Heatmap Distribution Matching for Human Pose Estimation

Anonymous Author(s)
Affiliation
Address
email

Abstract

For tackling the task of 2D human pose estimation, the great majority of the recent methods regard this task as a heatmap estimation problem, and optimize the heatmap prediction using the Gaussian-smoothed heatmap as the optimization objective. In this paper, we show that using the Gaussian-smoothed heatmap as the optimization objective, the model performance of body joint localization, which is the intrinsic objective of this task, may not be consistently improved during the optimization process of the heatmap prediction. To address this problem, from a novel perspective, we propose to formulate the optimization of the heatmap prediction as a distribution matching problem between the predicted heatmap and the dot annotation of the body joint directly. By doing so, our proposed method bypasses the step of constructing the Gaussian-smoothed heatmap and can achieve a more consistent model performance improvement during the optimization of the heatmap prediction. We show the effectiveness of our proposed method through extensive experiments on the COCO dataset and the MPII dataset.

1 Introduction

2D human pose estimation aims to locate body joints of a person in a given RGB image. It is relevant to a variety of applications, such as action recognition [34], human-machine interaction [40], and sign language understanding [19]. For tackling the task of 2D human pose estimation, most of the recent methods [29, 20, 33, 26, 14, 4, 39, 35, 15, 17, 37] are *heatmap-based*, i.e., they regard 2D human pose estimation as a heatmap estimation problem. Specifically, for each body joint, these methods generally estimate a grid-like heatmap, on which each pixel value represents the probability that this pixel contains the body joint. Compared to the methods [30, 2, 32, 13] that directly regress the coordinates of body joints (i.e. *coordinate regression-based methods*), the *heatmap-based methods* demonstrate a more robust performance since they maintain the spatial structure of the input image throughout the encoding and decoding process [7].

During the training process of the *heatmap-based methods*, an important step is the optimization of the heatmap prediction. This optimization can be done naively via constructing a dot-annotated heatmap for each body joint as shown in Fig. 1(a), and then measuring the difference (i.e., conducting pixel-wise comparison) between the predicted heatmap and the constructed ground-truth (GT) dot-annotated heatmap. However, such a dot-annotated heatmap is sparse, as it has the same zero value for all pixels except the pixel representing the dot annotation of the body joint. Because of this, optimizing the heatmap prediction in such a naive way can lead to a hard training process and a suboptimal model performance [28]. To address this problem, most *heatmap-based methods* [29, 20, 33, 26, 14, 4, 39, 35, 15, 17, 37] adopt a strategy to construct Gaussian-smoothed heatmaps,

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

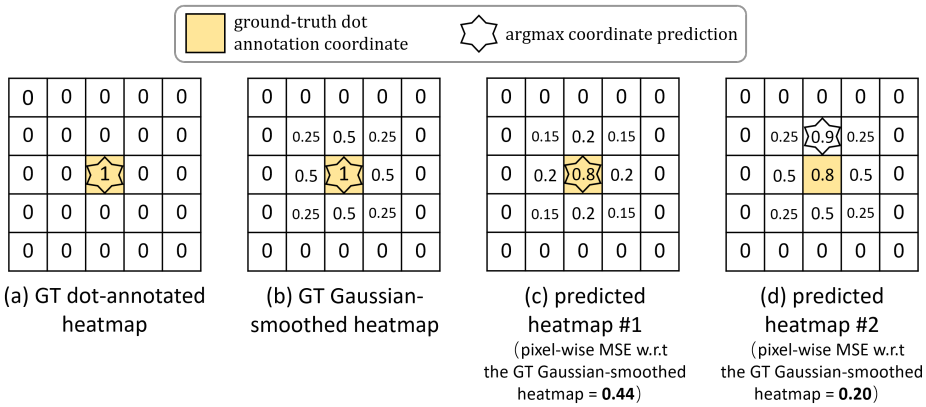

Figure 1: Illustration of heatmaps. Although the pixel-wise MSE loss calculated between the predicted heatmap #2 and the Gaussian-smoothed heatmap is smaller than the loss calculated between the predicted heatmap #1 and the Gaussian-smoothed heatmap, the predicted heatmap #2 localizes the body joint wrongly, whereas the predicted heatmap #1 localizes the body joint correctly.

where pixels near the dot annotation have larger pixel values than pixels far from the dot annotation. Specifically, they construct the Gaussian-smoothed heatmap via smoothing the dot annotation of the body joint through a Gaussian distribution as shown in Fig. 1(b), instead of only setting the pixel representing the dot annotation to be one.

While easing the training process, constructing the GT Gaussian-smoothed heatmap still brings problems into the model training process. Firstly, for constructing the Gaussian-smoothed heatmap, we need to choose a proper standard deviation of the Gaussian distributions. However, the proper standard deviations of the Gaussian distributions (i.e., the standard deviations that can lead to an optimal performance) often vary across different types of body joints, different body postures, and different body sizes [17]. Hence, the standard deviations of the Gaussian distributions often need to be carefully chosen, which is non-trivial. Secondly, during the process of optimizing the heatmap prediction by minimizing the pixel-wise loss (e.g. MSE) between the predicted heatmap and the Gaussian-smoothed heatmap, the model performance of body joint localization may not be consistently improved. As shown in Fig. 1, although compared to the loss calculated between the predicted heatmap #1 and the Gaussian-smoothed heatmap, the pixel-wise MSE loss calculated between the predicted heatmap #2 and the Gaussian-smoothed heatmap is smaller, the predicted heatmap #2 localizes the body joint wrongly, whereas the predicted heatmap #1 localizes the body joint correctly.

