IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 18 May 2024, accepted 12 July 2024, date of publication 17 July 2024, date of current version 26 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3429531

RESEARCH ARTICLE

MLE-Loss Driven Robust Hand Pose Estimation

XUDONG LOU, XIN LIN, XIANGXIAN ZHU, AND CHEN CHEN
Ningbo Preh Joyson Automotive Electronics Company Ltd., Ningbo 315100, China
Zhejiang Key Laboratory of Automotive Electronics Intelligence, Ningbo 315100, China
Corresponding author: Xiangxian Zhu (steven.zhu@preh.cn)

ABSTRACT This paper introduces a novel method for accurately estimating the 2D coordinates of hand keypoints from single static images, utilizing a sequential convolutional neural network optimized with Maximum Likelihood Estimation Loss. Unlike traditional heatmap-based techniques, our approach eliminates the need to generate label heatmaps and sidesteps the direct optimization of model parameters based on noisy labels. Instead, it concentrates on modeling the distribution of the discrepancies between predicted results and ground truth, rather than the potential presence of noisy labels, thus enabling the direct prediction of hand keypoint coordinates. Furthermore, we propose a sequential training and inference framework that consists of a deep convolutional backbone network and a multi-stage sequential network. Each stage of this network features similar structures, facilitating the progressive and precise prediction of hand keypoint coordinates. Our extensive experimental results demonstrate that our approach is both highly accurate and robust, outperforming mainstream methods under the experimental conditions detailed in this paper.

INDEX TERMS Hand pose estimation, maximum likelihood estimation, heatmap, deep learning.

I. INTRODUCTION

Hand pose estimation plays a crucial role in computer vision and finds widespread applications across various domains such as intelligent cockpits [1], [2], AR/VR [3], [4], [5], game control [6], and facilitating air gesture human-computer interaction. However, achieving accurate hand pose estimation based on a single RGB image presents significant challenges. These challenges stem from the variability in the scale and shape of hands in 2D images, as well as the inherent simi-larities in features among different fingers [7], [8]. Moreover, the intricate multi-degree-of-freedom movements of hands [9] often result in occluded parts, further complicating the estimation process. In summary, developing a hand pose estimation method that seamlessly combines accuracy and robustness remains a formidable challenge in the field of computer vision.

In the realm of pose estimation, heatmap-based methodologies [10], [11] have garnered widespread acclaim for their ability to deliver high levels of accuracy. Despite their efficacy, these methods face certain challenges. The heatmap loss, a staple in supervised training for heatmap methods, involves the generation of estimated likelihood heatmaps by the model. An increase in the resolution of these heatmaps results in a proportional rise in the number of parameters and computational complexity. To ascertain the precise coordinates, the argmax method is employed in post-processing to identify the point exhibiting the highest response within the likelihood heatmap. This non-differentiable discrete operation contravenes the end-to-end principle and introduces quantization errors into the estimation outcomes.

Heatmap-based pose estimation techniques have shown remarkable proficiency in managing situations with ambiguous labels, such as in-stances of occlusion, obstruction, and blurriness [12]. These methods employ a form of soft labeling by producing likelihood heatmaps from coordinate data in labels, facilitating a progressive smoothing regression process [13]. The likelihood heatmaps, derived from empirically predetermined distributions, offer an effective way to model label ambiguity. This enhancement im-proves the model's comprehension and pro-cessing abilities in the presence of ambiguous labels. Nevertheless, it is crucial to acknowledge that this approach involves learning the

The associate editor coordinating the review of this manuscript and approving it for publication was Junchi Yan.

distributions of labels with noise. The noise in labels may become particularly evident due to human annotation errors or technical constraints, ultimately making the model susceptible to noise and negatively impacting its performance in real-world fuzzy scenarios.

Furthermore, traditional regression paradigms commonly employ the or loss [12], [14] to quantify residuals, predicated on the assumption that the data conform to either a Laplace or Gaussian distribution. Nevertheless, the task of precisely modeling the true distribution via a simplistic form presents substantial challenges in real-world scenarios. This divergence between the anticipated distribution and the actual distribution impinges upon the sensitivity of the loss function to residuals, thus constraining the efficacy of the model.

In this study, we objective is to tackle the limitations inherent in existing heatmap methods. To achieve this, we introduce a novel 2D hand pose estimation framework with superior performance, comprising a sequential heatmap model and several flow models. The heatmap model consists of a deep convolutional backbone network and a sequential network consisting of multiple stages, each features similar structures. This design facilitates the propagation of contextual features be-tween adjacent stages through heatmaps.

As the stages progress, the model parameters gradually acquire the ability to learn complex implicit associations between the key parts of the hands in the provided image. This allows the model to capture intricate relationships and dependencies among these hand parts. To address the potential quantization errors that may arise during the process, we propose an alternative approach to compute the coordinates from the heatmap. Instead of using the argmax operation, we employ integral regression. This helps to miti-gate the impact of quantization errors and pro-vides more accurate and precise estimation of the hand pose. Furthermore, in order to improve the model's understanding of the underlying data distribution and alleviate the influence of label noise, we incorporate a flow model to characterize the deviation between the ground truth and the provided labels. In terms of optimization, we formulate a loss function based on the maximum log-likelihood estimation (MLE) principle. This loss function serves as a guiding force during the training phase, enabling the model to converge towards better hand pose estimation, as depicted in Figure 1. Experimental results demonstrate that our method achieves notable precision and robustness in estimating hand pose. In summary, the main contributions of this paper can be outlined as follows:

1. Introduction of a sequential heatmap model, which is designed to incorporate global feature fusion and context feature propagation mecha-nisms. The accuracy of prediction improves as the stage progresses.
2. Estimation and modeling of probability density distribution of deviations between ground truth and predicted outcomes. Additionally, a new re-gression paradigm based on MLE is utilized to mitigate the impact of label noise and enhance parameter training.
3. Design of a comprehensive loss function to achieve end-to-end training of the model while maintaining training efficiency and supplement-ing the backpropagation gradients.

We have conducted extensive experiments on two challenging hand pose datasets, namely Frei-hand [15] and RHD [16], to evaluate the effec-tiveness of our approach. The experimental re-sults demonstrate that our proposed MLE-Loss Driven Robust Hand Pose Estimation method outperforms mainstream methods on both da-tasets. Our method exhibits robustness and accu-racy even under complex and variable conditions.

II. RELATED WORKS

In the nascent phases of hand pose estimation research, scholars predominantly concentrated on modeling the spatial interrelations among different hand joints [17], or on the classification of hand shapes and joints [18], [19] to deduce ges-tures and their motion dynamics. Nonetheless, these methods frequently demonstrated a degree of fragility, and were heavily reliant on strong prior assumptions. Such dependencies frequently culminated in their underperformance or outright failure to yield satisfactory results when applied beyond the confines of controlled experimental settings.

In contrast, deep learning methods based on convolutional architectures [20], [21] directly capture implicit spatial relationships between different parts through the training inference process, often leading to commendable results. Further-more, some researchers have enhanced the robustness of their techniques by integrating the topological relationship of motion chains or co-ordinates as constraints within their models. For example, Chen et al. [22] leveraged the topological structure of hand joints as a constraint to re-fine the regression of hand joint positions in hand pose estimation from single depth images. In a similar vein, Zhou et al. [23] estimated hand model parameters via a convolutional neural network (CNN) and deduced the hand pose through the application of forward kinematics. In an effort to enrich the model input, Choi et al. [24] incorporated geometric characteristics as additional modalities and applied multi-task learning to enhance hand pose estimation. Likewise, Zheng et al. [25] employed vectors in lieu of joint coordinates within a finger-to-hand regression framework to achieve more consistent estimation outcomes. Addressing the pervasive issue of joint occlusion, certain studies have explored the use of multi-view RGB models; for instance, Simon et al. [26] reconstructed occluded information and augmented detection capabilities through multi-view reprojection and triangulation techniques. Ge et al. [27] executed 3D gesture pose estima-tion by applying CNN models to multiple RGB views. While these innovative methods have advanced the handling of occlusion issues, they have concurrently introduced additional deployment costs.

