Keypoints as Dynamic Centroids for Unified Human Pose and Segmentation

Niaz Ahmad[1] , **Jawad Khan**[2] , **Kang G. Shin**[3] , **Youngmoon Lee**[4*] and **Guanghui Wang**[1]
[1]Department of Computer Science, Toronto Metropolitan University, Canada
[2] School of Computing, Gachon University, Republic of Korea,
[3]Department of Electrical Engineering and Computer Science, University of Michigan, USA
[4] Department of Robotics, Hanyang University, Republic of Korea
{niazahmad89, wangcs}@torontomu.ca, jkhanbk1@gachon.ac.kr, kgshin@umich.edu,
youngmoonlee@hanyang.ac.kr

Abstract

The dynamic movement of the human body presents a fundamental challenge for human pose estimation and body segmentation. State-of-the-art approaches primarily rely on combining keypoint heatmaps with segmentation masks but often struggle in scenarios involving overlapping joints during pose estimation or rapidly changing poses for instance-level segmentation. To address these limitations, we leverage *Keypoints as Dynamic Centroid* (KDC), a new centroid-based representation for unified human pose estimation and instance-level segmentation. KDC adopts a bottom-up paradigm to generate keypoint heatmaps for easily distinguishable and complex keypoints, and improves keypoint detection and confidence scores by introducing KeyCentroids using a keypoint disk. It leverages high-confidence keypoints as dynamic centroids in the embedding space to generate MaskCentroids, allowing for the swift clustering of pixels to specific human instances during rapid changes in human body movements in a live environment. Our experimental evaluations focus on crowded and occluded cases using the CrowdPose, OCHuman, and COCO benchmarks, demonstrating KDC's effectiveness and generalizability in challenging scenarios in terms of both accuracy and runtime performance. Our implementation is available at: https://sites.google.com/view/niazahmad/projects/kdc.

1 Introduction

Human pose estimation and body segmentation are crucial for human-computer interaction and real-time visual human analysis. The primary objective is to identify individuals and their activities from 2D joint positions and body shapes. The underlying main challenges include handling an unknown number of overlapping, occluded, or entangled individuals and managing the rapidly increasing computational complexity as the number of individuals grows [Han *et al.*, 2025]. Human-to-human interactions further complicate spatial associations due to limb contact and obstructions, necessitating an efficient, scalable, and accurate unified model for human pose and segmentation.

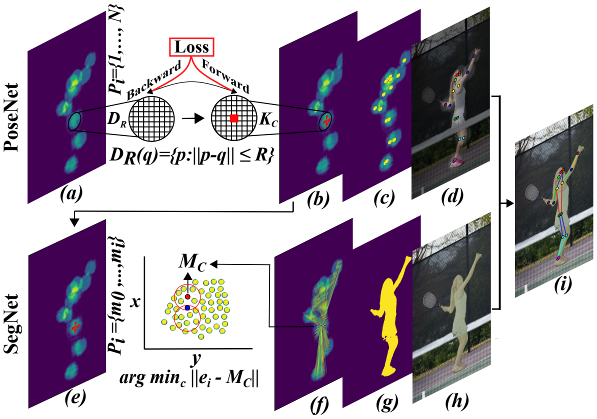


Figure 1: PoseNet operation begins by generating keypoint heatmaps in the feature space using a disk representation D_R to identify potential keypoint locations. It then introduces KeyCentroid to refine these keypoint coordinates to improve accuracy. SegNet leverages the KeyCentroid K_c defined by PoseNet to establish MaskCentroid M_c, which is essential for clustering mask pixels corresponding to specific human instances.

In this paper, we propose KDC, a new centroid-based unified representation for human pose estimation and instance-level segmentation. It first detects individual keypoints in a bottom-up manner and then employs high-confidence keypoints as dynamic centroids for mask pixels to perform instance-level segmentation. Unlike top-down approaches [Chen *et al.*, 2018; Fang *et al.*, 2017; Huang *et al.*, 2017], KDC detects humans without requiring a box detector or incurring runtime complexity.

KDC is not the first to leverage bottom-up approach [George *et al.*, 2018; Dantone *et al.*, 2013; He *et al.*, 2017; Zhang *et al.*, 2019; Ahmad *et al.*, 2025]. However, the model in [George *et al.*, 2018] employs human poses to refine pixel-wise clustering for segmentation, and thus does not perform segmentation well in segmentation tasks. Other models suffer from the computational overhead of a person detector [He *et al.*, 2017], the scalability problem for instance-level segmen-

[*]Youngmoon Lee is the corresponding author

tation [Zhang *et al.*, 2019], and model complexity [Dantone *et al.*, 2013], making them unsuitable for crowded scenarios and real-time applications. Unlike these models, KDC avoids the computational overhead of a person detector and suffers from neither the degraded segmentation performance nor the scalability problem of pixel-wise clustering.

KDC overcomes these problems using two primary networks: *PoseNet*, which generates keypoints, and *SegNet*, which produces segmentation masks using high-confidence keypoints (Fig. 1). PoseNet creates keypoint heatmaps using a keypoint disk representation that estimates the relative displacement between pairs of keypoints, enhancing the precision of long-range, occluded, and proximate keypoints (Fig. 1a). A KeyCentroid is defined for each keypoint heatmap locus offset vectors to the centroid of each keypoint disk, helping KDC identify the precise human keypoint coordinates (Fig. 1b). Additionally, KDC calculates the keypoint confidence score using the precise keypoint coordinates (Fig. 1c), with the final predicted keypoints illustrated in (Fig. 1d).

Meanwhile, SegNet performs pixel-level classification using dynamic high-confident keypoints as MaskCentroids (Fig. 1e). MaskCentroid defines an embedding space that associates pixels with the correct instance (Fig. 1f) and generates high-level semantic maps (Fig. 1g). Leveraging these semantic maps, the system produces instance-level segmentation (Fig. 1h). The PoseSeg module combines high-level features from both PoseNet and SegNet to provide a unified representation of human pose and instance-level segmentation (Fig. 1i).

We evaluated the performance of the KDC using the CrowdPose [Li *et al.*, 2019], OCHuman [Zhang *et al.*, 2019], and COCO [Lin *et al.*, 2014] benchmarks. To the best of our knowledge, KDC is the first real-time model with reliable performance to offer a unified representation of human pose estimation and instance-level segmentation. This paper makes the following contributions.

- The development of KeyCentroid, a novel method that directs keypoint vectors towards the centroid within the keypoint disk. This approach helps identify the precise keypoint coordinates in human pose estimation, thereby enhancing confidence in the results (§3.2).
- The developement of MaskCentroid leverages high-confidence keypoints as dynamic centroids for mask vectors in the embedding space. This approach effectively associates pixels with the correct instance, even during rapid changes in human body movements (§3.3).
- An in-depth evaluation (§4) and ablation studies (§5) demonstrate the effectiveness of the unified representation of human pose and instance-level segmentation.