As a result, optimizing the heatmap prediction using the dot-annotated heatmap and the Gaussian-smoothed heatmap as the ground-truth both have their respective problems. Hence, in this work, we aim to bypass the step of constructing the GT heatmap, and propose to optimize the heatmap prediction directly via minimizing the difference between the predicted heatmap and the dot annotation. By doing so, we can optimize the model directly towards accurately localizing the dot annotation of the body joint, which is the intrinsic objective of 2D human pose estimation, instead of optimizing the model indirectly towards either the dot-annotated heatmap or the Gaussian-smoothed heatmap. However, as the number of pixels in the predicted heatmap and the number of entries representing the dot annotation are different, we cannot measure the difference between the predicted heatmap and the dot annotation trivially by measuring their entry-wise difference. To handle this problem, inspired by the fact that we can measure the difference between two distributions via measuring their Earth Mover's Distance even if they have different numbers of entries, in this paper, we propose to first formulate the optimization of the heatmap prediction as a distribution matching problem. Specifically, we construct two distributions respectively from the predicted heatmap and the dot annotation. After that, we optimize the heatmap prediction via minimizing the distribution difference based on the Earth Mover's Distance. Using such a novel method to optimize the heatmap prediction directly from the dot annotation, we bypass the step of constructing the Gaussian-smoothed heatmap, as well as avoiding the issues of the binary dot-annotated heatmap. Thus, our method achieves superior performance.

Our proposed method is simple yet effective, which can be easily applied to various off-the-shelf 2D human pose estimation models by replacing their original loss function with our proposed loss function measuring the distribution difference between the predicted heatmap and the dot annotation. We experiment our proposed method on multiple models and our method achieves a consistent model performance improvement.

The contributions of our work are summarized as follows. 1) We analyze (in Sec. 4) that the performance of the human pose estimation model may not be consistently improved during the process of minimizing the pixel-wise loss between the predicted heatmap and the GT Gaussian-smoothed heatmap. 2) From a novel perspective, we formulate the optimization of the heatmap prediction as a distribution matching problem between the predicted heatmap and the GT dot annotation directly, which bypasses the step of constructing the Gaussian-smoothed heatmap and achieves consistent model performance improvement. 3) Our proposed method achieves state-of-the-art performance on the evaluated benchmarks.

2 Related Work

2D Human Pose Estimation. Due to the wide range of applications, the task of 2D human pose estimation has received lots of attention [30, 2, 32, 13, 21, 29, 20, 33, 26, 14, 4, 39, 35, 15, 17, 37, 27, 7, 8]. DeepPose [30] made the first attempt of applying deep neural networks into the task of 2D human pose estimation via directly regressing the coordinates of body joints. This type of *coordinate regression-based methods* [30, 2, 32, 13] often show inferior performances compared to the *heatmap-based methods* [29, 20, 33, 26, 14, 4, 39, 35, 15, 17, 37], as the *heatmap-based methods* can preserve the spatial structure of the input image throughout the encoding and decoding process [7]. Hence, recently, the great majority of the state-of-the-art methods [29, 20, 33, 26, 14, 4, 39, 35, 15, 17, 37] regard 2D human pose estimation as a heatmap estimation problem instead of the coordinate regression problem. Among the *heatmap-based methods*, Tompson et al. [29] proposed to apply Markov Random Field (MRF) into the task of 2D human pose estimation. After that, an "hourglass" network, with a conv-deconv architecture, was proposed by Newell et al. [20]. Xiao et al. [33] proposed a baseline method to predict the heatmap via adding several deconvolutional layers to a backbone network. Later on, to maintain high-resolution representations throughout the heatmap estimation process, HRNet was proposed by Sun et al. [26]. Yuan et al. [37] further proposed HRFormer to learn the high-resolution representations utilizing a transformer-based architecture. Besides the above *heatmap-based methods* that use the Gaussian-smoothed heatmap as the optimization objective, there are also some methods [27, 7, 8] that combine the idea of *heatmap* and *coordinate regression* by taking the expectation of the predicted heatmap as the predicted coordinates.

Here in this work, different from previous works, our method bypasses both the step of regression and the step of constructing the Gaussian-smoothed heatmap as the optimization objective. Instead, from a novel perspective, we propose to formulate the optimization of the heatmap prediction as a distribution matching problem by minimizing the distribution difference between the predicted heatmap and the dot annotation.

Distribution Matching. The idea of distribution matching has been studied in various tasks [24, 25, 31, 38, 41, 23], such as image retrieval [24], tracking [25], few-shot learning [38], and long-tail recognition [23]. In this work, from a novel perspective, we design a new distribution matching scheme to optimize the heatmap prediction for 2D human pose estimation.

3 Method

In 2D human pose estimation, optimizing the heatmap prediction using the dot-annotated heatmap and the Gaussian-smoothed heatmap as the ground-truth both have their respective problems. Hence, in this work, we aim to bypass the step of GT heatmap construction, and optimize the heatmap prediction directly with the dot annotation of the body joint. To achieve this goal, we propose

to formulate the optimization of the heatmap prediction as a distribution matching problem, and minimize the difference between the distribution constructed from the predicted heatmap and that constructed from the dot annotation based on the Earth Mover's Distance.

Below, we first give a brief review of the Earth Mover's Distance, and then discuss how we formulate the heatmap optimization process as a distribution matching problem. After that, we introduce how we construct the loss function measuring the distribution difference.

3.1 Revisiting Earth Mover's Distance

The Earth Mover's Distance is a a technique used for measuring the difference between two probability distributions, which can be understood as the optimal cost needed to transport the mass from one distribution to another. Specifically, for calculating the Earth Mover's Distance, we regard the source distribution as a set of (N) suppliers $S = (s_1, ..., s_N)^\top$, where s_n represents the total units of mass that the n-th supplier has, and we regard the target distribution as a set of (M) demanders $D = (d_1, ..., d_M)^\top$, where d_m represents the total units of mass that the m-th demander requires. Besides, we also denote $C \in R_{\geq 0}^{N \times M}$ as the cost function between the source and target distributions, where $C_{n,m}$ represents the cost for transporting a unit of mass from the n-th supplier to the m-th demander. Then we aim to find a least-cost transportation plan from the set of possible plans $P = \{p \in R_{\geq 0}^{N \times M} : \ p\mathbf{1} = S \ \& \ p^\top \mathbf{1} = D\}$ to transport all mass from the N suppliers to the M demanders, where **1** represents a vector of ones. The Earth Mover's Distance $E_C(S, D)$ denotes the cost of the least-cost transportation plan, which can be formulated as:

$$E_C(S, D) \ = \ \min_{p \in P} \quad \langle C, p \rangle \tag{1}$$

where $\langle \cdot, \cdot \rangle$ represents the Frobenius dot product.

