduce localization precision when used alone. Combinations further boost

Table 3. Ablation study of the proposed modules. We ablate SFEBlock, RABlock, and SC2Head individually and in combination. The baseline is RTMPose [10] with CSPNext backbone. The best results are highlighted in **bold**.

CattleMountNet		Head	Metrics (%)				Params/M
SFEBlock	RABlock	SC2Head	AP	AP_{75}	AR	AR_{75}	
Baseline			88.4	90.6	89.0	92.7	13.55
Baseline w/ MobileNet			87.8	89.5	89.0	91.3	1.609
✓			88.2	91.7	89.6	92.3	1.903
	✓		88.0	90.8	89.0	91.7	2.393
		✓	87.5	90.7	89.3	91.9	1.620
✓	✓		88.7	91.6	89.7	91.5	2.687
	✓	✓	88.3	91.8	89.9	91.9	2.404
✓		✓	88.6	92.1	89.8	92.1	1.914
✓	✓	✓	**89.0**	**92.5**	**89.9**	**93.1**	2.698

Table 4. Comparison of different attention mechanisms. Channel-only: CSA and ECA. Spatial-only: SAM and EMA. Joint spatial–channel: CBAM, SCAM and GCSA. Best performance is highlighted in **bold**.

Attention Mechanisms	AP/%	AP_{75}/%	AR/%	AR_{75}/%
CSA	88.5	91.4	89.4	92.1
ECA	88.1	90.3	89.3	91.0
SAM	88.4	91.6	89.1	91.3
EMA	88.1	89.0	89.2	90.5
CBAM	88.4	90.8	89.4	91.2
SCAM	88.2	90.4	89.6	91.7
GCSA	88.3	90.5	89.5	91.3
SC2Head	**89.0**	**92.5**	**89.9**	**93.1**

performance: SFEBlock+RABlock balances fine detail and global context, RABlock+SC2Head couples receptive-field expansion with attention to emphasize salient keypoints, and SFEBlock+SC2Head shows complementary gains with minimal overhead. The full model with all three modules attains the best overall accuracy and efficiency trade-off while keeping the parameter count compact.

Effect of SC2Head Attention. We evaluate SC2Head against several representative attention mechanisms, including channel-only (CSA, ECA), spatial-only (SAM, EMA), and joint spatial–channel attention (CBAM, SCAM, GCSA). To ensure a fair comparison, all attention modules are embedded in the same position as SC2Head, with identical backbone structures, training strategies, input resolutions, and datasets, and only the attention module type is replaced. The experimental results are shown in Table 4. The results show that SC2Head achieves an AP of 89.0% and an AR of 89.9%, improving AP by 0.5 over CSA and 0.6 over CBAM, and also yields the highest AR_{75} (93.1%), indicating its excellent performance under occlusion and low-contrast conditions. These gains stem from its spatial–channel co-calibration with a self-calibration branch that dynamically couples channel semantics and spatial responses, forming an efficient pathway for enhancing discriminative keypoint features.

Table 5. Comparison of inference speed. FLOPs/G, Params/M, and FPS denote computation, model size, and runtime speed, respectively. Best results are highlighted in **bold**.

Methods	FLOPs/G	Params/M	FPS
DEKR	8.328	29.548	37.57
CID	8.093	29.363	184.09
CoupledEmbedding	7.590	28.541	89.90
RTMO	31.656	22.475	78.23
Ours (FSMC-Pose)	**0.354**	**2.698**	**216.58**

5.4. Comparison of Inference Speed

To further evaluate the deployment potential of FSMC-Pose, we compare inference speed with several mainstream pose estimation methods under identical hardware and experimental conditions, as summarized in Table 5. FSMC-Pose achieves 216.58 FPS and requires only 0.354 GFLOPs and 2.698M parameters, clearly outperforming DEKR [7], CoupledEmbedding [27], RTMO [15], and the real-time oriented CID [25] in terms of both speed and model complexity. The slightly smaller than expected speed gap over CID is mainly due to the use of several complex modules with non-standard convolutions and frequent tensor reshaping, which introduce additional memory and scheduling overhead. Even so, FSMC-Pose still provides real-time inference at over 200 FPS with significantly reduced computation and parameters, making it well suited for edge devices and latency-sensitive applications where efficiency, resource consumption, and deployment cost are critical.