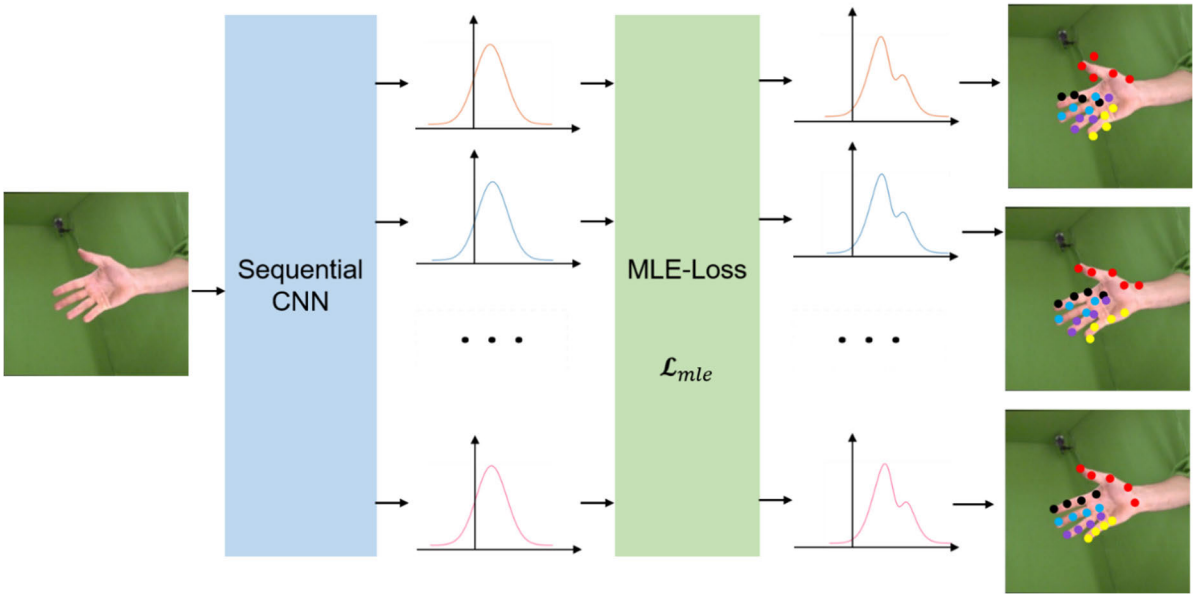


FIGURE 1. Process of estimating hand pose using MLE-Loss.

A. HEATMAP-BASED METHODS

In the realm of pose estimation, heatmap-based methods are universally recognized as the leading strategies in current practice. Wei et al. [28] developed a progressive convolutional neural net-work framework that excels in precise pose esti-mation. An et al. [29] introduced an expedited top-down hand pose estimation architecture that calculates joint heatmaps for the hand. Tompson et al. [30] developed an innovative architecture that incorporates an efficient position refinement model, which is trained using heatmaps to estimate the joint offset positions within diminutive image regions. Nonetheless, a common challenge encountered in the aforementioned methodologies is the utilization of the non-differentiable argmax operation for coordinate recovery, which hinders the capacity for end-to-end model train-ing and introduces quantization errors that are inherently tied to the heatmap resolution. To ad-dress this limitation, Sun et al. [31] and Luvizon et al. [32] proposed the utilization of smooth, differentiable soft-argmax functions for coordinate regression from the heatmap. These ap-proaches enable end-to-end training of the model, although the regression accuracy remains contingent on heatmap resolution.

B. REGRESSION-BASED METHODS

Regression-based approaches are designed to directly deduce the mapping relationship from images to keypoint coordinates or alternative parameters, with the ultimate aim of pinpointing each joint's position. These approaches are favored for their succinct representation and typically result in reduced computational demands. For instance, Gomez-Donoso et al. [33] engineered a convolutional neural network for the regression of joint positions from single RGB images. Toshev and Szegedy [34] innovated a regression-based estimator using Deep Neural Networks (DNNs) for achieving high-precision in pose estimation tasks. Zhang et al. [35] introduced Mediapipe Hands, which employs a two-stage regression process for the detection of hands and subsequent identification of 21 keypoint coordinates. The use of regression methods is particularly prevalent in 3D pose estimation research, where their lightweight computational footprint is a significant advantage over methods typically employed for 2D pose estimation. Chen et al. [36] enhanced their model with a semantic segmentation sub-network, which assigns semantic labels to input point clouds before regressing the hand pose, thereby integrating high-level understanding with precise localization. Ge et al. [37] exploited a 3D deep network architecture to regress hand joint coordinates directly. Zimmerman and Brox [16] took a unique approach by estimating 2D coordinates through a regression network and converting them into 3D coordinates using the PosePrior network.

Overall, regression-based methods provide dis-tinct advantages in terms of computational efficiency. However, these methods often fall short of the accuracy achieved by heatmap-based methods. Regression methods are typically more suitable for computationally intensive tasks such as 3D pose estimation [38] or for deployment on resource-constrained edge devices. Conversely, heatmap-based methods excel in performance but may lack end-to-end training capability and face challenges in achieving a balance between computational accuracy and resources constraints.

In our work, we employ a heatmap-based approach for our model due to its superior accuracy. Our approach incorporates a distinctive strategy, wherein our model employs differentiable integration functions to regress coordinates,

while simultaneously preserving the heatmap as an intermediate output. This innovative technique al-lows us to effectively supervise the training of our model by utilizing the coordinate information from the labels, thereby obviating the requirement to generate supplementary label heatmaps.

III. METHOD

In this paper, we introduce an accurate and robust 2D hand pose estimation framework. Our framework consists of a sequential heatmap model augmented by several flow model components. The heatmap model comprises two primary components: a deep convolutional backbone network responsible for global feature extraction, and a sequential network consisting of multiple stages, each features similar structures. The global features extracted by the backbone network are shared as the same input across all stages. In the first stage, only global features are utilized as input, and in subsequent stages, the heatmaps of the current stage are generated by combining the global features and the heatmaps generate in the previous stage. Each stage includes shallow head networks to estimate keypoint coordinates $\hat{\mu}$ and their variance $\hat{\sigma}$ from contextual information, evaluating the deviation ϵ between the ideal coordinates μ and the predicted coordinates $\hat{\mu}$. Besides, at each stage, we use the flow model to fit the probability density function $P(\epsilon|I)$ of the deviation ϵ conditioned on the input image I and calculate the loss value for the current stage through MLE. Contextual feature information is transmitted between adjacent stages through likelihood heat-maps resulting in progressively refined probability estimates of key parts distribution. The overall loss function $\mathcal{L}_{mle}$ is derived by aggregating the loss $\mathcal{L}_t$ of each stage, facilitating end-to-end joint training of all stages within the heatmap model and flow models. This approach enables supervision of the overall model parameters and supplementing backward gradients.

In the subsequent sections, we will begin by introducing the design of the sequential heatmap model. Following that, we will explore the design of the MLE loss and elucidate its principles in enhancing hand pose estimation. Finally, we will discuss some crucial implementation details.