However, optimizing Eq. 1 directly is computationally expensive. Hence, to reduce the computational cost especially when handling large-scale problems, Cuturi [5] proposed a regularized formulation of the Earth Mover's Distance $E_C^{reg}(S, D)$ as:

$$E_C^{reg}(S, D) \ = \ \langle C, p^{reg} \rangle \ \textbf{where} \ p^{reg} \ = \ \arg\min_{p \in P} \quad \left[\langle C, p \rangle - \frac{1}{\lambda} h(p) \right] \tag{2}$$

where $\lambda > 0$, and $h(p) = -\sum_{n=1}^{N} \sum_{m=1}^{M} C_{n,m} \log C_{n,m}$. Optimizing Eq. 2 is computationally cheaper than optimizing Eq. 1, since Eq. 2 can be optimized with matrix scaling through the Sinkhorn algorithm.

3.2 Problem Formulation

Below, we formulate the optimization of the heatmap prediction as a distribution matching problem by constructing the suppliers S, the demanders D, and the cost function C.

The suppliers S. For a body joint, let $H_{pred} \in R^{H \times W}$ denote its corresponding predicted heatmap with height H and width W. To formulate the suppliers S to represent H_{pred}, we first localize a set of $H \times W$ suppliers corresponding to the $H \times W$ pixels of H_{pred}. Then, for determining the units of mass each supplier stores, as the suppliers cannot hold negative units of mass, we construct S based on a non-nagetive formulation of H_{pred}. Besides, we also constrain the total units of mass stored by the suppliers S to be the same as the total units of mass required by the demanders D. To meet these requirements, we derive S by first passing H_{pred} through a $relu$ activation function and then normalizing it as:

$$S \ = \ \frac{relu(H_{pred})}{\|relu(H_{pred})\|_1} \tag{3}$$

The demanders D. As for the demanders, we aim to construct D w.r.t. each body joint to represent its corresponding GT dot annotation. To achieve this goal, for each body joint, a naive formulation of D is to identify the pixel containing the GT dot annotation (i.e., the upper left pixel in Fig. 2(a)) and construct a single demander (i.e., the yellow dot in Fig. 2(a)) at the center of this pixel. However, this naive formulation can result in a suboptimal model performance, as the demander formulated

in this way can be noticeably different from what the GT dot annotation might suggest. Generally, the predicted heatmaps outputted by most of the existing *heatmap-based methods* [29, 20, 33, 26, 14, 4, 39, 35, 15, 17, 37] have a lower resolution compared to the input image. For example, for the method HRNet [26], when the size of the input image is 384×288, the size of the predicted heatmap is 96×72 only. Due to such a resolution gap, as shown in Fig. 2(a), there can exist a non-negligible distance between the location of the demander and the location of the dot annotation, which can affect the performance of the pose estimation model.

To address this problem, we aim to make the formulated demanders D a more accurate representation of the dot annotation by overcoming the resolution gap. To achieve this, inspired by the fact that the more accurate location information of an object can be deduced using sub-pixel resolutions [12], we propose to formulate the demanders D with the following four steps. (1) As illustrated in the upper left of Fig. 2(b), we first split the pixel containing the dot annotation into four "sub-pixels" (four squares separated from each other by dashed lines). (2) After that, we identify the "sub-pixel" that the dot annotation lies in (i.e., the green "sub-pixel" shown in Fig. 2(b)). (3) Then, as shown by the four yellow dots in Fig. 2(b), we localize a set of four demanders at the centers of both the pixel containing the identified "sub-pixel" (i.e., the upper left pixel in Fig. 2(b)), and the three pixels adjacent to the identified "sub-pixel" (i.e., the three blue pixels in Fig. 2(b)). (4) Finally, we determine the units of mass each demander requires in the following way so that the dot annotation can be precisely represented using the set of the four demanders. Specifically, denote the 2D Euclidean distance between the centers of two adjacent pixels as g (which is also the pixel size), the coordinates of the GT dot annotation as (x_{dot}, y_{dot}), and the coordinates of the i-th demander as (x_i, y_i), where $i \in \{1, 2, 3, 4\}$. Then we construct the demanders D as:

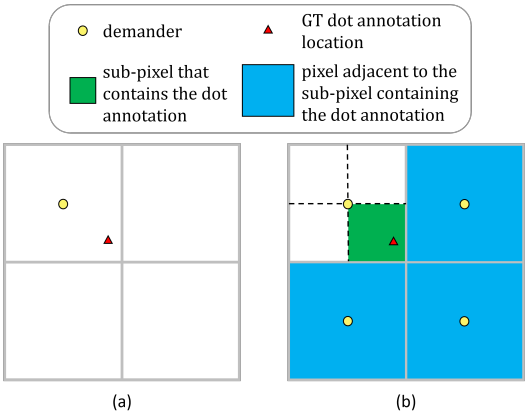

Figure 2: Illustration of (a) the naive formulation of the demanders D, and (b) formulation of the demanders D that involves the idea of sub-pixel resolution.

$$D = (d_1, d_2, d_3, d_4), \ \textbf{where } d_i = \frac{(g - |x_{dot} - x_i|) \times (g - |y_{dot} - y_i|)}{g^2} \tag{4}$$

where d_i denotes the units of mass the i-th demander requires. By designing d_i in this way, we can achieve the following two properties. (1) Among the four demanders, via assigning more mass to the demanders nearer the dot annotation and less mass to the demanders farther from the dot annotation, we can accurately derive the coordinates of the dot annotation as $(x_{dot}, y_{dot}) = (d_1 \times x_1 + d_2 \times x_2 + d_3 \times x_3 + d_4 \times x_4, d_1 \times y_1 + d_2 \times y_2 + d_3 \times y_3 + d_4 \times y_4)$. Hence, the set of demanders D constructed in this way can represent the GT dot annotation accurately, regardless of how many times the resolution of the heatmap is lower than the resolution of the input image. (2) Besides, as $d_1 + d_2 + d_3 + d_4 = 1$, the total units of mass required by the demanders D are also constrained to be the same as the total units of mass stored by the suppliers S.