2 Related Work

Human Pose Estimation. Approaches for human pose estimation can be classified as top-down or bottom-up. The top-down approach first runs a human detector and then identifies keypoints. Representative works include HRNet [Cheng *et al.*, 2020], RMPE [Fang *et al.*, 2017], Multiposenet [Kocabas *et al.*, 2018], convolutional pose machine [Wei *et al.*, 2016], CPN [Chen *et al.*, 2018], Mask r-cnn [He *et al.*, 2017], simple baseline [Xiao *et al.*, 2018], CSM-SCARB [Su *et al.*, 2019], RSN [Cai *et al.*, 2020], and Graph-PCNN [Wang *et al.*, 2020a]. The top-down approach explores the human pose in a person detector, thus achieving a satisfactory performance, but it is computationally expensive. The bottom-up approach like DeepCut [Pishchulin *et al.*, 2016] and DeeperCut [Insafutdinov *et al.*, 2016], unlike the top-down counterpart, detects the keypoints in a one-shot manner. It formulates the association between keypoints as an integer linear scheme which takes a longer processing time. Part-affinity field techniques like OpenPose [Cao *et al.*, 2017] and other extensions, such as PersonLab [George *et al.*, 2018], and HGG [Jin *et al.*, 2020] have been developed based on grouping techniques that often fail in crowd. KDC aims to specifically improve hard keypoint detection in crowded and occluded cases by introducing the keypoint heatmaps using keypoint disks and KeyCentroid.

Instance-level Segmentation. Instance-level segmentation is done in either single-stage [Dai *et al.*, 2016; Long *et al.*, 2015; Bolya *et al.*, 2019] or multi-stage [He *et al.*, 2017; Ren *et al.*, 2015]. The single-stage approach generates intermediate and distributed feature maps based on the input image. InstanceFCN [Dai *et al.*, 2016] produces instance-sensitive scoring maps and applies the assembly module to the output instance. This approach is based on repooling and other non-trivial computations (e.g., mask voting), which is not suitable for real-time processing. YOLACT [Bolya *et al.*, 2019] runs a set of mask prototypes and uses coefficient masks, but this method is critical to obtain a high-resolution output. The multi-stage approach follows the detect-then-segment paradigm. It first performs box detection, and then pixels are classified to obtain the final mask in the box region. Mask R-CNN [He *et al.*, 2017] is based on multi-stage instance segmentation that extends Faster R-CNN [Ren *et al.*, 2015] by adding a branch for predicting segmentation masks for each Region of Interest. The subsequent work in [Liu *et al.*, 2018] improves the accuracy of Mask R-CNN by enriching the Feature Pyramid Network [Lin *et al.*, 2017]. In contrast, our segmentation pipeline introduces MaskCentroid, a dynamic clustering point that helps cluster the mask pixels to a particular instance under the rapid changes in human-body movements.

Joint Human Pose and Instance-level Segmentation. In the line of multi-task learning paradigm, joint pose estimation and instance-level segmentation have received significant attention in recent years. Mask R-CNN [He *et al.*, 2017] was the first pioneer method, but it suffers from high computational costs due to its top-down nature. PersonLab [George *et al.*, 2018] and Pose*Plus*Seg [Dantone *et al.*, 2013] are closest to KDC. Both of them can be considered as end-to-end joint pose and instance-level segmentation models that use a bottom-up approach. However, there are several major differences that make KDC more effective, scalable, and real-time. First, they rely on static features to detect or group keypoints by using greedy decoding; in contrast, KDC introduces KeyCentroid that calculates the optimal keypoint coordinates, and

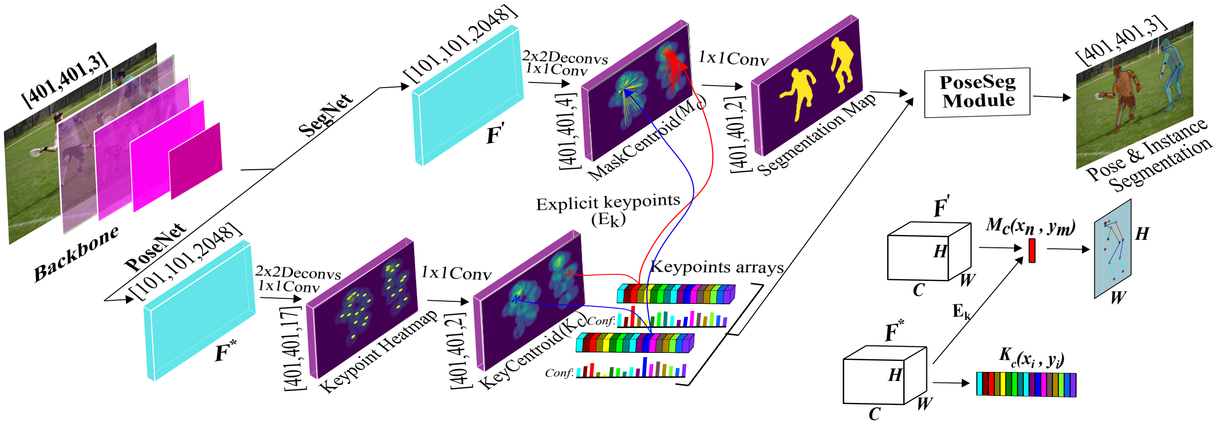


Figure 2: The overview of the proposed KDC model. PoseNet generates keypoint heatmaps and refines them with KeyCentroid K_c, improving keypoint accuracy. SegNet uses K_c to create MaskCentroid M_c, clustering mask pixels for precise instance segmentation. The PoseSeg module integrates these outputs, resulting in accurate unified human pose estimation and instance-level segmentation.

uses MaskCentroid, a dynamic clustering point for instance-level segmentation. Second, their segmentation does not perform well on highly entangled instances due to part-induced geometric embedding descriptors. Finally, they involve the complex structure model with a couple of refined networks, making them infeasible for real-time purposes.

3 Technical Approach

3.1 Keypoint Heatmap using Disk Representation

KDC generates keypoint heatmaps using disk representation (KHDR) through PoseNet, forming the foundation for human pose estimation (Fig. 3a). In this phase, individual keypoints are detected and aggregated in the output feature maps. We adopt a residual-based network for a multi-person pose setting to produce keypoint heatmaps—one channel per keypoint—and KeyCentroid, with two channels per keypoint for vertical and horizontal displacement within the keypoint disk.

The keypoint prediction approach is as follows: Let p_i represent the keypoint position in the image, where $i \in \{1, \ldots, N\}$ corresponds to the 2D positions of the pixels. A keypoint disk $D_R(q) = \{p : \|p - q\| \leq R\}$ of radius R is focused at point q, centered in the disk. Similarly, $q_{j,k}$ signifies the 2D position of the jth keypoint of the kth person instance, where $j \in \{1, \ldots, I\}$ and I is the number of individual keypoints in the image. A binary classification approach is followed for each known keypoint j. Specifically, every predicted keypoint pixel p_i is binary classified such that $p_i = 1$ if $p_i \in D_R$ for each person keypoint j; otherwise, $p_i = 0$. Independent dense binary classification tasks are performed for each keypoint, leading to distinct keypoint maps.