A. SEQUENTIAL HEATMAP MODEL DESIGN

In the field of hand pose estimation, leveraging deep convolutional neural networks has emerged as an efficient strategy due to their expansive receptive fields. These networks enable the capture of intricate implicit relationships among various joints depicted in images. Furthermore, em-ploying a sequential architectural promotes the flow and sharing of information among different stages of the model. This facilitates the enhanced utilization of features acquired in earlier stages [39] during subsequent phases, thereby improving overall performance. However, as the depth of model increases, certain side effects occur, including an increase in the number of parameters and an elevated risk of gradient vanishing during training. To address these challenges, the implementation of the global feature concatenation mechanism proves to be an effective strategy. This mechanism merges the heatmap output from the previous stage with the global feature. By doing so, richer contextual information is introduced to the current stage of the network. Additionally, this approach establishes a direct gradient feedback path, thereby aiding in alleviating the issue of gradient vanishing in deep network. Through the adoption of this design, both the ac-curacy and the robustness of the model can be effectively enhanced.

1) BACKBONE NETWORK DESIGN

The backbone network utilized in our model is a customized modification based on ResNet-50 [40]. While retaining most of the original structure of ResNet-50, we removed its classification head, prioritizing the extraction of high-level features essential for estimating keypoints on the hands. Consequently, the output feature map of this backbone network undergoes 1/32 downsampling, with 2048 channels. These high-level features are regarded as global features, and their weights are shared across all stages of the model.

2) SEQUENTIAL NETWORK DESIGN

In order to estimate the coordinates and their uncertainties of key hand parts, we devise a shallow head architecture with dual output heads at each stage of the sequential network. These two output heads are responsible for generating a heatmap describing the probability distribution of hand keypoint coordinates, and estimating its variance.

When processing the input feature maps, we initially reduce the number of channels of the feature map to 256 using a 1×1 convolutional layer. This helps decrease computational complexity while preserving sufficient information. Subsequently, to seamlessly integrate with the heatmaps generate in the previous stage, we employ an average pooling layer to uniformly adjust the resolution of the heatmap to 8×8, and another 1×1 convolutional layer is applied to adjust its channels to 256. This preprocessing step ensures that both the shared feature maps and the heatmaps have a balanced influence at the current stage, as they maintain a consistent number of channels. The sharing mechanism of global features ensures that the model can access the same high-level visual information in each stage, which is crucial for effective feature fusion and information propagation within each stage.

After the preprocessing of inputs, the concatenated feature maps are entered into two specialized head, the heatmap head and the variance head. The heatmap head utilizes a pure convolutional network architecture and incorporates three layers of deconvolutional operations. Each layer applies a 4×4 convolutional kernel with a stride of 2 to upsample the feature map until it reaches the target resolution of 64×64. At the same time, the number of channels is adjusted to K, resulting in K heatmaps (where $K = 21$). These heatmaps represent the different types of hand keypoints that to be detected. On the other hand, the input features were compressed into one-dimensional data after being processed

by an adaptive average pooling layer. These data were subsequently input into the variance head, which consists of two fully connected layers, to predict $2K$ values of variance corresponding to the variance in the x and y directions for each keypoint. During actual operation, all convolutional and fully connected layers, except for the last layer, are followed by a batch normalization layer and ReLU activation function, aiming to prevent issues like vanishing gradients and to accelerate the convergence speed of the model.

The predicted coordinates $\hat{\mu}_k$ of the hand joints for the k-th type of hand are calculated through integral regression of the heatmap H_k. This method involves weighted summation of the pixel values in the heatmap to obtain the position estimation of each hand joint, as shown below:

$$\hat{\mu}_k = \sum_{p_y=1}^{H} \sum_{p_x=1}^{W} p \bullet \tilde{H}_k(p) \tag{1}$$

where H and W represent the height and width of the heatmap, p represents an arbitrary position (p_x, p_y) in the domain ω of the heatmap H_k. $\tilde{H}_k$ is the normalized heatmap within the domain ω, and $\tilde{H}_k(p) = \frac{e^{H_k(p)}}{\int_{q\in\omega} e^{H_k(q)}}$, where all elements are non-negative and sum up to 1, and q represents an arbitrary position within ω but independent of p. The integration method facilitates a smooth and differentiable process for mapping the heatmap to coordinates. Consequently, this characteristic makes it possible to train the model directly via backpropagation, and avoids the consideration of quantization errors.

B. LEARN WITH MLE LOSS

1) BASIC HEATMAP LOSS

The most prevalent heatmap losses usually involve calculating the first-order or second-order distances between the estimated heatmap H and the target heatmap H_g, followed by the calculation of the ℓ_1 or ℓ_2 loss. Taking the ℓ_2 loss as an example, it can be represented by the following equation:

$$f = \sum_{k=1}^{K} \sum_{p\in\omega} \|H^k(p) - H_g^k(p)\|_2^2 \tag{2}$$

where ω denotes the domain of the heatmap, and p represents any position in the heatmap. This equation quantifies the overall disparity between the estimated heatmap H^k and the target heatmap H_g^k by evaluating the ℓ_2 distance across all pixel positions between each keypoint class. The target heatmap H_g is conventionally constructed through a Gaussian kernel function applied to the label coordinates. This strategy employs soft labeling techniques, fostering a gradual, smooth regression process aimed at mitigating the influence of the noise of labels. However, fundamentally, this method still induces the model parameters to conform the distribution of labels with noise.

In addition, various forms of regression paradigms stem from maximum likelihood estimation of the expected distribution of the estimated state. The ℓ_2 loss assumes that the target distribution follows a standard Gaussian distribution, which is a strong assumption in its design. However, in real scenarios, simple probability density functions often fail to fully capture the distribution of real data. The purpose of the loss function is to quantify the difference between the predictions generated by the model and the actual data. Obviously, when we can more accurately estimate the true probability distribution $P(\mu)$ of key coordinates μ, the loss function becomes more sensitive to deviations between model predictions and reality. As a result, it provides more accurate parametric feedback during training.

2) MLE LOSS DESIGN

Building on the previous discussion, we aim to build an expected distribution model that is more in line with the actual data distribution than a simple Laplace or Gaussian distribution. This endeavor seeks to formulate a more effective loss function conducive to the model's understanding of the real data distribution. Simultaneously, under the assumption that the perfect labeling is impossible, our goal is for the model to be able to estimate the distribution of label $\hat{\mu}_g$ around the ground truth μ. In other words, we aim to estimate the distribution of deviations rather than the distribution of $\hat{\mu}_g$ itself. By doing so, we mitigate the influence of the noise of labels and address the issue of weight dispersion during training, as the distribution of μ_g varies with different inputs I. Conversely, the distribution of deviations between the labels and the ground truth tends to exhibit greater stability, rendering it more amenable to neural network fitting. We define the estimated deviation as ϵ, which quantifies the disparity between the label and the ground truth. The impact of deviation ϵ on the coordinates is expressed as follows:

$$\epsilon = \frac{\mu - \hat{\mu}}{\hat{\sigma}} \tag{3}$$

where μ represents the ideal coordinates to be estimated, $\hat{\mu}$ represents the initial coordinates estimated by the heatmap model, and $\hat{\sigma}$ represents the variance corresponding to the initial coordinates. When formulating the loss function, we follow the methodology proposed by Li et al. [41]. The central concept revolves around estimating the distribution of deviations ϵ utilizing a flow model. This entails learning intricate and unknown true distributions by employing smooth reversible mappings of simple predefined distributions. Within the implicit space, we posit that the random variable z adheres a straightforward initial distribution: $P(z) = \mathcal{N}(0, 1)$. The flow model defines a smooth reversible mapping: $f_\phi : \mathbb{R}^2 \to \mathbb{R}^2$, where ϕ represents the learnable parameters in the flow model. Through the mapping f_ϕ, we can transform the random variable z, characterized by a known distribution in the implicit space, to the random variable ϵ to be estimated in the real space: $\epsilon = f_\phi(z)$, and we have $z = f_\phi^{-1}(\epsilon)$. With a well-trained and understood reversible mapping f_ϕ, we can ascertain the probability density distribution of ϵ, denoted as $P(\epsilon|I)$. Given the input image I, we construct the loss function