The cost function C. To measure the distribution difference between the $H \times W$ suppliers (S) constructed from the predicted heatmap and the 4 demanders (D) constructed from the GT dot annotation via calculating their Earth Mover's Distance, we also need to formulate a cost function $C \in R_{\geq 0}^{HW \times 4}$. Denote the coordinates of the n-th supplier as (x_{s_n}, y_{s_n}), where $n \in \{1, ..., H \times W\}$, and the coordinates of the m-th demander as (x_{d_m}, y_{d_m}), where $m \in \{1, 2, 3, 4\}$. We here simply formulate $C_{n,m}$, the cost per unit transported from the n-th supplier to the m-th demander, as the L2 distance between the n-th supplier and the m-th demander, i.e., $C_{n,m} = \sqrt{(x_{d_m} - x_{s_n})^2 + (y_{d_m} - y_{s_n})^2}$. Using such a cost function, our method can optimize the model directly towards accurately localizing the dot annotation of the body joint via minimizing the distribution difference between the suppliers S and the demanders D.

3.3 Loss Function

Above we formulate the optimization of the heatmap prediction as a distribution matching problem for a single body joint. In this section, we introduce how we construct the loss function following such a formulation. Note that in 2D human pose estimation, we need to locate multiple body joints. Therefore, to construct the loss function, we first construct the suppliers, the demanders, and the cost function for each body joint respectively. After that, we calculate the loss value corresponding to each body joint via measuring the distribution difference (calculating the Earth Mover's Distance) between its corresponding demanders and suppliers. Specifically, denote K the total number of body joints and L_k the loss term calculated w.r.t. the k-th body joint. We then formulate the loss function as:

$$L_{Matching} = \sum_{k=1}^{K} L_k, \text{ where } L_k \ = \ E^{reg}_{C^k}(S^k, D^k) \tag{5}$$

where S^k, D^k, and C^k respectively denote the constructed suppliers, demanders, and cost function for the k-th body joint. The corresponding Earth Mover's Distance $E^{reg}_{C^k}(S^k, D^k)$ is calculated using Eq. 2.

3.4 Training and Testing

Our proposed method can be flexibly applied on various off-the-shelf 2D human pose estimation models. During training, we optimize the heatmap prediction via minimizing the loss function in Eq. 5. During testing, for each body joint, we first select a square of four adjacent pixels with the largest sum of pixel values from the predicted heatmap, and then normalize the sum of these four pixel values to 1. The normalized pixel values of these four pixels can be seen as the units of mass each of them requires. Hence, in the same way as how we get the coordinates of the GT dot annotation from the demanders in Sec. 3.2, we can get the predicted coordinates of this joint. Specifically, denote the coordinates of the i-th selected pixel as (x_i, y_i), and the normalized pixel value of this pixel as d_i, where $i \in \{1, 2, 3, 4\}$. We can derive the predicted coordinates of this joint as $(x_{pred}, y_{pred}) = (d_1 \times x_1 + d_2 \times x_2 + d_3 \times x_3 + d_4 \times x_4, d_1 \times y_1 + d_2 \times y_2 + d_3 \times y_3 + d_4 \times y_4)$.

4 Analysis

Most of the *heatmap-based methods* [29, 20, 33, 26, 14, 4, 39, 35, 15, 17, 37] optimize the heatmap prediction via minimizing the pixel-wise MSE loss between the predicted heatmap and the Gaussian-smoothed heatmap. Below we do some analysis about this type of methods.

We denote K the number of joints per input image I. Then we denote $\mathbf{H_{dot}} = \{H^1_{dot}, ..., H^K_{dot}\}$, $\mathbf{H_{Gau}} = \{H^1_{Gau}, ..., H^K_{Gau}\}$, and $\mathbf{H_{pred}} = \{H^1_{pred}, ..., H^K_{pred}\}$ respectively the corresponding K GT dot-annotated heatmaps, GT Gaussian-smoothed heatmaps, and predicted heatmaps of the input image I. We denote $\mathcal{D}_{dot} = \{(I, \mathbf{H_{dot}})\}$ the joint distribution of the input image and the corresponding K dot-annotated heatmaps, and $\mathcal{D}_{Gau} = \{(I, \mathbf{H_{Gau}})\}$ the joint distribution of the input image and the corresponding K Gaussian-smoothed heatmaps. Besides, we denote $l_{MSE}(a, b) = \|a - b\|_2^2$ the pixel-wise MSE loss, and ϕ the model parameters where $\phi(I) = \mathbf{H_{pred}}$. After that, we denote $R(\mathcal{D}_{dot}, \phi, l_{MSE}) = \mathbb{E}_{(I,\mathbf{H_{dot}})\sim\mathcal{D}_{dot}}[l_{MSE}(\phi(I), \mathbf{H_{dot}})]$ as the expected risk calculated between the predicted heatmaps and the GT dot-annotated heatmaps, and $R(\mathcal{D}_{Gau}, \phi, l_{MSE}) = \mathbb{E}_{(I,\mathbf{H_{Gau}})\sim\mathcal{D}_{Gau}}[l_{MSE}(\phi(I), \mathbf{H_{Gau}})]$ as the expected risk calculated between the predicted heatmaps and the Gaussian-smoothed heatmaps.