During the training process, the heatmap loss is computed using the binary cross-entropy (logistic loss) function defined as:

$$\mathcal{L}_{\text{heatmap}} = -\frac{1}{N} \sum_{i=1}^{N} \left[y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i) \right], \tag{1}$$

where N is the total number of pixels, y_i is the true binary label for pixel p_i, and $\hat{y}_i$ is the predicted probability that pixel p_i belongs to the keypoint. This loss function measures the difference between the predicted probability and the true label, and the average loss across all pixels in the heatmap is used to train the model. Back-propagation is performed throughout the entire image, except for regions that encompass individuals lacking comprehensive keypoint annotations (e.g., crowded and small-scale person segments).

Point-wise Gaussian Optimization. To obtain optimal keypoint coordinates, we apply a Gaussian smoothing technique [Chung, 2020] for each individual keypoint, referred to as *point-wise Gaussian optimization*. This approach effectively reduces noise while preserving valuable information, producing the keypoint heatmap as:

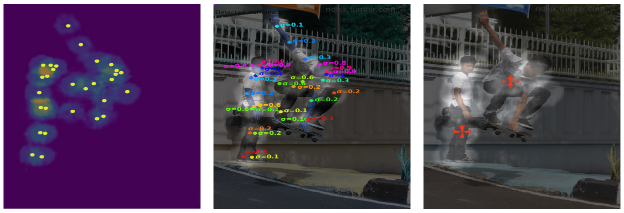
$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\dfrac{x^2 + y^2}{2\sigma^2}}, \tag{2}$$

where $G(x, y)$ is the Gaussian kernel, σ is the standard deviation of the distribution, and x and y represent the 2D keypoint coordinates. We define the σ range from 0.1 to 1 to accommodate variations among keypoints. For high-variance keypoints (HVK) such as the wrist, ankle, elbow, and knee, we set $0.1 \leq \sigma < 0.5$. Conversely, for low-variance keypoints (LVK) like the nose, shoulder, and hip, we set $0.5 \leq \sigma < 1$, as depicted in Fig. 3b.

A smaller σ value, close to 0.1, intensifies pixel values of keypoints, proving effective in congested and intricate scenarios. In contrast, a larger σ value, close to 1, yields optimal results in less crowded cases. Our analysis investigates how the σ value impacts system performance in ablation studies (§5.3).

3.2 KeyCentroid

In addition to keypoint heatmaps, our PoseNet, in conjunction with the residual network, introduces KeyCentroid k_c for each keypoint as shown in Fig. 2. The objective of KeyCentroid is to improve both the accuracy of keypoint localization and the confidence scores.



(a) Keypoint heatmap (b) PGO (c) KeyCentroid

Figure 3: (a) presents Keypoint heatmap using keypoint disk, (b) shows Point-wise Gaussian optimization (PGO) where σ values are defined for each keypoint (c) Indicates KeyCentroid defined for the right knee using the keypoint disk.

For each keypoint pixel p_i within the disk D_R, the 2D KeyCentroid vector $k_v = q_{j,k} - p_i$ originates from the pixel position p_i and points to the j^{th} keypoint of the k^{th} person instance, as illustrated in Fig. 3c. We generate a vector field within D_R by solving a 2D regression problem for the j^{th} keypoint with spatial coordinates (x_j, y_j), and compute its response on the ground truth feature map F_j^* as:

$$F_j^*(x, y) = \exp\left(-\frac{(x - x_j)^2 + (y - y_j)^2}{2\sigma^2}\right), \quad (3)$$

where σ^2 is the variance related to the disk radius $R = 32$, used to normalize the KeyCentroid and align its dynamic range with the keypoint heatmap loss.

During training, we penalize the KeyCentroid error using the L1 loss function, which is defined as:

$$\mathcal{L}_{\text{KeyCentroid}} = \frac{1}{N} \sum_{i=1}^{N} \|k_{v,i} - \hat{k}_{v,i}\|_1, \quad (4)$$

where N is the number of pixels in the disk D_R, $k_{v,i}$ is the ground truth KeyCentroid vector for pixel p_i, and $\hat{k}_{v,i}$ is the predicted KeyCentroid vector. This loss function measures the difference between the predicted and true KeyCentroid vectors, and the average loss across all pixels in the disk is used to train the model.

The error is back-propagated for each pixel $p_i \in D_R$. We then aggregate the keypoint heatmap and KeyCentroid to determine the optimal keypoint coordinates (x_j, y_j), which improves the detection of both easily distinguishable and challenging keypoints. Our ablation experiments examine the impact of our uniquely designed KHDR and KeyCentroid on keypoint detection (§5.1).

3.3 MaskCentroid

Instance-level segmentation is a pixel classification problem focused on allocating pixels to the correct instance. We introduce MaskCentroid C_i (a dynamic high-confidence keypoint), as illustrated in Fig. 4a. Our mechanism clusters mask pixels using the defined centroid C_i inside each annotated person, pointing from the image position x_i to the centroid C_i of the corresponding instance. At each semantically identified human instance, the pixel embedding $e(x_i)$ reflects a local approximation of each mask pixel's absolute location relative to the individual it pertains to, effectively capturing the anticipated 2D structure of the human body.

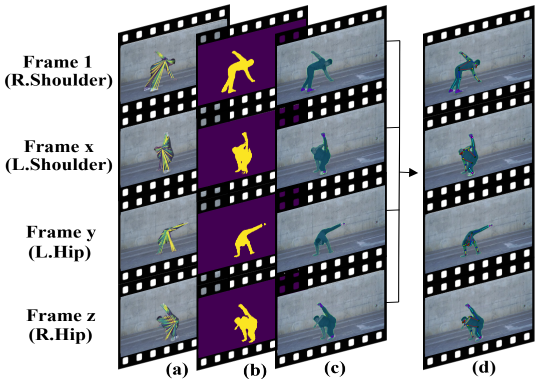


Figure 4: (a) Introduces MaskCentroid a dynamic high confident keypoint; (b) presents a precise segmentation map; (c) indicates instance-level segmentation; and (d) shows unified representation of human pose and estimation.