$\mathcal{L}$ by maximizing the probability of the predicted coordinate μ being at the true label μ_g:

$$\begin{aligned}\mathcal{L} &= -logP_{\theta,\phi}(\mu|I)|_{\mu=\mu_g} \\ &= -logP_{\phi}(\epsilon_g|I) - log(det\frac{d\epsilon_g}{d\mu_g}) \\ &= -logP_{\phi}(\epsilon_g|I) + log\hat{\sigma}\end{aligned} \tag{4}$$

where $\epsilon_g = (\mu_g - \hat{\mu})/\hat{\sigma}$, represents the normalized deviation between the true label μ_g and the predicted coordinate $\hat{\mu}$ with respect to the estimated variance $\hat{\sigma}$. The learning process aims to minimize the value of the loss function to optimize the parameters θ of the heatmap model and the parameters ϕ of the flow model. The specific form of the loss function is adaptive and depends on $P_\phi(\epsilon|I)$, which is determined by the parameters ϕ of the flow model.

During training, both θ and ϕ are optimized simultaneously, with ϕ gradually converging to stability throughout the training process. For different input images I, the heatmap model estimates different values of $\hat{\mu}$ and $\hat{\sigma}$, leading to a distribution of μ that belongs to a density function family with varying means and variances.

3) TRAINING WITH MLE LOSS

Our pipeline consists of a heatmap convolutional network and multiple flow models. The heatmap model is responsible for extracting high-level features from input images and estimating the deviation between the predicted results and ground truth at different stages. At stage-t, the deviation ϵ_t is evaluated based on the estimated key coordinates $\hat{\mu}_t$ and its variance $\hat{\sigma}_t$ from the current stage. The probability density function $P_t(\epsilon|I)$ of the deviation is fitted through the flow model of the current stage, and the current loss function $\mathcal{L}_t$ is constructed by maximizing the probability of the predicted coordinates μ_t being at μ_g in the current stage:

$$\begin{aligned}\mathcal{L}_t &= -logP_{\theta,\phi}(\mu_t|I)|_{\mu_t=\mu_g} \\ &= -logP_{\phi}(\epsilon_{tg}|I) + log\hat{\sigma}_t\end{aligned} \tag{5}$$

where $\epsilon_{tg} = (\mu_g - \hat{\mu}_t)/\hat{\sigma}_t$. At each stage-$t(t > 1)$, the shallow head network takes global features along with the heatmap output from the preceding stage as inputs (with only global features used in stage-1). It then computes estimates for the coordinates $\hat{\mu}_t$ and its variance $\hat{\sigma}_t$. The flow model estimates the distribution of labels around the ground truth ϵ_t based on a predetermined distribution $P(z)$ in the latent space, enabling the determination of the specific probability density function of ϵ_t and facilitating the calculation of the regression paradigm through MLE.

Deeper stages in our model progressively refine predictions towards the ideal coordinates μ, albeit with increasing training complexity. While the global feature concatenation helps alleviate gradient vanishing, training challenges may still arise. To address this, we aggregate stage losses into an overall loss function:

$$\mathcal{L}_{mle} = \sum_{t=1}^{T} \mathcal{L}_t \tag{6}$$

Eq. (6) guiding parameter optimization across the heatmap network and flow models, while gradient supplementation at each stage ensures stable and efficient learning. We choose $T = 5$ to balance model depth and size. Additionally, this overall loss function supervises the outputs of each stage to ensure their meaningfulness. The specific calculation process is illustrated in Figure 3. The effectiveness of the proposed loss function will be further demonstrated in Section IV-B of this paper, along with showcasing its optimization benefits in practical applications.

C. IMPLEMENTATION DETAILS

1) TRAINING AND INFERENCE

The overall loss function $\mathcal{L}_{mle}$ relies on the parameters of both the heatmap model θ and the flow models ϕ_t. Through an end-to-end training strategy, both θ and ϕ_t are optimized simultaneously. In the ideal scenario upon completing training, the deviation ϵ_t gradually decreases. Consequently, during inference, only the heatmap model needs to be invoked without running the flow models, and the estimated initial coordinates $\hat{\mu}_t$ from the heatmap model can be considered as ideal outputs. The pseudocode for training and inference is shown in Algorithm 1.

2) FLOW MODEL

In terms of flow model, we adopt the design of RealNVP [42], which does not directly construct complex mapping functions, but constructs a flexible and easy-to-handle bijection function by superimposing a series of simple bijections, avoiding loss of control in computational complexity, as:

$$x = f_\phi(z) = f_k \circ f_{k-1} \ldots \circ f_1(z) \tag{7}$$

where $\circ$ represents the composition of functions. In each step of the transformation f_k, a portion of the input vector z remains unchanged, while the remaining part is translated or scaled as needed. The entire transformation process is repeated k times, where k is set to 6 in our study.

IV. EXPERIMENT

In this section, we conduct extensive experiments to demonstrate the effectiveness of our proposed approach. We evaluate the performance of our method using two widely used hand pose datasets: FreiHand and RHD.

The Freihand dataset offers a challenging dataset for hand pose and shape estimation from RGB images, comprising 13,240 training samples and 3,960 testing samples. The training set encompasses 32,560 unique real images generated through four distinct post-processing methodologies. We acquire the 2D coordinates of 21 hand keypoints in the image coordinate system by depth normalization, utilizing the

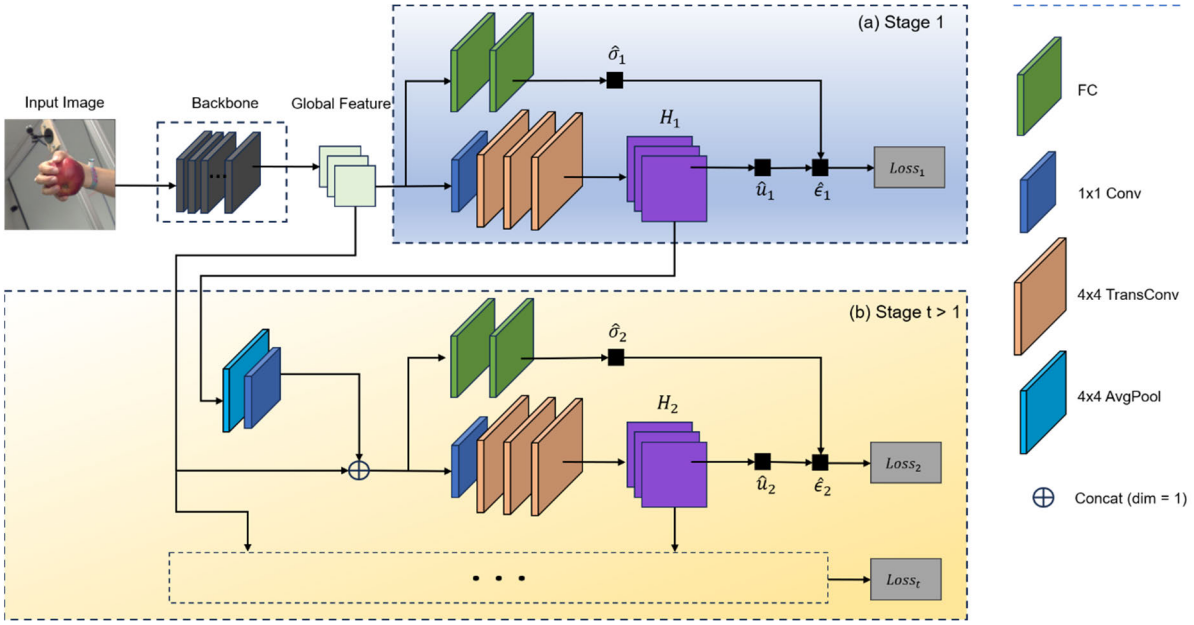


FIGURE 2. Illustration of the heatmap model. The input image size is constrained to 256 × 256, with a ResNet50 backbone fine-tuned for feature extraction. The extracted global features serve as the shared input across all stages. In stage-1, the global features are used to compute the deviation between the predicted coordinates and the ground truth, followed by the calculation of the corresponding MLE loss. In stage-2 and subsequent stages, the input comprises the global features and the heatmap estimated in the previous stage. This process is repeated iteratively throughout the pipeline.