Theorem 1. *The relationship between $R(\mathcal{D}_{dot}, \phi, l_{MSE})$ and $R(\mathcal{D}_{Gau}, \phi, l_{MSE})$ can be written as*

$$\begin{aligned} R(\mathcal{D}_{dot}, \phi, l_{MSE}) = & R(\mathcal{D}_{Gau}, \phi, l_{MSE}) + 2 \times \mathbb{E}_{(I,\mathbf{H_{Gau}})\sim\mathcal{D}_{Gau}}[\langle \mathbf{H_{pred}}, \mathbf{H_{Gau}} \rangle] \\ & - 2 \times \mathbb{E}_{(I,\mathbf{H_{dot}})\sim\mathcal{D}_{dot}}[\langle \mathbf{H_{pred}}, \mathbf{H_{dot}} \rangle] - C \end{aligned} \tag{6}$$

where $C = \mathbb{E}_{(I,\mathbf{H_{dot}})\sim\mathcal{D}_{dot}}[\|\mathbf{H_{Gau}} - \mathbf{H_{dot}}\|_2^2]$ is a constant.

Table 1: The improvement of AP on COCO validation set when our proposed method is applied to various baselines.

Method	Venue	Backbone	Input size	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}	AR
Hourglass[20]	ECCV 2016	8-Stage Hourglass	256×192	66.9	-	-	-	-	-
CPN[3]	CVPR 2018	ResNet-50	256×192	69.4	-	-	-	-	-
CPN[3]	CVPR 2018	ResNet-50	384×288	71.6	-	-	-	-	-
DarkPose[39]	CVPR 2020	HRNet-W32	384×288	76.6	90.7	82.8	72.7	83.9	81.5
UDP[10]	CVPR 2020	HRNet-W32	384×288	77.8	91.7	84.5	74.2	84.3	82.4
UDP[10]	CVPR 2020	HRNet-W48	384×288	77.8	92.0	84.3	74.2	84.5	82.5
TokenPose[15]	ICCV 2021	TokenPose-L/D24	256×192	75.8	90.3	82.5	72.3	82.7	80.9
Removing Bias[7]	ICCV 2021	ResNet-152	384×288	74.4	-	-	-	-	-
Removing Bias[7]	ICCV 2021	HRNet-W32	256×192	75.8	-	-	-	-	-
Simple Baseline[33]	ECCV 2018	ResNet-152	384×288	75.0	90.8	82.1	67.8	78.3	80.0
+ Ours		ResNet-152	384×288	**76.7(↑1.7)**	**92.1**	**83.6**	**69.7**	**80.0**	**81.3**
HRNet[26]	CVPR 2019	HRNet-W32	384×288	76.7	91.9	83.6	73.2	83.2	81.6
+ Ours		HRNet-W32	384×288	**78.2(↑1.5)**	**92.2**	**84.5**	**74.3**	**84.7**	**82.5**
HRNet[26]	CVPR 2019	HRNet-W48	384×288	77.1	91.8	83.8	73.5	83.5	81.8
+ Ours		HRNet-W48	384×288	**78.8(↑1.7)**	**92.5**	**85.1**	**75.0**	**85.3**	**83.1**
HRFormer[37]	NIPS 2021	HRFormer-Base	384×288	78.0	92.2	84.8	74.3	84.6	82.6
+ Ours		HRFormer-Base	384×288	**78.9(↑0.9)**	**92.6**	**85.4**	**75.3**	**85.3**	**83.3**

The proof of Theorem 1 is provided in the supplementary. As shown in Theorem 1, when we minimize the pixel-wise loss between $\mathbf{H_{pred}}$ and $\mathbf{H_{Gau}}$, while $R(\mathcal{D}_{Gau}, \phi, l_{MSE})$ decreases, the second term $2 \times \mathbb{E}_{(I,\mathbf{H_{Gau}})\sim\mathcal{D}_{Gau}}[\langle \mathbf{H_{pred}}, \mathbf{H_{Gau}} \rangle]$ can increase and the third term $2 \times \mathbb{E}_{(I,\mathbf{H_{dot}})\sim\mathcal{D}_{dot}}[\langle \mathbf{H_{pred}}, \mathbf{H_{dot}} \rangle]$ cannot be guaranteed to increase or decrease. Because of this, $R(\mathcal{D}_{dot}, \phi, l_{MSE})$ cannot be guaranteed to decrease, and the model performance of body joint localization may not be consistently improved during such optimization of heatmap prediction. A more intuitive analysis is as follows. In the optimization process of most of the *heatmap-based methods*, since all the $H \times W$ pixels from the predicted heatmap contribute to the overall pixel-wise loss, learning to fit the other pixels better instead of the pixel representing the dot annotation can also lead to a smaller overall loss, as shown in Fig. 1. Hence, when minimizing the overall pixel-wise loss, the model performance of body joint localization cannot be guaranteed to improve consistently. Besides, during training, the GT Gaussian-smoothed heatmap is constructed via using the Gaussian blob. Therefore, during testing, the predicted heatmap often has a relatively large area of pixels with large values around the dot annotation, as shown in Fig. 3, which can make it difficult to accurately locate the body joint.

Differently, our method optimizes the heatmap prediction from a novel perspective via minimizing the difference between the distribution constructed from the predicted heatmap (i.e., the suppliers S), and the distribution constructed from the dot annotation (i.e., the demanders D). During training, since we formulate the cost function between each pair of supplier and demander as their L2 distance, minimizing the distribution difference based on such a cost function can aggregate the pixel values in the predicted heatmap towards the dot annotation. Hence, our method can help to achieve a more consistent model performance improvement when minimizing the distribution difference. Besides, during testing, equipped with such a mechanism of aggregating the predicted pixel values towards the dot annotation, our method can achieve a more compact body joint localization as shown in Fig. 3.

