CLIP-Hand: CLIP-based Regressor for Hand Pose Estimation and Mesh Recovery

Feng Zhou[1], Shuang Ji[1], Pei Shen[1], Ju Dai ✉[2], Junjun Pan[2,3], Yu-Kun Lai[4], Paul L. Rosin[4]

[1]North China University of Technology, Beijing, 100144, China.
[2]Peng Cheng Laboratory, Shenzhen, 518000, China.
[3]Beihang University, Beijing, 100144, China.
[4]Cardiff University, Cardiff, Wales, UK.

Contributing authors: zhoufeng@ncut.edu.cn; jishuang@mail.ncut.edu.cn; shenpei0418@163.com; daij@pcl.ac.cn; pan_junjun@buaa.edu.cn; laiy4@cardiff.ac.uk; RosinPL@cardiff.ac.uk;

Abstract

Despite significant advancements in 3D hand pose estimation, it still faces challenges due to self-occlusion and complex backgrounds. To tackle those issues, we propose a CLIP-based Regressor for Hand Pose Estimation and Mesh Recovery (CLIP-Hand) from a single RGB image. Specifically, we propose an innovative method that combines high-resolution feature aggregation with contrastive language-image pre-trained model (CLIP) to enhance feature representations through language-guided visual prompts. Our approach utilizes a multi-layer Transformer encoder-decoder module to improve the prediction accuracy of hand meshing and joint points. To boost the performance, a predefined 3D joint module and a text dataset are proposed to augment the training data and improve the model's generalization ability across different scenarios. Extensive experiments on datasets such as FreiHAND, RHD, and Dexter+Object demonstrate the effectiveness of our approach, showing improved performance in terms of accuracy and robustness compared to existing methods. The source code and data will be released once the paper is accepted.

Keywords: Hand Pose, Mesh Recovery, CLIP, Heatmap, Human-computer Interaction

1 Introduction

With the rapid advancement of computer vision and machine learning, 3D hand pose estimation has emerged as a pivotal technology across several domains, including human-computer interaction (HCI) [1–3], virtual reality (VR) [4–6], augmented reality (AR) [7–9], and robotics [10–12]. The primary goal of hand pose estimation is to accurately track and interpret the 3D position and orientation of the hand, enabling systems to understand and simulate complex hand movements. This technology not only offers the potential to enhance user experiences in digital environments but also promises transformative benefits across diverse areas. For example, it can improve accessibility for individuals with disabilities, enabling more intuitive interaction with digital devices and assistive technologies. Furthermore, 3D hand pose estimation can enhance the capabilities of robotic systems, enabling more precise manipulation and interaction with the physical world. As such, this field holds vast potential with far-reaching social and technological implications.

Traditional 2D hand pose recognition methods are limited by viewing angles and lighting conditions, which make accurate recognition challenging. 3D hand pose recognition technology overcomes these limitations by using depth cameras, structured light scanning, and other devices that are able to capture hand depth information from multiple angles. Gesture pose estimation methods usually rely on depth maps or RGB images. While depth-map-based methods tend to achieve high accuracy in hand pose estimation, they are often constrained by factors such as the limited depth camera range and challenges in handling complex or dynamic environments. In contrast, RGB-based methods are more flexible and can process images in real time, but they may struggle with accuracy, particularly in occluded or low-light conditions. In recent years, breakthroughs in deep learning [13] have provided strong algorithmic support for 3D hand pose recognition, significantly improving its accuracy and robustness.

Despite its advancements, 3D hand pose estimation still faces several significant challenges. One key issue is the complexity and diversity of human hands, which necessitates large, labeled datasets for effective model training. Additionally, the dynamic nature of real-world environments demands high real-time performance, putting pressure on the algorithms to respond quickly to changes. Illumination variability and occlusion further complicate the task, as performance can degrade under different lighting conditions or when parts of the hand are hidden. Another ongoing challenge is adapting 3D hand pose estimation for practical applications and ensuring its generalization across diverse scenarios, thereby making it robust across diverse use cases and environments.