6. Conclusion and Discussion

In this work we address mounting pose estimation for dairy cattle in cluttered herd environments, a setting largely overlooked in animal pose estimation. We propose FSMC-Pose, a lightweight top-down framework that couples the CattleMountNet backbone with SFEBlock and RABlock for frequency–spatial fusion and multiscale aggregation, and SC2Head for spatial–channel self-calibration under inter-animal overlap. On a benchmark combining the self-collected MOUNT-Cattle and public NWAFU-Cattle dataset, FSMC-Pose outperforms baselines in AP and AR while retaining low computational cost and real-time inference on commodity GPUs. Extensive Qualitative visualizations further show that FSMC-Pose generalizes well to complex farm scenes and yields robust mounting pose estimation. In future work, we plan to extend mounting analysis to full estrus behavior pipelines by integrating temporal cues and multi-camera views and to explore large-scale deployment in precision livestock farming.

Acknowledgement

This work was supported by the Beijing Natural Science Foundation (No. 4242037), the Youth Project of MOE (Ministry of Education) Foundation on Humanities and Social Sciences (No. 23YJCZH223), and the National Natural Science Foundation of China (No. 72501020, U2568225).

References

[1] Xiaoqi An, Lin Zhao, Chen Gong, Nannan Wang, Di Wang, and Jian Yang. Sharpose: Sparse high-resolution representation for human pose estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 691–699, 2024.

[2] Bruno Artacho and Andreas Savakis. Bapose: Bottom-up pose estimation with disentangled waterfall representations. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 528–537, 2023.

[3] Jinkun Cao, Hongyang Tang, Hao-Shu Fang, Xiaoyong Shen, Cewu Lu, and Yu-Wing Tai. Cross-domain adaptation for animal pose estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9498–9507, 2019.

[4] Ling Chen, Lianyue Zhang, Jinglei Tang, Chao Tang, Rui An, Ruizi Han, and Yiyang Zhang. Grmpose: Gcn-based real-time dairy goat pose estimation. *Computers and Electronics in Agriculture*, 218:108662, 2024.

[5] Qingcheng Fan, Sicong Liu, Shuqin Li, and Chunjiang Zhao. Bottom-up cattle pose estimation via concise multi-branch network. *Computers and Electronics in Agriculture*, 211: 107945, 2023.

[6] Shahaf E Finder, Roy Amoyal, Eran Treister, and Oren Freifeld. Wavelet convolutions for large receptive fields. In *European Conference on Computer Vision*, pages 363–380. Springer, 2024.

[7] Zigang Geng, Ke Sun, Bin Xiao, Zhaoxiang Zhang, and Jingdong Wang. Bottom-up human pose estimation via disentangled keypoint regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14676–14686, 2021.

[8] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1314–1324, 2019.

[9] Shengxiang Hu, Huaijiang Sun, Dong Wei, Xiaoning Sun, and Jin Wang. Continuous heatmap regression for pose estimation via implicit neural representation. *Advances in Neural Information Processing Systems*, 37:102036–102055, 2024.

[10] Tao Jiang, Peng Lu, Li Zhang, Ningsheng Ma, Rui Han, Chengqi Lyu, Yining Li, and Kai Chen. Rtmpose: Real-time multi-person pose estimation based on mmpose. *arXiv preprint arXiv:2303.07399*, 2023.

[11] Rawal Khirodkar, Visesh Chari, Amit Agrawal, and Ambrish Tyagi. Multi-instance pose networks: Rethinking top-down pose estimation. In *Proceedings of the IEEE/CVF International conference on computer vision*, pages 3122–3131, 2021.

[12] Yanjie Li, Sen Yang, Peidong Liu, Shoukui Zhang, Yunxiao Wang, Zhicheng Wang, Wankou Yang, and Shu-Tao Xia. Simcc: A simple coordinate classification perspective for human pose estimation. In *European Conference on Computer Vision*, pages 89–106. Springer, 2022.

[13] Zhihao Li, Jianzhuang Liu, Zhensong Zhang, Songcen Xu, and Youliang Yan. Cliff: Carrying location information in full frames into human pose and shape estimation. In *European Conference on Computer Vision*, pages 590–606. Springer, 2022.

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[15] Peng Lu, Tao Jiang, Yining Li, Xiangtai Li, Kai Chen, and Wenming Yang. Rtmo: Towards high-performance one-stage real-time multi-person pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1491–1500, 2024.

[16] Chengqi Lyu, Wenwei Zhang, Haian Huang, Yue Zhou, Yudong Wang, Yanyi Liu, Shilong Zhang, and Kai Chen. Rtmdet: An empirical study of designing real-time object detectors. *arXiv preprint arXiv:2212.07784*, 2022.