5 Experiments

To evaluate the effectiveness of our proposed method, we conduct experiments on the COCO dataset [16] and the MPII Human Pose dataset [1]. Besides, to test the generality of our method, we apply it to various backbones, e.g., ResNet [33], HRNet [26], and HRFormer [37].

5.1 COCO Keypoint Detection

Dataset & evaluation metric. The COCO dataset [16] contains more than 200k images and 250k person instances, which are annotated with 17 body joints. This dataset has three subsets including COCO training set, COCO validation set, and COCO test-dev set, which have 57k, 5k and 20k images, respectively. We conduct experiments on this dataset via first training the model on the train2017 set, and then evaluating the model on the val2017 set and test-dev2017 set. Following

Table 2: The improvement of AP on COCO test-dev set when our proposed method is applied to various baselines.

Method	Venue	Backbone	Input size	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}	AR
G-RMI[22]	CVPR 2017	ResNet-101	353×257	64.9	85.5	71.3	62.3	70.0	69.7
Mask-RCNN[9]	ICCV 2017	ResNet-50-FPN	-	63.1	87.3	68.7	57.8	71.4	-
RMPE[6]	ICCV 2017	PyraNet[36]	320×256	72.3	89.2	79.1	68.0	78.6	-
CFN[11]	ICCV 2017	-	-	72.6	86.1	69.7	78.3	64.1	-
CPN[3]	CVPR 2018	ResNet-Inception	384×288	72.1	91.4	80.0	68.7	77.2	78.5
CPN(ensemble)[3]	CVPR 2018	ResNet-Inception	384×288	73.0	91.7	80.9	69.5	78.1	79.0
Integral Pose Regression[27]	ECCV 2018	ResNet-101	256×256	67.8	88.2	74.8	63.9	74.0	-
Posefix[18]	CVPR 2019	ResNet-152	384×288	73.6	90.8	81.0	70.3	79.8	79.0
DarkPose[39]	CVPR 2020	HRNet-W48	384×288	76.2	92.5	83.6	72.5	82.4	81.1
UDP[10]	CVPR 2020	HRNet-W48	384×288	76.5	92.7	84.0	73.0	82.4	81.6
TokenPose[15]	ICCV 2021	TokenPose-L/D24	384×288	75.9	92.3	83.4	72.2	82.1	80.8
Removing Bias[7]	ICCV 2021	HRNet-W48	384×288	76.1	-	-	-	-	81.0
Simple Baseline[33]	ECCV 2018	ResNet-152	384×288	73.8	91.7	81.2	70.3	80.0	79.1
+ Ours		ResNet-152	384×288	**75.3(↑1.5)**	**92.6**	**83.1**	**71.7**	**81.1**	**80.3**
HRNet[26]	CVPR 2019	HRNet-W32	384×288	74.9	92.5	82.8	71.3	80.9	80.1
+ Ours		HRNet-W32	384×288	**76.7(↑1.8)**	**92.6**	**84.0**	**73.0**	**82.8**	**81.5**
HRNet[26]	CVPR 2019	HRNet-W48	384×288	75.5	92.5	83.3	71.9	81.5	80.5
+ Ours		HRNet-W48	384×288	**77.2(↑1.7)**	**93.0**	**84.4**	**73.4**	**83.3**	**82.0**
HRFormer[37]	NIPS 2021	HRFormer-Base	384×288	76.2	92.7	83.8	72.5	82.3	81.2
+ Ours		HRFormer-Base	384×288	**77.2(↑1.0)**	**93.1**	**84.7**	**73.8**	**83.0**	**82.1**

[33, 26, 37], we use standard average precision (AP) calculated based on Object Keypoint Similarity (OKS) to evaluate model performance.

Implementation details. We apply our method to various baselines including Simple Baseline [33], HRNet [26], and HRFormer [37], with their respective backbones including ResNet-152, HRNet-W32, HRNet-W48, and HRFormer-Base. For these baselines, we follow their original learning and optimization configurations for model training. To calculate the Earth Mover's Distance using the Sinkhorn algorithm, we set the Sinkhorn entropic regularization parameter to 1 and the number of Sinkhorn iterations to 1000 in our experiments.

Results. In Tab. 1 and Tab. 2, we report results on the COCO validation and test-dev sets. We observe that after applying our method on various baselines, a significant performance enhancement is achieved, which shows effectiveness of our proposed method. Moreover, we compare our method with other state-of-the-art 2D human pose estimation methods. Our method achieves superior performance compared to these methods, further demonstrating the effectiveness of our method.

5.2 MPII Human Pose Estimation

Dataset & evaluation metric. The MPII dataset [1] contains around 25K images and more than 40k person instances, which are annotated with 16 body joints. We adopt the standard train/val split in [1] to build the MPII training set and validation set, and conduct all the experiments on this dataset via first training the model on the MPII training set, and then evaluating it on the MPII validation set. Following [26], we use the head-normalized probability of correct keypoint (PCKh) [1] score as the evaluation metric on this dataset and report the PCKh@0.5 score.

Implementation details. On the MPII dataset, we also apply our method to various methods as our baselines, including Simple Baseline [33] and HRNet [26], with their respective backbones including ResNet-152, HRNet-W32, and HRNet-W48. We follow the original learning and optimization configurations for model training for both Simple Baseline [33] and HRNet [26]. Besides, same as the experiments on the COCO dataset, we also set the Sinkhorn entropic regularization parameter to 1 and the number of Sinkhorn iterations to 1000 in our experiments on the MPII dataset.

Results on the MPII validation set. In Tab. 3, we report the results on the MPII validation set. As shown, applying our proposed method on various baselines results in a consistent performance improvement, which demonstrates the effectiveness of our proposed method.

5.3 Ablation Studies

We conduct ablation studies on the COCO validation set via applying our proposed method on HRNet-W48 [26].

Table 3: The improvement of AP on MPII validation set when our proposed method is applied to various baselines.