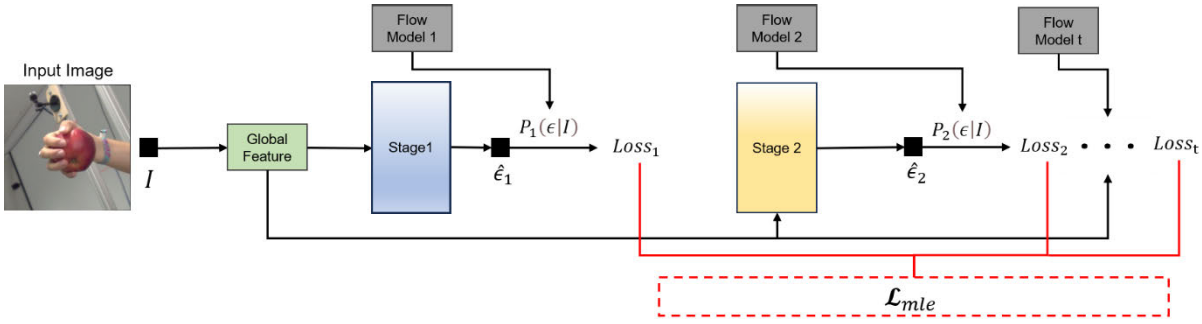


FIGURE 3. Calculation of the overall loss function. In each stage, the probability distribution of the estimated deviation $\hat{\epsilon}_t$ is calculated through the flow model t, and the MLE loss for that stage is obtained by maximizing the probability of the ideal coordinates being located at the label. The overall loss function $\mathcal{L}_{mle}$ is then computed as the sum of losses across all stages.

3D coordinates in the camera coordinate system along with the intrinsic matrix provided by the dataset.

The RHD dataset consists of over 40,000 training samples and 2,728 testing samples, focusing on hand pose estimation. Each sample includes the 2D coordinates of 21 keypoints for both the left and right hands. To integrate it into our training framework, we crop each sample into partial images of the left and right hands based on the bounding boxes of the hands, and assign the corresponding hand keypoint labels to the cropped images. Samples with fewer than 10 visible keypoints are excluded. Following these preprocessing steps and augmentations, we get 60,375 training samples and 3,807 testing samples.

In this section, we first outline our experimental setup, introducing the training procedure and evaluation metrics. Subsequently, we conduct self-comparisons to validate the efficacy of individual components within our model. Finally, we compare our approach to mainstream methods, providing both quantitative and qualitative results to comprehensively demonstrate the performance of our approach.

A. EXPERIMENTAL SETTING

1) TRAINING PROCESS

All experiments are conducted on a workstation equipped with four NVIDIA RTX 3090 GPUs, each with 24GB of memory. We conduct training for over 120 epochs on both the Freihand and RHD datasets. The resolution of all images is set to 256×256. We initialize the learning rate to 1e-3 and decrease it to 1e-4 after 90 epochs, continuing training

Algorithm 1 Pseudocode for Training and Inference

Set: Num of stages N
Initialize: Heatmap model parameters θ, Flow model parameters $\Phi_N // \Phi_N = [\phi_1, \phi_2, \ldots, \phi_N]$
for I, μ_g in data-loader **do**
  if mode == training **then**
    $\mathcal{L}_{mle} = 0$// Initialize MLE loss
    for t in range N **do**
      $\hat{\mu}_t, \hat{\sigma}_t = f_\theta(I)[t]$// Evaluate coordinates and its variance using heatmap model
      $\phi_t = \Phi_N[t]$
      $\epsilon_t = \left(\mu_g - \hat{\mu}_t\right)/\hat{\sigma}_t$// Evaluate deviation to Eq. (3)
      $\mathcal{L}_t = f_{\phi_1}(\epsilon_t)$ // Evaluate MLE Loss in stage-t to Eq.(5)
      $\mathcal{L}_{mle} += \mathcal{L}_t$// Evaluate the overall loss to Eq. (6)
    end for
    $[\theta, \Phi_N]$.update
  else ▷ mode == inference
    $\hat{\mu}_N = f_\theta(I)[N]$// Evaluate estimated coordinates from heatmap model
    return $\hat{\mu}_N$// Directly return estimated coordinates from heatmap model to speed up inference
  end if
end for

until the model reaches convergence. We utilize the Adam optimizer with a batch size of 128. The implementation of the network is based on the PyTorch framework.

2) TRAINING PROCESS

We employ Normalized Mean Error (NME) and Probability of Correct Keypoints (PCK) as evaluation metrics, as shown below:

$$PCK_\sigma = \frac{\sum_{s=1}^{D}\left(\sum_{i=1}^{K}\left(\delta\left(\frac{||x_{si}-\hat{x}_{si}||}{K\times max(w,h)} <= \sigma\right)\right)\right)}{D} \tag{8}$$

$$NME = \frac{\sum_{s=1}^{D}\left(\sum_{i=1}^{K}\left(\frac{||x_{si}-\hat{x}_{si}||}{K\times max(w,h)}\right)\right)}{D} \tag{9}$$

where the term x_{si} represents the ground truth coordinates of landmark i, and $\widehat{x}_{si}$ corresponds to the predicted coordinate by the model. Here, i signifies the index of hand landmarks, encompassing a total of 21 types of hand joints ($K = 21$), and s indicates the index of the sample within the dataset. D represents the total number of samples contained in the dataset. In Eq. (8), δ is defined as the indicator function, which is set to 1 in our experiments, and σ represents the normalized distance threshold. The variables w and h stand for the width and height of the hand bounding box.

B. SELF COMPARISONS

We first evaluate the effectiveness of the different stages in our method on the RHD dataset. As shown in Figure 4, we conduct a comparative analysis of the performance exhibited by each stage of our approach, utilizing the PCK metric as a benchmark. Our observations show that there is a significant improvement in performance during the first three stages, followed by a phase of diminishing returns, with no substantial improvement after the fifth stage.