[17] Alexander Mathis, Pranav Mamidanna, Kevin M Cury, Taiga Abe, Venkatesh N Murthy, Mackenzie Weygandt Mathis, and Matthias Bethge. Deeplabcut: markerless pose estimation of user-defined body parts with deep learning. *Nature neuroscience*, 21(9):1281–1289, 2018.

[18] Faisal Mehmood, Xin Guo, Enqing Chen, Muhammad Azeem Akbar, Arif Ali Khan, and Sami Ullah. Extended multi-stream temporal-attention module for skeleton-based human action recognition (har). *Computers in Human Behavior*, 163:108482, 2025.

[19] Sachin Mehta and Mohammad Rastegari. Separable self-attention for mobile vision transformers. *arXiv preprint arXiv:2206.02680*, 2022.

[20] Talmo D Pereira, Diego E Aldarondo, Lindsay Willmore, Mikhail Kislin, Samuel S-H Wang, Mala Murthy, and Joshua W Shaevitz. Fast animal pose estimation using deep neural networks. *Nature methods*, 16(1):117–125, 2019.

[21] Haoxuan Qu, Yujun Cai, Lin Geng Foo, Ajay Kumar, and Jun Liu. A characteristic function-based method for bottom-up human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13009–13018, 2023.

[22] Helena Russello, Rik van der Tol, and Gert Kootstra. T-leap: Occlusion-robust pose estimation of walking cows using temporal information. *Computers and Electronics in Agriculture*, 192:106559, 2022.

[23] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[24] JH Van Vliet and FJCM Van Eerdenburg. Sexual activities and oestrus detection in lactating holstein cows. *Applied Animal Behaviour Science*, 50(1):57–69, 1996.

[25] Dongkai Wang and Shiliang Zhang. Contextual instance decoupling for robust multi-person pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11060–11068, 2022.

[26] Dongkai Wang and Shiliang Zhang. Spatial-aware regression for keypoint localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 624–633, 2024.

[27] Haixin Wang, Lu Zhou, Yingying Chen, Ming Tang, and Jinqiao Wang. Regularizing vector embedding in bottom-up human pose estimation. In *European conference on computer vision*, pages 107–122. Springer, 2022.

[28] Haonan Wang, Jie Liu, Jie Tang, Gangshan Wu, Bo Xu, Yanbing Chou, and Yong Wang. Gtpt: Group-based token pruning transformer for efficient human pose estimation. In *European Conference on Computer Vision*, pages 213–230. Springer, 2024.

[29] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3349–3364, 2020.

[30] Zehua Wang, Suyin Zhou, Ping Yin, Aijun Xu, and Junhua Ye. Ganpose: Pose estimation of grouped pigs using a generative adversarial network. *Computers and Electronics in Agriculture*, 212:108119, 2023.

[31] Jianyang Xie, Yanda Meng, Yitian Zhao, Anh Nguyen, Xiaoyun Yang, and Yalin Zheng. Dynamic semantic-based spatial graph convolution network for skeleton-based human action recognition. In *Proceedings of the AAAI conference on artificial intelligence*, pages 6225–6233, 2024.

[32] Haojun Xu, Yan Gao, Zheng Hui, Jie Li, and Xinbo Gao. Language knowledge-assisted representation learning for skeleton-based action recognition. *IEEE Transactions on Multimedia*, 2025.

[33] Jie Yang, Ailing Zeng, Ruimao Zhang, and Lei Zhang. X-pose: Detecting any keypoints. In *European Conference on Computer Vision*, pages 249–268. Springer, 2024.

[34] Zhendong Yang, Ailing Zeng, Chun Yuan, and Yu Li. Effective whole-body pose estimation with two-stages distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4210–4220, 2023.

[35] Rui Yin, Yulun Zhang, Zherong Pan, Jianjun Zhu, Cheng Wang, and Biao Jia. Srpose: Two-view relative pose estimation with sparse keypoints. In *European Conference on Computer Vision*, pages 88–107. Springer, 2024.

[36] Hang Yu, Yufei Xu, Jing Zhang, Wei Zhao, Ziyu Guan, and Dacheng Tao. Ap-10k: A benchmark for animal pose estimation in the wild. *arXiv preprint arXiv:2108.12617*, 2021.

[37] Yuxuan Zhou, Xudong Yan, Zhi-Qi Cheng, Yan Yan, Qi Dai, and Xian-Sheng Hua. Blockgcn: Redefine topology awareness for skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2049–2058, 2024.