Consequently, for every pixel, we determine pixel offsets pointing to C_i. Each C_i serves as a high-confidence keypoint that can change with the rapid variation in keypoints, as shown in Fig. 4a. The objective of human-body segmentation is to assign a set of pixels $P_i = \{m_0, m_1, m_2, \ldots, m_i\}$ and its 2D embedding vectors $e(m_i)$ into a set of instances $I = \{N_0, N_1, N_2, \ldots, N_j\}$ to generate a 2D mask for each human instance, as shown in Fig. 4b. Pixels are clustered to their corresponding centroid $C_i = \frac{1}{N}\sum_{m_i \in N_j} m_i$. This is achieved by defining a pixel offset vector v_i for each known pixel m_i, so that the resulting embedding $e_i = m_i + v_i$ points from its respective instance centroid. We penalize pixel offset loss using the L1 loss function during model training:

$$\mathcal{L}_{\text{offset}} = \frac{1}{N} \sum_{i=1}^{N} \|e_i - (m_i + v_i)\|_1. \quad (5)$$

To cluster the pixels to their centroid, it is important to specify the positions of the instance centroids and assign pixels to a particular instance centroid. We employ a Gaussian function $\phi_j(e_i)$ for each instance N_j, which converts the distance between a pixel embedding $e_i = m_i + v_i$ and the instance centroid C_i into a probability of belonging to that instance:

$$\phi_j(e_i) = \exp\left(-\frac{\|e_i - C_i\|^2}{2\sigma_j^2}\right). \quad (6)$$

Dynamic Center of Attraction. A significant innovation has been introduced in SegNet over the state-of-the-art [Dantone *et al.*, 2013], as shown in Figure 2. The previous model relied on a fixed centroid as a parameter to cluster mask pixels, which could lead to inferior results if the centroid is occluded in real-time scenarios. However, we allow the network to learn the optimal center of attraction by introducing the concept of a dynamic centroid. This is achieved by defining the high-confidence keypoint as a learnable parameter.

This approach is especially valuable in scenarios where rapid occlusions occur during real-time operations, allowing the network to dynamically adjust the learned parameter and modify the center of attraction. As a result, the network can influence the location of the center of attraction by altering the embedding positions.

$$\phi_j(e_i) = \exp\left(-\frac{\|e_i - \left(\frac{1}{|N_j|}\sum_{e_j \in N_j} e_j\right)\|^2}{2\sigma_j^2}\right). \quad (7)$$

In the inference phase, using keypoints as dynamic centroids for mask pixels effectively addresses complex scenarios where over 70% of the human body is occluded. Our experimental study analyzes the effectiveness of both Static MaskCentroid SM_c and Dynamic MaskCentroid DM_c in human instance segmentation (§5.2).

Instance-wise Gaussian Optimization. To precisely align the predicted semantic maps, SegNet performs Gaussian smoothing [Chung, 2020] at the instance level, i.e., *instance-wise Gaussian optimization*. We apply instance-wise smoothing to reduce noise while retaining useful information, producing distinct semantic maps. The Gaussian kernel used for smoothing is defined as:

$$G(x, y) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{m_0^2 + m_1^2}{2\sigma^2}\right), \quad (8)$$

where $G(x, y)$ is the Gaussian kernel, σ is the standard deviation of the distribution, and (m_0, m_1) represents the pixel's coordinates within the kernel. We maintain σ within the range of 0.1 to 1.

We find that a σ value close to 0.1 yields finer segmentation masks, particularly in scenarios where individuals are overlapped and entangled. Our ablation experiments support this observation and demonstrate the effectiveness of instance-wise smoothing (§5.3).

3.4 PoseSeg Module

We introduce a new algorithm called PoseSeg, which simultaneously presents human pose estimation and instance segmentation, as illustrated in Fig. 4d. The PoseSeg module leverages high-level features generated by PoseNet and SegNet. Initially, keypoints and their coordinates are stored in a priority queue, facilitating the detection of body instances and the connection of adjacent keypoints. The pose kinematic graph is then followed to accurately estimate the human pose. Additionally, KDC performs instance-level segmentation by clustering pixels around centroids defined for each human instance. Specifically, pixels with a probability exceeding 0.5 are assigned to the corresponding human instances.

4 Evaluation

We evaluate KDC on COCO [Lin *et al.*, 2014], CrowdPose [Li *et al.*, 2019], and OCHuman [Zhang *et al.*, 2019] benchmarks. The model is trained end-to-end using the COCO keypoint and segmentation training set, and ablations are conducted on the COCO *val* set. ResNet-101 (RN-101) and ResNet-152 (RN-152) [He *et al.*, 2016] are used for training and testing. Hyperparameters for training are: learning rate = $0.1 \times e^{-4}$, image size = 401×401, batch size = 4, training epochs = 400, and Adam optimizer is employed. Various transformations are applied during model training, such as scale, flip, and rotate operations. Unless otherwise specified, a disk D_R's radius is set to be $R = 32$.

Keypoint Results. Table 1 presents the performance of KDC using the COCO keypoint *test* set, outperforming the recent single-stage and top-down methods. We also compare our method with bottom-up competitors. KDC with ResNet-152 yields an mAP of 76.1, outperforming existing approaches by a large margin. Specifically, 5% over Qu *et al* [Qu *et al.*, 2023], 4.9% over DecentNet [Wang *et al.*, 2023], 4.9% over QueryPose [Xiao *et al.*, 2022a], 3.3% over Pose+Seg [Ahmad *et al.*, 2022], and 3.3% over GroupPose [Liu *et al.*, 2023]. Table 2 shows the results on the CrowdPose *test* set compared to recent single-stage methods, top-down, and bottom-up models. KDC (mAP 74.5) outperforms bottom-up OpenPose [Cao *et al.*, 2017], HrHRNet [Cheng *et al.*, 2020], C.Atten. [Brasó *et al.*, 2021], and BUCTD [Zhou *et al.*, 2023]. Table 3 shows the results of KDC compared with state-of-the-art models on OCHuman challenging dataset. We assess keypoint accuracy with top competitors LOGO-CAP [Khirodkar *et al.*, 2021], MIPNet [Khirodkar *et al.*, 2021], BUCTD [Zhou *et al.*, 2023], and CID [Khirodkar *et al.*, 2021] both on *val* and *test* sets.

Segmentation Results. Table 4 presents instance-level segmentation results using COCO segmentation *test* sets. KDC delivered a top accuracy of 47.6 mAP and improved the AP by 10.5% over Mask-RCNN [He *et al.*, 2017], 5.9% over PersonLab [George *et al.*, 2018] (multi-scale), and 3.1% over Pose+Seg [Ahmad *et al.*, 2022]. Table 5 presents the segmentation performance on the OCHuman *val* and *test* sets. KDC (mAP 58.3), demonstrating a significant improvement of 3.9% and 4.4% over Pose2Seg [Zhang *et al.*, 2019] on the *val* and *test* sets, respectively.

Comparing 2D vs. 3D Pose Estimation. We also compare the pose performance with state-of-the-art 3D models CRMH [Golda and others, 2019] and ROMP [Huang *et al.*, 2017] in crowded scenes. We calculate the average precision ($AP^{0.5}$) between the 2D projection of the 3D pose on the Crowdpose *val* and *test* sets shown in Table 6.