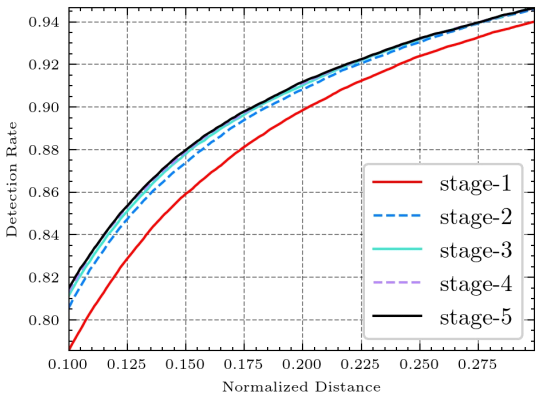
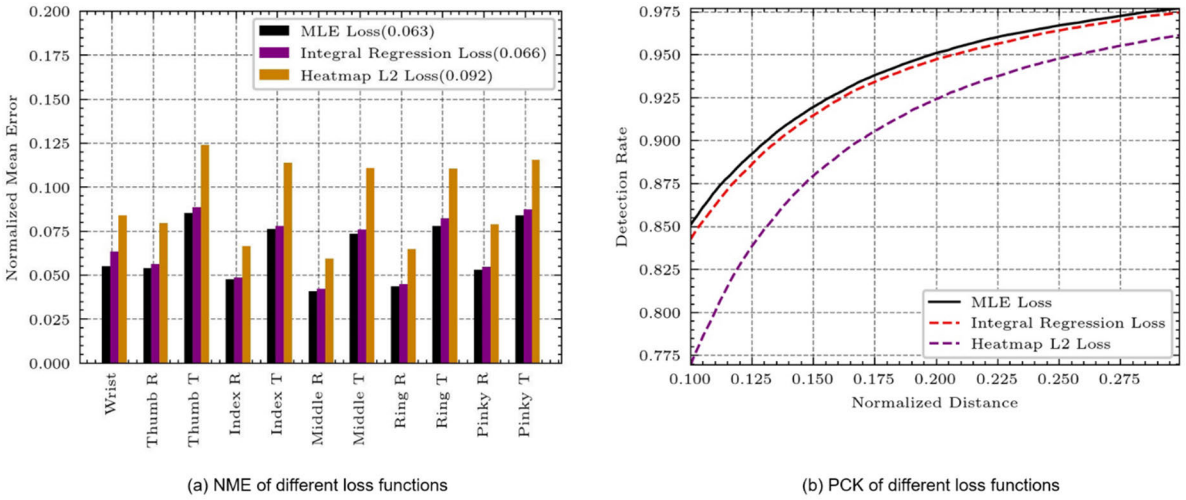


FIGURE 4. Comparison of PCK scores across various stages of our approach on the RHD dataset. The accuracy increases with each additional stage, achieving optimal perfor-mance at stage-5.

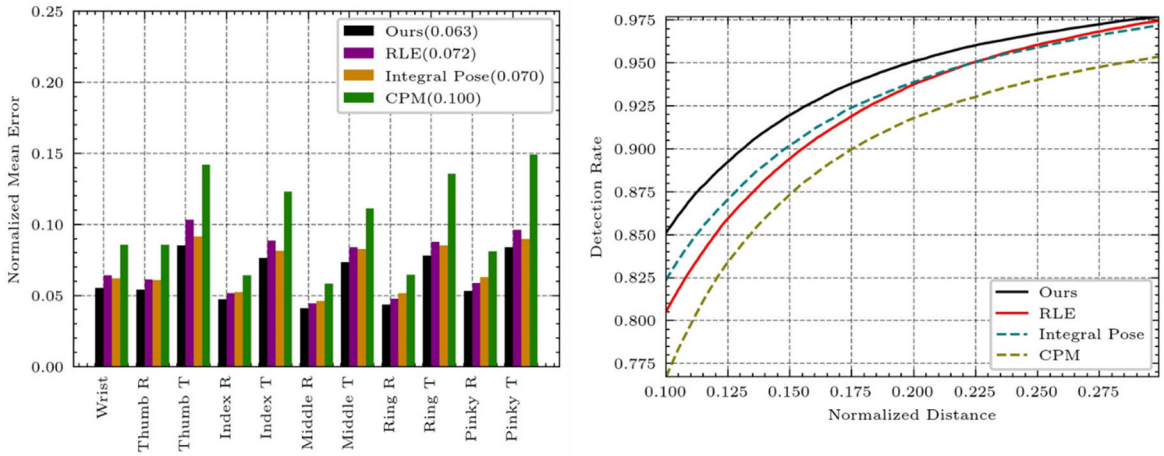
Consequently, to achieve an optimal balance between model complexity and performance efficacy, we limit the architecture to encompass five stages. This consistent improvement in model performance underscores the significance of integrating shared features and heatmaps from preceding stages. Such integration facilitates the incorporation of a broader contextual spectrum into the current stage network, thereby furnishing reliable prior information that helps produce more precise results.

Moreover, we delve into the effects of MLE-Loss on the efficacy of our approach. While retaining the core architecture of the heatmap model, we replace the loss function of our method with the widely utilized heatmap ℓ_2 loss [43], [44], [45] and Integral Regression Loss [31], conducting retraining on the Freihand dataset accordingly.



(a) NME of different loss functions (b) PCK of different loss functions

FIGURE 5. Comparison of different loss functions on the Freihand dataset. The left panel (a) illustrates the influence of different loss functions on the NME metric for estimating 11 hand joints (R: root, T: tip). The overall mean NME is presented in paren-theses. The right panel (b) demonstrates the effect of different loss functions on the percentage of good frames in terms of PCK metric.



(a) NME on Freihand for 11 hand joints (b) PCK on Freihand

FIGURE 6. Comparison of different loss functions on the Freihand dataset. The left panel (a) illustrates the influence of different loss functions on the NME metric for estimating 11 hand joints (R: root, T: tip). The overall mean NME is presented in paren-theses. The right panel (b) demonstrates the effect of different loss functions on the percentage of good frames in terms of PCK metric.

As depicted in Figure 5, MLE Loss demonstrates superior performance in estimating 11 hand joints according to the NME metric. When comparing using the $PCK_{0.2}$ metric, MLE Loss outperforms the Integral Regression Loss by 0.37% and the heatmap ℓ_2 loss by 2.7%.

C. COMPARISONS WITH MAINSTREAM METHODS

We conduct a comprehensive comparison of our proposed pipeline against three widely recognized methods, namely RLE [41], Internal Pose [31], and CPM [28]. To ensure fairness, we meticulously train each method using their recommended configurations on both the Freihand and RHD datasets, resulting in satisfactory training outcomes.

On the Freihand dataset, the performance of our method and mainstream methods in terms of NME and PCK at different thresholds is depicted in Figure 6. The performance on the RHD dataset is shown in Figure 7. Our method outperforms the aforementioned mainstream methods in both PCK and NME metrics on both the Freihand and RHD datasets.

Some qualitative results on the Freihand and RHD datasets are shown in Figure 8. Representative scenarios from the datasets are selected as examples, encompassing different viewpoints, various gestures, backgrounds with complex features that may be confused with the hand region, and challenging scenarios with occlusions either from external objects or self-occlusion of hand joints. The observed performance of our method in

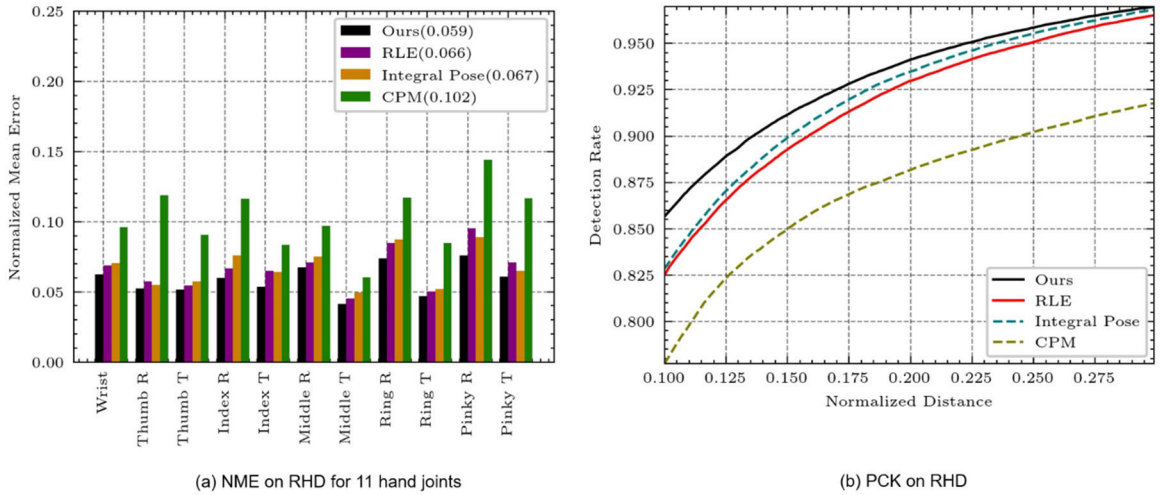


FIGURE 7. Comparison of different loss functions on the Freihand dataset. The left panel (a) illustrates the influence of different loss functions on the NME metric for estimating 11 hand joints (R: root, T: tip). The overall mean NME is presented in paren-theses. The right panel (b) demonstrates the effect of different loss functions on the percentage of good frames in terms of PCK metric.