Method	Venue	Backbone	Input size	Mean	Hea	Sho	Elb	Wri	Hip	Kne	Ank
UDP[10]	CVPR 2020	HRNet-W32	256×256	90.4	97.4	96.0	91.0	86.5	89.1	86.6	83.3
DarkPose[39]	CVPR 2020	HRNet-W32	256×256	90.6	97.2	95.9	91.2	86.7	89.7	86.7	84.0
TokenPose[15]	ICCV 2021	TokenPose-L/D6	256×256	90.1	97.1	95.9	91.0	85.8	89.5	86.1	82.7
TokenPose[15]	ICCV 2021	TokenPose-L/D12	256×256	90.1	97.2	95.8	90.7	85.9	89.2	86.2	82.3
TokenPose[15]	ICCV 2021	TokenPose-L/D24	256×256	90.2	97.1	95.9	90.4	86.0	89.3	87.1	82.5
Removing Bias[7]	ICCV 2021	ResNet-152	256×256	89.9	-	-	-	-	-	-	-
Removing Bias[7]	ICCV 2021	HRNet-W32	256×256	90.6	-	-	-	-	-	-	-
Simple Baseline[33]	ECCV 2018	ResNet-152	256×256	89.6	**97.0**	95.9	90.0	85.0	**89.2**	85.3	81.3
+ Ours		ResNet-152	256×256	**90.3(↑0.7)**	**97.0**	**96.1**	**90.6**	**86.1**	**89.2**	**86.7**	**83.1**
HRNet[26]	CVPR 2019	HRNet-W32	256×256	90.4	97.1	95.9	90.7	86.1	89.4	86.9	83.2
+ Ours		HRNet-W32	256×256	**90.9(↑0.5)**	**97.3**	**96.2**	**91.2**	**86.8**	**90.1**	**87.4**	**84.1**
HRNet[26]	CVPR 2019	HRNet-W48	256×256	90.5	96.9	96.0	90.9	86.2	89.6	87.1	83.5
+ Ours		HRNet-W48	256×256	**90.9(↑0.4)**	**97.1**	**96.3**	**91.2**	**87.0**	**90.2**	**87.5**	**84.2**

Table 4: Evaluation on the effectiveness of formulating the demanders D involving the idea of sub-pixel resolution.

Method	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}	AR
Baseline(HRNet-W48)	77.1	91.8	83.8	73.5	83.5	81.8
Naive demanders formulation	77.9	92.5	84.8	74.5	83.9	82.4
Sub-pixel demanders formulation	78.8	92.5	85.1	75.0	85.3	83.1

Table 5: Evaluation on the number of Sinkhorn iterations.

Method	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}	AR
Baseline(HRNet-W48)	77.1	91.8	83.8	73.5	83.5	81.8
500 Sinkhorn iterations	78.3	92.4	84.9	74.8	84.4	82.7
1000 Sinkhorn iterations	78.8	92.5	85.1	75.0	85.3	83.1
1500 Sinkhorn iterations	78.7	92.4	85.2	74.8	85.4	83.0

Impact of involving the idea of sub-pixels in formulating D. In our proposed method, we formulate the demanders D by involving the idea of sub-pixel resolution. To investigate the impact of formulating the demanders D in such a way, we compare our proposed method (**sub-pixel demanders formulation**) with a variant (**naive demanders formulation**). This variant still formulates the suppliers S and the cost function C in the same way, but formulates the demanders D naively as a single demander at the center of the pixel containing the dot annotation, as shown in Fig. 2(a). As shown in Tab. 4, our proposed method consistently outperforms this variant, which shows effectiveness of our **sub-pixel demanders formulation**.

Impact of the number of Sinkhorn iterations. For measuring the Earth Mover's Distance utilizing the Sinkhorn algorithm, we need to set the number of Sinkhorn iterations, which we set to 1000 in our experiments. We evaluate other choices of the number of Sinkhorn iterations in Tab. 5. As shown, all variants outperform the baseline method, and after the number of Sinkhorn iterations becomes larger than 1000, the model performance becomes stabilized. Hence, we set the number of Sinkhorn iterations to be 1000 in all our experiments.

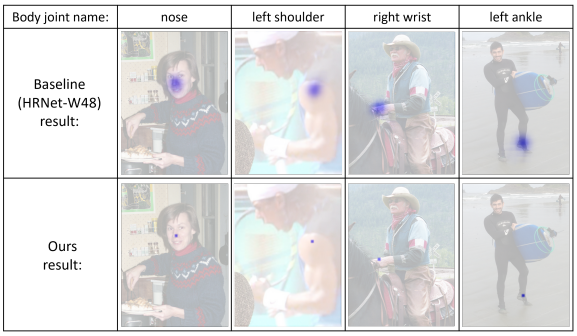

Figure 3: Qualitative results of our method formulating the optimization of the heatmap prediction as a distribution matching problem and the baseline method [26] using the Gaussian-smoothed heatmap as the optimization objective. As shown, our method localizes body joints much more compactly.

Qualitative results. Some qualitative results are shown in Fig. 3. As shown, via formulating the cost function as the L2 distance, our proposed method can aggregate the pixel values in the predicted heatmap towards the dot annotation, and thus localize body joints much more compactly than the baseline method [26] relying on the Gaussian-smoothed heatmap. This demonstrates that our method can effectively optimize the model in the direction of accurately localizing the body joint.

6 Conclusion

In this paper, from a novel perspective, we formulate the optimization of the heatmap prediction as a distribution matching problem between the predicted heatmap and the dot annotation via calculating their Earth Mover's Distance. Our proposed method is simple yet effective, and can be easily applied to various 2D human pose estimation models. Our method achieves superior performance on the COCO dataset and the MPII dataset.

References

[1] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *Proceedings of the IEEE Conference on computer Vision and Pattern Recognition*, pages 3686–3693, 2014.