Computational Cost. We calculate the computational cost and FPS using an image resolution of 401×401. Fig. 6 shows that KDC has fewer parameters, high FPS, and lower computational complexity compared to the representative models Mask R-CNN [He *et al.*, 2017], PersonLab [George *et al.*, 2018], and Pose+Seg [Ahmad *et al.*, 2022].

5 Ablation Experiments

5.1 KHDR and KeyCentroid

Initially, we evaluate the performance of the proposed KHDR and examine its effectiveness with and without the integration of K_c, as presented in Table 7. Through our ablation study, we observe that the combination of KHDR and K_c is a highly effective approach for human pose estimation, particularly in challenging scenarios and dynamic movement of

Models	F.Work	AP	AP^{50}	AP ^{75}	AP^{M}	AP^{L}
Single-stage:						
FCPose [Mao *et al.*, 2021]	RN101	65.6	87.9	72.6	62.1	72.3
DEKR [Geng *et al.*, 2021]	HR32	67.3	87.9	74.1	61.5	76.1
PETR[Shi *et al.*, 2022]	Swin-L	70.5	91.5	78.7	65.2	78.0
CID [Wang *et al.*, 2022]	HR48	70.7	90.3	77.9	66.3	77.8
ED-Pose[Yang *et al.*, 2023]	Swin-L	72.2	92.3	80.9	67.6	80.0
RTMO[Lu *et al.*, 2024]	Darknet	71.6	91.1	79.0	66.8	79.1
Top-down:						
Mask-RCNN [He *et al.*, 2017]	RN50	63.1	87.3	68.7	57.8	71.4
Grmi[Papandreou *et al.*, 2017]	RN-101	64.9	85.5	71.3	62.3	70.0
IntegralPose [Sun *et al.*, 2018]	RN-101	67.8	88.2	74.8	63.9	74.0
CPN [Chen *et al.*, 2018]	RN-50	72.1	91.4	80.0	68.7	77.2
RMPE [Fang *et al.*, 2017]	PyraNet	72.3	89.2	79.1	68.0	78.6
HRNet [Wang *et al.*, 2020b]	HR48	75.5	92.5	83.3	71.9	81.5
Bottom-up:						
OpenPose∗[Cao *et al.*, 2017]	-	61.8	84.9	67.5	57.1	68.2
Directpose‡[Tian *et al.*, 2019]	RN-101	64.8	87.8	71.1	60.4	71.5
PifPaf [Kreiss *et al.*, 2019]	RN-152	66.7	-	-	62.4	72.9
SPM [Nie *et al.*, 2019]	HG	66.9	88.5	72.9	62.6	73.1
PoseTrans[Jiang *et al.*, 2022]	HrHR48	67.4	88.3	73.9	62.1	75.1
SWAHR [Luo *et al.*, 2021]	HR32	67.9	88.9	74.5	62.4	75.5
Per.Lab‡[George *et al.*, 2018]	RN-152	68.7	89.0	75.4	64.1	75.5
MPose[Kocabas *et al.*, 2018]	RN-101	69.6	86.3	76.6	65.0	76.3
HrHR‡[Cheng *et al.*, 2020]	HRNet	70.5	89.3	77.2	66.6	75.8
LOGP-CAP[Xue *et al.*, 2022]	HR48	70.8	89.7	77.8	66.7	77.0
CIR&QEM[Xiao *et al.*, 2022b]	HR48	71.0	90.2	78.2	66.2	77.8
SIMPLE‡[Zhang *et al.*, 2021]	HR32	71.1	90.2	79.4	69.1	79.1
Qu *et al* [Qu *et al.*, 2023]	HrHR48	71.1	90.4	78.2	66.9	77.2
DecentNet[Wang *et al.*, 2023]	HR48	71.2	89.0	78.1	66.7	77.8
QueryPose[Xiao *et al.*, 2022a]	Swin-L	72.2	92.0	78.8	67.3	79.4
Pose+Seg[Ahmad *et al.*, 2022]	RN-152	72.8	88.4	78.7	67.8	79.4
GroupPose[Liu *et al.*, 2023]	Swin-L	72.8	92.5	81.0	67.7	80.3
KDC	**RN-101**	74.2	89.0	80.2	69.3	81.1
KDC	**RN-152**	76.1	92.9	83.9	71.1	83.5

Table 1: Performance comparison with recent works using **COCO** keypoint *test* set. F.work indicates Framework, + is trained on extra data, ∗ means refinement, ‡ is multi-scale results, HG indicates Hourglass network, and HR indicates High-Resolution Net.

Models	F.Work	AP	AP^{50}	AP^{75}	AP$_{E}$	AP$_{M}$	AP$_{H}$
Single-stage:							
DEKR[Geng *et al.*, 2021]	HRNet	65.7	85.7	70.4	73.0	66.4	57.5
PINet [Guo *et al.*, 2021]	HRNet	68.9	88.7	74.7	75.4	69.6	61.5
CID[Wang *et al.*, 2022]	HRNet	72.3	90.8	77.9	78.7	73.0	64.8
Top-down:							
MaskR[He *et al.*, 2017]	-	57.2	83.5	60.3	69.4	57.9	45.8
Al.Pose [Fang *et al.*,]	-	61.0	81.3	66.0	71.2	61.4	51.1
J-SPPE [Li *et al.*, 2019]	-	66.0	84.2	71.5	75.5	66.3	57.4
Bottom-up:							
OpenPose[Cao *et al.*, 2017]	-	-	-	-	62.7	48.7	32.3
HrHR‡[Cheng *et al.*, 2020]	HRNet	65.9	86.4	70.6	73.3	66.5	57.9
C.Atten.[Brasó *et al.*, 2021]	HRNet	67.6	87.7	72.7	75.8	68.1	58.9
BUCTD [Zhou *et al.*, 2023]	HR48	72.9	-	-	79.2	73.4	66.1
KDC	**RN-101**	71.6	87.1	75.2	78.4	71.9	59.7
KDC	**RN-152**	74.5	89.7	76.8	80.1	74.8	62.6

Table 2: Performance comparison on **CrowdPose** keypoint *test* set. ‡ is multi-scale testing.

the human body. Fig. 5 shows the visual performance of keypoint heatmap improved by KeyCenroid. Fig. 7 shows the predicted confidence score of 17 keypoints using the keypoint disk at radius $R = 8$, 16, and 32.

5.2 Static vs. Dynamic MaskCentroids

Models	F.Work	Val mAP	Test mAP
HGG [Jin *et al.*, 2020]	HG	35.6	34.8
DEKR [Geng *et al.*, 2021]	HRNet	37.9	36.5
HrHR [Cheng *et al.*, 2020]	HrHR32	40.0	39.4
LOGO-CAP [Xue *et al.*, 2022]	HR48	41.2	40.4
MIPNet [Khirodkar *et al.*, 2021]	RN-101	42.0	42.5
BUCTD [Zhou *et al.*, 2023]	HrHR32	44.1	43.5
CID [Wang *et al.*, 2022]	HR32	45.7	44.6
KDC	**RN-101**	44.1	44.6
KDC	**RN-152**	46.3	46.0

Table 3: Performance using **OCHuman** keypoint *val* and *test* datasets.