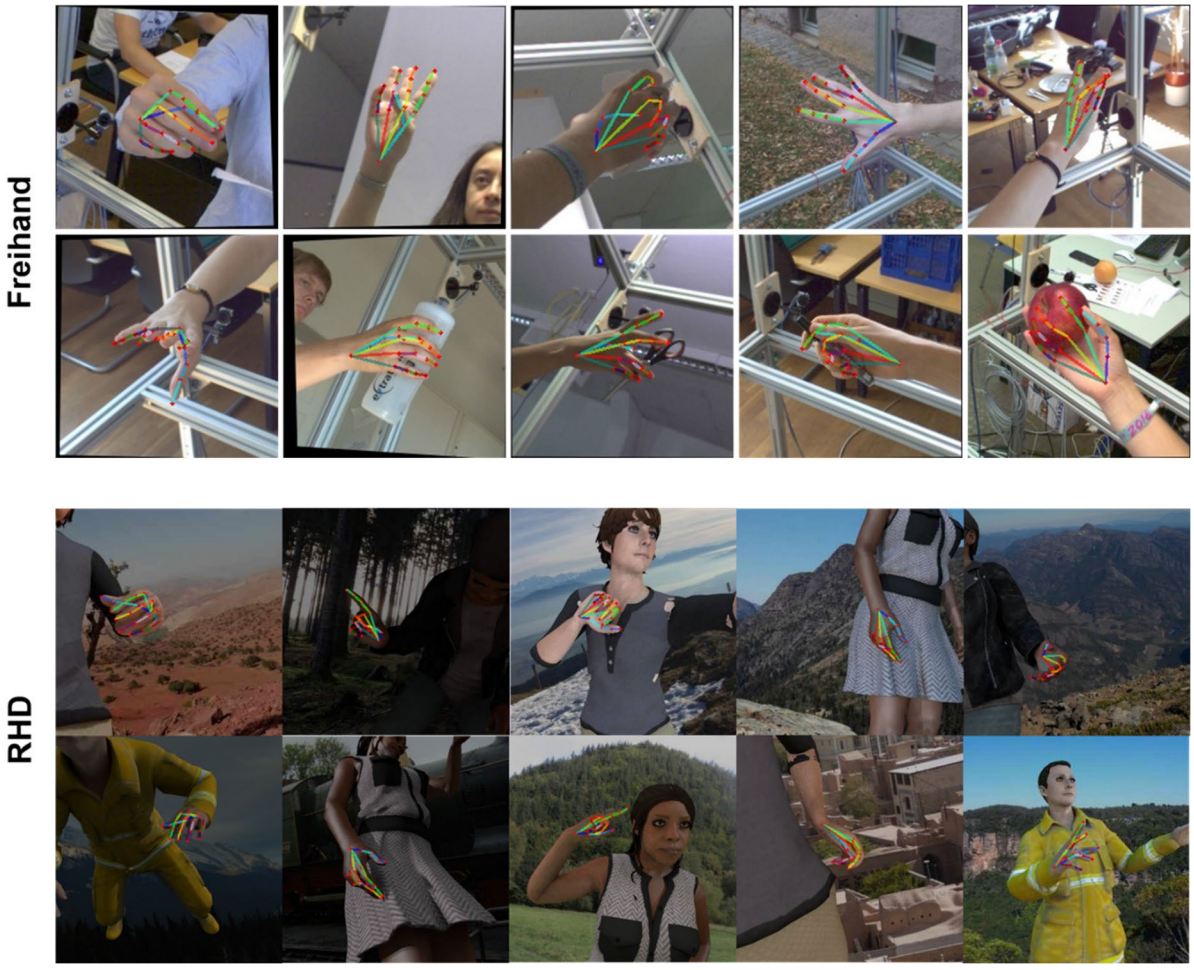


FIGURE 8. Qualitative results of our method on the Freihand and RHD datasets respectively. It is evident from the results that our approach maintains robustness and accuracy even in challenging scenarios.

handling these challenging scenarios is ideal, demonstrating good accuracy and robustness under such conditions.

V. CONCLUSION

In this study, we propose a novel sequential convolutional neural network model driven by MLE-Loss. The main

objective of this model is to characterize the distribution of deviations between the estimated results and the ground truth, aiming to improve the accuracy of predicting the ideal labels at the labeled data points. This is achieved through the calculation of MLE Loss, which facilitates the prediction of ideal coordinates while effectively mitigating the influence of noise. To evaluate the effectiveness of our approach, we conduct comprehensive evaluations by performing self-comparisons and comparisons with mainstream methods on two challenging public datasets, namely Freihand and RHD. Our experimental findings demonstrate that our proposed MLE-Loss Driven Robust Hand Pose Estimation achieves robust hand pose estimation with high precision.

Furthermore, we acknowledge that our network architecture is overall overly complex, leaving room for further simplification and improvement. In future research, we will focus on enhancing the conciseness of the network structure and continue to explore the untapped potential of the MLE-Loss. This will enable us to further enhance the accuracy and efficiency of our approach for high-precision and robust gesture recognition.

REFERENCES

[1] V. John, M. Umetsu, A. Boyali, S. Mita, M. Imanishi, N. Sanma, and S. Shibata, "Real-time hand posture and gesture-based touchless automotive user interface using deep learning," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 869–874.

[2] A. Tewari, B. Taetz, F. Grandidier, and D. Stricker, "[POSTER] a probabilistic combination of CNN and RNN estimates for hand gesture based interaction in car," in *Proc. IEEE Int. Symp. Mixed Augmented Reality (ISMAR-Adjunct)*, Oct. 2017, pp. 1–6.

[3] A. Ahmad, C. Migniot, and A. Dipanda, "Hand pose estimation and tracking in real and virtual interaction: A review," *Image Vis. Comput.*, vol. 89, pp. 35–49, Sep. 2019.

[4] K. Ahuja, V. Shen, C. M. Fang, N. Riopelle, A. Kong, and C. Harrison, "ControllerPose: Inside-out body capture with VR controller cameras," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, Apr. 2022, pp. 1–13.

[5] M.-Y. Wu, P.-W. Ting, Y.-H. Tang, E.-T. Chou, and L.-C. Fu, "Hand pose estimation in object-interaction based on deep learning for virtual reality applications," *J. Vis. Commun. Image Represent.*, vol. 70, Jul. 2020, Art. no. 102802.

[6] H. Yi, J. Hong, and H. Kim, "DexController: Designing a VR controller with grasp-recognition for enriching natural game experience," *Virtual Real*, vol. 22, pp. 1–22, Nov. 2019.

[7] N. Santavas, I. Kansizoglou, L. Bampis, E. Karakasis, and A. Gasteratos, "Attention! A lightweight 2D hand pose estimation approach," *IEEE Sensors J.*, vol. 21, no. 10, pp. 11488–11496, May 2021.