[2] Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human pose estimation with iterative error feedback. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4733–4742, 2016.

[3] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7103–7112, 2018.

[4] 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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5386–5395, 2020.

[5] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

[6] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. Rmpe: Regional multi-person pose estimation. In *Proceedings of the IEEE international conference on computer vision*, pages 2334–2343, 2017.

[7] Kerui Gu, Linlin Yang, and Angela Yao. Removing the bias of integral pose regression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11067–11076, 2021.

[8] Kerui Gu, Linlin Yang, and Angela Yao. Dive deeper into integral pose regression. In *International Conference on Learning Representations*, 2022.

[9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[10] Junjie Huang, Zheng Zhu, Feng Guo, and Guan Huang. The devil is in the details: Delving into unbiased data processing for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5700–5709, 2020.

[11] Shaoli Huang, Mingming Gong, and Dacheng Tao. A coarse-fine network for keypoint localization. In *Proceedings of the IEEE international conference on computer vision*, pages 3028–3037, 2017.

[12] Siamak Khademi, Ahmad Darudi, and Zahra Abbasi. A sub pixel resolution method. *arXiv preprint arXiv:1211.2221*, 2012.

[13] Jiefeng Li, Siyuan Bian, Ailing Zeng, Can Wang, Bo Pang, Wentao Liu, and Cewu Lu. Human pose regression with residual log-likelihood estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11025–11034, 2021.

[14] Wenbo Li, Zhicheng Wang, Binyi Yin, Qixiang Peng, Yuming Du, Tianzi Xiao, Gang Yu, Hongtao Lu, Yichen Wei, and Jian Sun. Rethinking on multi-stage networks for human pose estimation. *arXiv preprint arXiv:1901.00148*, 2019.

[15] Yanjie Li, Shoukui Zhang, Zhicheng Wang, Sen Yang, Wankou Yang, Shu-Tao Xia, and Erjin Zhou. Tokenpose: Learning keypoint tokens for human pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11313–11322, 2021.

[16] 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.

[17] Zhengxiong Luo, Zhicheng Wang, Yan Huang, Liang Wang, Tieniu Tan, and Erjin Zhou. Rethinking the heatmap regression for bottom-up human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13264–13273, 2021.

[18] Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Posefix: Model-agnostic general human pose refinement network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7773–7781, 2019.

[19] Amit Moryossef, Ioannis Tsochantaridis, Roee Aharoni, Sarah Ebling, and Srini Narayanan. Real-time sign language detection using human pose estimation. In *European Conference on Computer Vision*, pages 237–248. Springer, 2020.

[20] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, pages 483–499. Springer, 2016.

[21] Aiden Nibali, Zhen He, Stuart Morgan, and Luke Prendergast. Numerical coordinate regression with convolutional neural networks. *arXiv preprint arXiv:1801.07372*, 2018.

[22] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4903–4911, 2017.

[23] Hanyu Peng, Mingming Sun, and Ping Li. Optimal transport for long-tailed recognition with learnable cost matrix. In *International Conference on Learning Representations*, 2022.

[24] Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. The earth mover's distance as a metric for image retrieval. *International journal of computer vision*, 40(2):99–121, 2000.

[25] Samuel Schulter, Paul Vernaza, Wongun Choi, and Manmohan Chandraker. Deep network flow for multi-object tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6951–6960, 2017.

[26] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5693–5703, 2019.

[27] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *Proceedings of the European conference on computer vision (ECCV)*, pages 529–545, 2018.

[28] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[29] Jonathan J Tompson, Arjun Jain, Yann LeCun, and Christoph Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. *Advances in neural information processing systems*, 27, 2014.

[30] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1653–1660, 2014.

[31] Boyu Wang, Huidong Liu, Dimitris Samaras, and Minh Hoai Nguyen. Distribution matching for crowd counting. *Advances in Neural Information Processing Systems*, 33:1595–1607, 2020.

[32] Fangyun Wei, Xiao Sun, Hongyang Li, Jingdong Wang, and Stephen Lin. Point-set anchors for object detection, instance segmentation and pose estimation. In *European Conference on Computer Vision*, pages 527–544. Springer, 2020.

[33] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *Proceedings of the European conference on computer vision (ECCV)*, pages 466–481, 2018.

[34] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Thirty-second AAAI conference on artificial intelligence*, 2018.

[35] Sen Yang, Zhibin Quan, Mu Nie, and Wankou Yang. Transpose: Towards explainable human pose estimation by transformer. *arXiv e-prints*, pages arXiv–2012, 2020.

[36] Wei Yang, Shuang Li, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. Learning feature pyramids for human pose estimation. In *proceedings of the IEEE international conference on computer vision*, pages 1281–1290, 2017.

[37] Yuhui Yuan, Rao Fu, Lang Huang, Weihong Lin, Chao Zhang, Xilin Chen, and Jingdong Wang. Hrformer: High-resolution transformer for dense prediction. 2021.

[38] Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12203–12213, 2020.

[39] Feng Zhang, Xiatian Zhu, Hanbin Dai, Mao Ye, and Ce Zhu. Distribution-aware coordinate representation for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7093–7102, 2020.

[40] Yizhai Zhang, Kuo Chen, Jingang Yi, and Liu Liu. Pose estimation in physical human-machine interactions with application to bicycle riding. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3333–3338. IEEE, 2014.

[41] Lu Zhou, Yingying Chen, Yunze Gao, Jinqiao Wang, and Hanqing Lu. Occlusion-aware siamese network for human pose estimation. In *European Conference on Computer Vision*, pages 396–412. Springer, 2020.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [No]
   (c) Did you discuss any potential negative societal impacts of your work? [No]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

   (a) If your work uses existing assets, did you cite the creators? [Yes]

   (b) Did you mention the license of the assets? [Yes]

   (c) Did you include any new assets either in the supplemental material or as a URL? [No]

   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No]

   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No]

5. If you used crowdsourcing or conducted research with human subjects...

   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]