Models	F.Work	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}
MaskRCNN[He *et al.*, 2017]	RN-101	37.1	60.0	39.4	39.9	53.5
Per.Lab†[George *et al.*, 2018]	RN-101	37.7	65.9	39.4	48.0	59.5
Per.Lab†[George *et al.*, 2018]	RN-152	38.5	66.8	40.4	48.8	60.2
Per.Lab‡[George *et al.*, 2018]	RN-101	41.1	68.6	44.5	49.6	62.6
Per.Lab‡[George *et al.*, 2018]	RN-152	41.7	69.1	45.3	50.2	63.0
Pose+Seg[Ahmad *et al.*, 2022]	RN-152	44.5	79.4	47.1	52.4	65.1
KDC	**RN-101**	45.7	80.4	47.8	53.5	67.4
KDC	**RN-152**	47.6	81.8	48.7	54.6	67.8

Table 4: Performance comparison on **COCO** Segmentation *test* set. † is single-scale testing. ‡ is multi-scale testing.

Models	F.Work	Val mAP	Test mAP
Pose2Seg [Zhang *et al.*, 2019]	RN-50-fpn	54.4	55.2
KDC	**RN-101**	56.7	57.0
KDC	**RN-152**	58.3	59.6

Table 5: Performance comparison using **OCHuman** segmentation *val* and *test* datasets.

Models	F.Work	Val mAP	Test mAP
CRMH [Golda and others, 2019]	-	32.9	33.9
ROMP [Huang *et al.*, 2017]	RN-50	55.6	54.1
ROMP+CAR [Huang *et al.*, 2017]	RN-50	58.6	59.7
KDC	**RN-101**	86.3	87.1
KDC	**RN-152**	88.1	89.7

Table 6: Comparisons with 3D methods on the **CrowdPose** benchmark using $AP^{.50}$ evaluation metric.

KDC w and w/o						
KHDR	k_c	**AP**	**AP**$^{.50}$	**AP** $^{.75}$	**AP**^{M}	**AP**^{L}
✓		74.8	89.7	75.6	70.3	79.1
	✓	76.2	91.8	78.9	72.5	82.7
✓	✓	77.5	94.9	86.4	73.8	84.6

Table 7: Performance of KHDR with and without KeyCentroid k_c mechanism.

We analyze the Static MaskCentroid (SM_c) and the Dynamic MaskCentroid (DM_c), with the results presented in Fig. 8. The exceptional performance of the proposed DM_c approach demonstrates its effectiveness in human body segmentation, particularly in scenarios involving dynamic human body movement. Fig. 5 shows the visual performance of SM_c improved by DM_c. This capability significantly contributes to advancements in instance-level segmentation.

5.3 Point & Instance-wise Gaussian Optimization

We generated keypoint heatmap utilizing point-wise Gaussian optimization using $0.1 \leq \sigma \leq 1$. Fig. 9 summarizes the mAP for different σ with high variation of keypoints (e.g., wrist, ankle, elbow, and knee) and low variation of keypoints (e.g., nose, shoulder, hip).

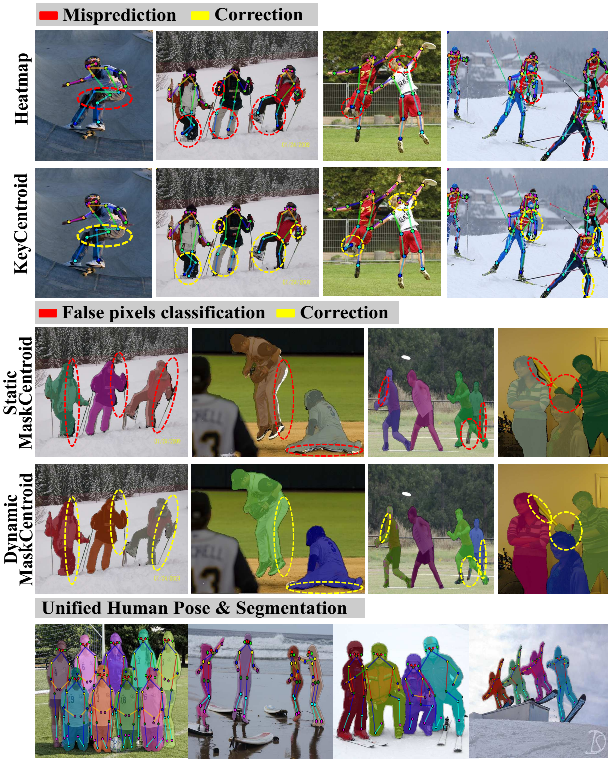

🟥 Misprediction 🟨 Correction
Heatmap
KeyCentroid
🟥 False pixels classification 🟨 Correction
Static MaskCentroid
Dynamic MaskCentroid
Unified Human Pose & Segmentation

Figure 5: Visual results from various components of the system reveal initial mispredictions and inaccuracies in the keypoint heatmap (first row), corrected by KeyCentroid (second row). False pixel classification in segmentation with Static MaskCentroid (third row) was resolved using Dynamic MaskCentroid (fourth row). Unified human pose and segmentation are shown in the fifth row.

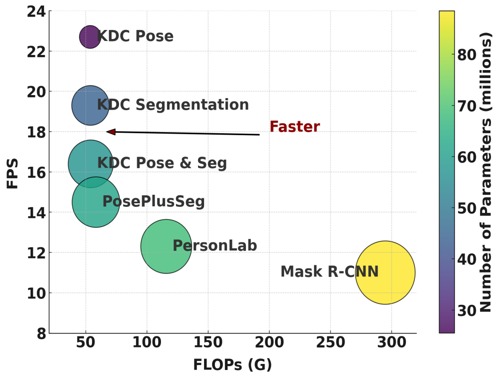


Figure 6: Computational cost with the representative sister models. Models are tested on a single Titan RTX.

Finally, we examine the impact of instance-wise Gaussian optimization on the instance segmentation task. We tested the sensitivity of σ ranging from 0.1 to 0.5 on human instance segmentation. Fig. 10 shows the results with different σ values, where low σ provides precise segmentation mask and performs better in crowded cases.

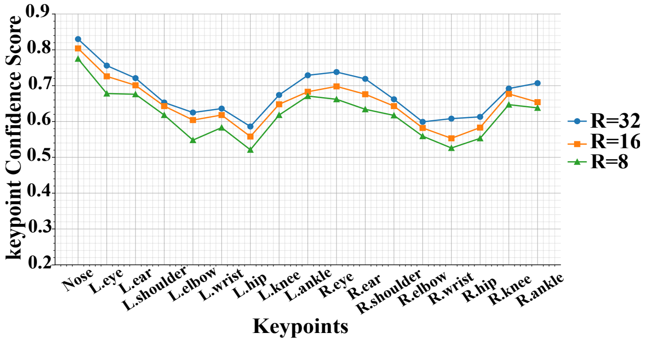


Figure 7: Left(L) and right(R) keypoint confidence score with varied disk radius $R = \{32, 16, 8\}$.