[8] I. U. Rehman, S. Ullah, and D. Khan, "FPSI-fingertip pose and state-based natural interaction techniques in virtual environments," *Multimedia Tools Appl.*, vol. 82, no. 14, pp. 20711–20740, Jun. 2023.

[9] K. Picos and U. Orozco-Rosas, "Evolutionary correlation filtering based on pseudo-bacterial genetic algorithm for pose estimation of highly occluded targets," *Multimedia Tools Appl.*, vol. 80, no. 15, pp. 23051–23072, Jun. 2021.

[10] Z. Luo, Z. Wang, Y. Huang, L. Wang, T. Tan, and E. Zhou, "Rethinking the heatmap regression for bottom-up human pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 13259–13268.

[11] I. Sárándi, T. Linder, K. O. Arras, and B. Leibe, "MeTRAbs: Metric-scale truncation-robust heatmaps for absolute 3D human pose estimation," *IEEE Trans. Biometrics, Behav., Identity Sci.*, vol. 3, no. 1, pp. 16–30, Jan. 2021.

[12] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1302–1310.

[13] M. Zhang, Z. Zhou, and M. Deng, "Cascaded hierarchical CNN for 2D hand pose estimation from a single color image," *Multimedia Tools Appl.*, vol. 81, no. 18, pp. 25745–25763, Jul. 2022.

[14] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5686–5696.

[15] C. Zimmermann, D. Ceylan, J. Yang, B. Russell, M. J. Argus, and T. Brox, "FreiHAND: A dataset for markerless capture of hand pose and shape from single RGB images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 813–822.

[16] C. Zimmermann and T. Brox, "Learning to estimate 3D hand pose from single RGB images," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4913–4921.

[17] D. Tzionas, L. Ballan, A. Srikantha, P. Aponte, M. Pollefeys, and J. Gall, "Capturing hands in action using discriminative salient points and physics simulation," *Int. J. Comput. Vis.*, vol. 118, no. 2, pp. 172–193, Jun. 2016.

[18] H. Liang, J. Yuan, and D. Thalmann, "Resolving ambiguous hand pose predictions by exploiting part correlations," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 7, pp. 1125–1139, Jul. 2015.

[19] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree, and J. Kautz, "Online detection and classification of dynamic hand gestures with recurrent 3D convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4207–4215.

[20] M. Zhang, Z. Zhou, X. Tao, N. Zhang, and M. Deng, "Hand pose estimation based on fish skeleton CNN: Application in gesture recognition," *J. Intell. Fuzzy Syst.*, vol. 44, no. 5, pp. 8029–8042, May 2023.

[21] P. Malavath and N. Devarakonda, "Estimation of 3D anatomically précised hand poses using single shot corrective CNN," *J. Intell. Fuzzy Syst.*, vol. 45, no. 5, pp. 8263–8277, Nov. 2023.

[22] X. Chen, G. Wang, H. Guo, and C. Zhang, "Pose guided structured region ensemble network for cascaded hand pose estimation," *Neurocomputing*, vol. 395, pp. 138–149, Jun. 2020.

[23] X. Zhou, Q. Wan, W. Zhang, X. Xue, and Y. Wei, "Model-based deep hand pose estimation," 2016, *arXiv:1606.06854*.

[24] C. Choi, S. Kim, and K. Ramani, "Learning hand articulations by hallucinating heat distribution," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3123–3132.

[25] Z. Zheng, Z. Hu, and H. Qin, "Stacked graph bone region U-Net with bone representation for hand pose estimation and semi-supervised training," *Image Vis. Comput.*, vol. 134, May 2023, Art. no. 104673.

[26] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, "Hand keypoint detection in single images using multiview bootstrapping," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4645–4653.

[27] L. Ge, H. Liang, J. Yuan, and D. Thalmann, "Robust 3D hand pose estimation from single depth images using multi-view CNNs," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4422–4436, Sep. 2018.

[28] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4724–4732.

[29] S. An, X. Zhang, D. Wei, H. Zhu, J. Yang, and K. A. Tsintotas, "FastHand: Fast monocular hand pose estimation on embedded systems," *J. Syst. Archit.*, vol. 122, Jan. 2022, Art. no. 102361.

[30] J. Tompson, R. Goroshin, and A. Jain, "Efficient object localization using convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Aug. 2015, pp. 648–656.

[31] X. Sun, B. Xiao, and F. Wei, "Integral human pose regression," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 529–545.

[32] D. C. Luvizon, H. Tabia, and D. Picard, "Human pose regression by combining indirect part detection and contextual information," *Comput. Graph.*, vol. 85, pp. 15–22, Dec. 2019.

[33] F. Gomez-Donoso, S. Orts-Escolano, and M. Cazorla, "Robust hand pose regression using convolutional neural networks," in *Proc. 3rd Iberian Robot. Conf.*, 2017, pp. 591–602.

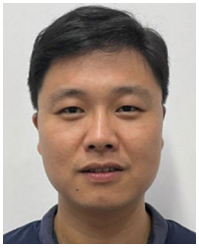
[34] A. Toshev and C. Szegedy, "DeepPose: Human pose estimation via deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1653–1660.

[35] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.-L. Chang, and M. Grundmann, "MediaPipe hands: On-device real-time hand tracking," 2020, *arXiv:2006.10214*.

[36] X. Chen, G. Wang, C. Zhang, T.-K. Kim, and X. Ji, "SHPR-net: Deep semantic hand pose regression from point clouds," *IEEE Access*, vol. 6, pp. 43425–43439, 2018.

[37] L. Ge, H. Liang, J. Yuan, and D. Thalmann, "Real-time 3D hand pose estimation with 3D convolutional neural networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 4, pp. 956–970, Apr. 2019.
[38] M.-Y. Ng, C.-B. Chng, W.-K. Koh, C.-K. Chui, and M. C.-H. Chua, "An enhanced self-attention and A2J approach for 3D hand pose estimation," *Multimedia Tools Appl.*, vol. 81, no. 29, pp. 41661–41676, Dec. 2022.
[39] B. Gao, K. Ma, H. Bi, L. Wang, and C. Wu, "Learning high resolution reservation for human pose estimation," *Multimedia Tools Appl.*, vol. 80, no. 19, pp. 29251–29265, Aug. 2021.
[40] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[41] J. Li, S. Bian, and A. Zeng, "Human pose regression with residual log-likelihood estimation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 11025–11034.
[42] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real NVP," 2016, *arXiv:1605.08803*.
[43] Y. Chen, C. Shen, X.-S. Wei, L. Liu, and J. Yang, "Adversarial PoseNet: A structure-aware convolutional network for human pose estimation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1221–1230.
[44] C.-J. Chou, J.-T. Chien, and H.-T. Chen, "Self adversarial training for human pose estimation," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA ASC)*, Nov. 2018, pp. 17–30.
[45] X. Chu, W. Yang, W. Ouyang, C. Ma, A. L. Yuille, and X. Wang, "Multi-context attention for human pose estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5669–5678.

XUDONG LOU was born in Ningbo, Zhejiang, China, in 1997. He received the master's degree in electronic information from China Jiliang University, in 2023. His current research interests include computer vision, deep learning, and visual SLAM.



XIN LIN received the B.S. degree from Changchun University of Science and Technology, in 2023. His current research interests include intelligent cockpit, robotics, and automatic control theory.

XIANGXIAN ZHU received the master's degree from Zhejiang University, in 2018. His current research interests include intelligent cockpit, robotics, and autonomous vehicles.

CHEN CHEN received the master's degree in computer science from Ningbo University. His current research interests include the Internet of Things (IoT) technology, computer vision, speech recognition, and natural language processing (NLP).