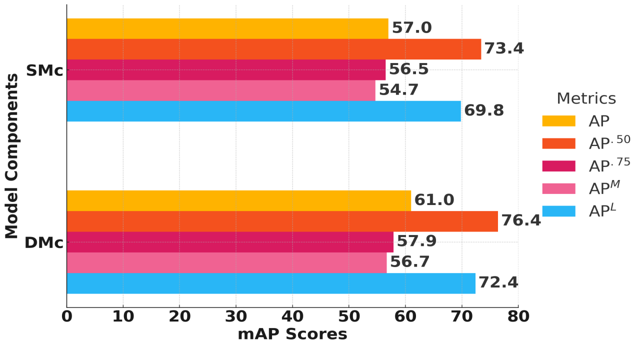


Figure 8: Performance of SM_c and DM_c on human instance-level segmentation.

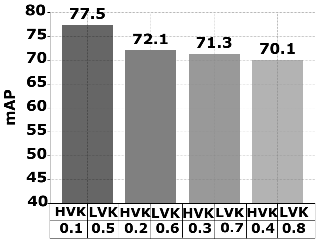


Figure 9: Point-wise Gaussian optimization with different σ values for HVK and LVK.

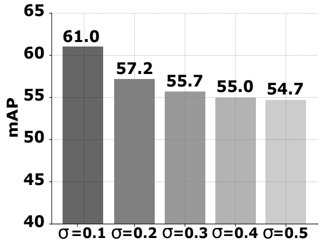


Figure 10: Instance-wise Gaussian optimization with different σ. Small σ provides precise instance mask.

6 Conclusion

This paper considers the challenge of unified human pose estimation and instance-level segmentation, particularly in complex multi-person dynamic movement scenarios. KDC generate keypoint heatmaps by defining keypoint disks and KeyCentroid to determine the optimal 2D keypoint coordinates within the specified keypoint disk. Additionally, MaskCentroid is introduced, representing highly confident keypoint as dynamic centroids to cluster the mask pixels with the correct instance in the embedding space, even under significant occlusion or body movement. The effectiveness of KDC is evaluated on COCO, CrowdPose, and OCHuman benchmarks and proves to be a highly effective approach for unified human pose estimation and instance-level segmentation.

Acknowledgments

This work was partly funded by the Natural Sciences and Engineering Research Council of Canada (NSERC) grant RGPIN-2021-04244, the Institute of Information and Communications Technology Planning and Evaluation (IITP) grant IITP-2025-RS-2020-II201741, RS-2022-00155885, RS-2024-00423071 funded by the Korea government (MSIT).

References

[Ahmad *et al.*, 2022] Niaz Ahmad, Jawad Khan, Jeremy Yuhyun Kim, and Youngmoon Lee. Joint Human Pose Estimation and Instance Segmentation with PosePlusSeg. In *AAAI*, 2022.

[Ahmad *et al.*, 2025] Niaz Ahmad, Youngmoon Lee, and Guanghui Wang. Visualcent: Visual human analysis using dynamic centroid representation. In *FG*. IEEE, 2025.

[Bolya *et al.*, 2019] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: Real-time instance segmentation. In *ICCV*, 2019.

[Brasó *et al.*, 2021] Guillem Brasó, Nikita Kister, and Laura Leal-Taixé. The center of attention: Center-keypoint grouping via attention for multi-person pose estimation. In *ICCV*, 2021.

[Cai *et al.*, 2020] Yuanhao Cai, Zhicheng Wang, Zhengxiong Luo, Binyi Yin, Angang Du, Haoqian Wang, Xiangyu Zhang, Xinyu Zhou, Erjin Zhou, and Jian Sun. Learning delicate local representations for multi-person pose estimation. In *ECCV*, 2020.

[Cao *et al.*, 2017] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017.

[Chen *et al.*, 2018] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *CVPR*, 2018.

[Cheng *et al.*, 2020] Bowen Cheng, Bin Xiao, Jingdong Wang, Honghui Shi, Thomas S Huang, and Lei Zhang. Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation. In *CVPR*, 2020.

[Chung, 2020] Moo K Chung. Gaussian kernel smoothing. *arXiv preprint arXiv:2007.09539*, 2020.

[Dai *et al.*, 2016] Jifeng Dai, Kaiming He, Yi Li, Shaoqing Ren, and Jian Sun. Instance-sensitive fully convolutional networks. In *ECCV*, 2016.

[Dantone *et al.*, 2013] Matthias Dantone, Juergen Gall, Christian Leistner, and Luc Van Gool. Human pose estimation using body parts dependent joint regressors. In *CVPR*, 2013.

[Fang *et al.*,] Hao-Shu Fang, Jiefeng Li, Hongyang Tang, Chao Xu, Haoyi Zhu, Yuliang Xiu, Yong-Lu Li, and Cewu Lu. Alphapose: Whole-body regional multi-person pose estimation and tracking in real-time. *TPAMI*.

[Fang *et al.*, 2017] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. Rmpe: Regional multi-person pose estimation. In *ICCV*, 2017.

[Geng *et al.*, 2021] Zigang Geng, Ke Sun, Bin Xiao, Zhaoxiang Zhang, and Jingdong Wang. Bottom-up human pose estimation via disentangled keypoint reg. In *CVPR*, 2021.

[George *et al.*, 2018] George, Zhu, Chen, Jonathan Gidaris, Tompson, and Kevin Murphy. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In *ECCV*, 2018.

[Golda and others, 2019] Golda et al. Human pose estimation for real-world crowded scenarios. In *AVSS*, 2019.

[Guo *et al.*, 2021] Wen Guo, Enric Corona, Francesc Moreno-Noguer, and Xavier Alameda-Pineda. Pi-net: Pose interacting network for multi-person monocular 3d pose estimation. In *WACV*, 2021.

[Han *et al.*, 2025] Gangtao Han, Chunxiao Song, Song Wang, Hao Wang, Enqing Chen, and Guanghui Wang. Occluded human pose estimation based on limb joint augmentation. *Neural Computing and Applications*, 37(3):1241–1253, 2025.

[He *et al.*, 2016] He, Xiangyu, Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[He *et al.*, 2017] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017.

[Huang *et al.*, 2017] Shaoli Huang, Mingming Gong, and Dacheng Tao. A coarse-fine network for keypoint localization. In *ICCV*, 2017.

[Insafutdinov *et al.*, 2016] Eldar Insafutdinov, Leonid Pishchulin, Bjoern Andres, Mykhaylo Andriluka, and Bernt Schiele. Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In *ECCV*, 2016.

[Jiang *et al.*, 2022] Wentao Jiang, Sheng Jin, Wentao Liu, Chen Qian, Ping Luo, and Si Liu. Posetrans: A simple yet effective pose transformation augmentation for human pose estimation. In *ECCV*, 2022.

[Jin *et al.*, 2020] Sheng Jin, Wentao Liu, Enze Xie, Wenhai Wang, Chen Qian, Wanli Ouyang, and Ping Luo. Differentiable hierarchical graph grouping for multi-person pose estimation. In *ECCV*, 2020.

[Khirodkar *et al.*, 2021] Rawal Khirodkar, Visesh Chari, Amit Agrawal, and Ambrish Tyagi. Multi instance pose nets: Rethinking topdown pose estimation. In *ICCV*, 2021.

[Kocabas *et al.*, 2018] Muhammed Kocabas, Salih Karagoz, and Emre Akbas. Multiposenet: Fast multi-person pose estimation using pose residual network. In *ECCV*, 2018.

[Kreiss *et al.*, 2019] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. Pifpaf: Composite fields for human pose estimation. In *CVPR*, 2019.

[Li *et al.*, 2019] Jiefeng Li, Can Wang, Hao Zhu, Yihuan Mao, Hao-Shu Fang, and Cewu Lu. Crowdpose: Efficient crowded scenes pose estimation and a new benchmark. In *CVPR*, 2019.

[Lin *et al.*, 2014] Tsung Lin, Michael Maire, Serge Belongie, James, Perona, Deva, Piotr, and Lawrence. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[Lin *et al.*, 2017] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid nets for object detection. In *CVPR*, 2017.

[Liu *et al.*, 2018] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *CVPR*, 2018.

[Liu *et al.*, 2023] Huan Liu, Qiang Chen, Zichang Tan, Jiang-Jiang Liu, Jian Wang, Xiangbo Su, Xiaolong Li, Kun Yao, Junyu Han, Errui Ding, et al. Group pose: A simple baseline for end-to-end multi-person pose estimation. In *ICCV*, 2023.

[Long *et al.*, 2015] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[Lu *et al.*, 2024] 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 *CVPR*, 2024.

[Luo *et al.*, 2021] Zhengxiong Luo, Zhicheng Wang, Yan Huang, Liang Wang, Tieniu Tan, and Erjin Zhou. Rethinking the heatmap regression for bottom-up human pose estimation. In *CVPR*, 2021.

[Mao *et al.*, 2021] Weian Mao, Zhi Tian, Xinlong Wang, and Chunhua Shen. Fcpose: Fully convolutional multi-person pose estimation with dynamic instance-aware convolutions. In *CVPR*, 2021.

[Nie *et al.*, 2019] Xuecheng Nie, Jiashi Feng, Jianfeng Zhang, and Shuicheng Yan. Single-stage multi-person pose machines. In *ICCV*, 2019.

[Papandreou *et al.*, 2017] George Papandreou, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, and Kevin Murphy. Towards accurate multi-person pose estimation in the wild. In *CVPR*, 2017.

[Pishchulin *et al.*, 2016] Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter V Gehler, and Bernt Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In *CVPR*, 2016.

[Qu *et al.*, 2023] Haoxuan Qu, Yujun Cai, Lin Geng Foo, Ajay Kumar, and Jun Liu. A char. function based method for bottom-up human pose estimation. In *CVPR*, 2023.

[Ren *et al.*, 2015] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal nets. In *NeurIPS*, 2015.

[Shi *et al.*, 2022] Dahu Shi, Xing Wei, Liangqi Li, Ye Ren, and Wenming Tan. End-to-end multi-person pose estimation with transformers. In *CVPR*, 2022.

[Su *et al.*, 2019] Kai Su, Dongdong Yu, Zhenqi Xu, Xin Geng, and Changhu Wang. Multi-person pose estimation with enhanced channel-wise and spatial information. In *CVPR*, 2019.

[Sun *et al.*, 2018] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *ECCV*, 2018.

[Tian *et al.*, 2019] Zhi Tian, Hao Chen, and Chunhua Shen. Directpose: Direct end-to-end multi-person pose estimation. *arXiv preprint arXiv:1911.07451*, 2019.

[Wang *et al.*, 2020a] Jian Wang, Xiang Long, Yuan Gao, Errui Ding, and Shilei Wen. Graph-pcnn: Two stage human pose estimation with graph pose refine. In *ECCV*, 2020.

[Wang *et al.*, 2020b] 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. *TPAMI*, 43(10):3349–3364, 2020.

[Wang *et al.*, 2022] Dongkai Wang, Shiliang Zhang, and Shiliang Zhang. Contextual instance decoupling for robust multi-person pose estimation. In *CVPR*, 2022.

[Wang *et al.*, 2023] Tao Wang, Lei Jin, Zhang Wang, Xiaojin Fan, Yu Cheng, Yinglei Teng, Junliang Xing, and Jian Zhao. Decenternet: Bottom-up human pose estimation via decentralized pose representation. In *MM*, 2023.

[Wei *et al.*, 2016] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *CVPR*, 2016.

[Xiao *et al.*, 2018] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *ECCV*, 2018.

[Xiao *et al.*, 2022a] Yabo Xiao, Kai Su, Xiaojuan Wang, Dongdong Yu, Lei Jin, Mingshu He, and Zehuan Yuan. Querypose: sparse multi-person pose regression via spatial-aware part-level query. *NeurIPS*, 2022.

[Xiao *et al.*, 2022b] Yabo Xiao, Dongdong Yu, Xiao Juan Wang, Lei Jin, Guoli Wang, and Qian Zhang. Learning quality-aware representation for multi-person pose regression. In *AAAI*, 2022.

[Xue *et al.*, 2022] Nan Xue, Tianfu Wu, Gui-Song Xia, and Liangpei Zhang. Learning local-global contextual adaptation for multi-person pose estimation. In *CVPR*, 2022.

[Yang *et al.*, 2023] Jie Yang, Ailing Zeng, Shilong Liu, Feng Li, Ruimao Zhang, and Lei Zhang. Explicit box detection unifies end-to-end multi-person pose estimation. *arXiv preprint arXiv:2302.01593*, 2023.

[Zhang *et al.*, 2019] Song-Hai Zhang, Ruilong Li, Xin Dong, Paul Rosin, Zixi Cai, Xi Han, Dingcheng Yang, Haozhi Huang, and Shi-Min Hu. Pose2seg: Detection free human instance segmentation. In *CVPR*, 2019.

[Zhang *et al.*, 2021] Jiabin Zhang, Zheng Zhu, Jiwen Lu, Junjie Huang, Guan Huang, and Jie Zhou. Simple: Single-network with mimicking and point learning for bottom-up human pose estimation. In *AAAI*, 2021.

[Zhou *et al.*, 2023] Mu Zhou, Lucas Stoffl, Mackenzie Weygandt Mathis, and Alexander Mathis. Rethinking pose estimation in crowds: overcoming the detection information bottleneck and ambiguity. In *ICCV*, 2023.