Recently, the evolution of the Transformer architecture in natural language processing has driven the introduction of the Vision Transformer (ViT) [14] into computer vision. This groundbreaking development has opened up new perspectives and approaches in computer vision research. Transformer-based models have since been applied to a wide range of computer vision tasks, including object detection, semantic segmentation, and video understanding, achieving impressive results. This trend has also been extended to tasks such as hand pose estimation and mesh reconstruction, yielding remarkable outcomes [15, 16]. One of the pioneering works, METRO [15], demonstrates the potential of Transformer-based architectures for human mesh recovery, showing significant advantages over traditional methods. However, METRO demands substantial computational resources. To address this challenge, FastMETRO [16] introduces a novel Transformer encoder-decoder architecture that mitigates bottlenecks by disentangling the interactions. While Transformer-based approaches have shown promising results, they still face challenges in effectively addressing tasks such as hand pose estimation and mesh recovery.

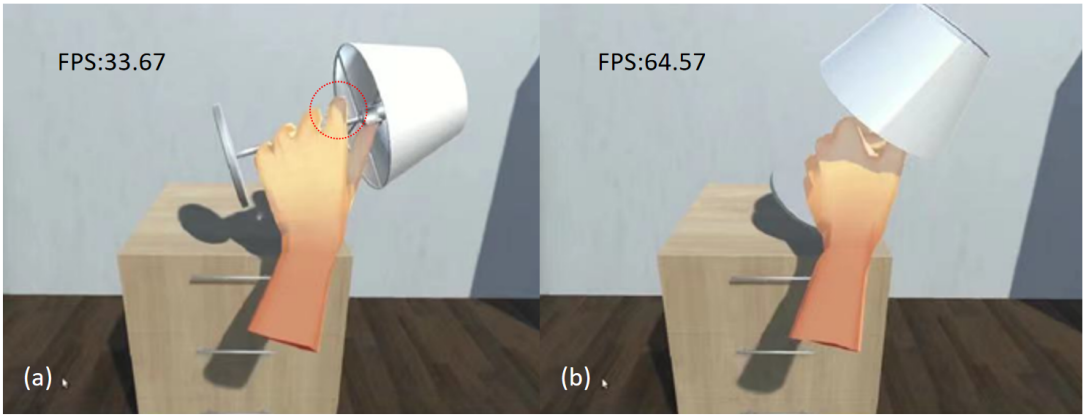

Fig. 1 A simple game interaction scenario: (a) A scene where the linear blend skinning model of the hand is driven by the predicted 3D joint positions from the video, interacting with the object at a low frame rate (FPS). (b) The same scene at a high FPS. At low FPS, inaccurate predictions of 3D joint positions for certain frames can lead to abnormal hand deformations when driving the linear blend skinning model, as demonstrated in the red-circled example, where the finger exhibits abnormal deformation. Best viewed in color.

In this paper, we propose an innovative 3D gesture pose estimation method, CLIP-Hand, that combines high-resolution feature aggregation with CLIP [17] to enhance feature representations via language-guided visual prompts. The proposed CLIP-hand consists of a High-resolution Feature Aggregation Module (HFA) for extracting high-resolution feature maps, a Multi-scale Heatmap Fusion Module (MHF) for learning vertices, joint heatmaps, and

attention map features, as well as a Multi-layer Transformer Encoder-Decoder Fusing Attention Module (MTED) for regressing 3D hand vertices and joints, and CLIP Module (CM) to obtain more plausible feature representation. While many existing hand pose estimation methods achieve competitive results without CLIP, most of them rely heavily on low-level visual cues such as texture, color, and edges. These features are sensitive to occlusion, lighting, and background variations, which often lead to performance degradation in real-world conditions. In contrast, CLIP provides semantically aligned image–text representations that capture high-level visual concepts rather than pixel-level details. By integrating CLIP into our framework, the model can focus on meaningful regions (i.e., the hand) and suppress irrelevant background information. This design enhances robustness against occlusion and background clutter, while also improving zero-shot generalization across unseen domains and subjects. To address the challenges posed by diverse poses and varying correlations between joints in 3D hand pose estimation, we propose the integration of a predefined 3D joint (PJ) module, which serves to initialize the gesture pose, providing a structured starting point that can enhance the model's performance. We evaluate the proposed CLIP-Hand on widely used datasets, i.e., FreiHAND, RHD, and Dexter+Object. Experimental results demonstrate that our CLIP-Hand achieves state-of-the-art performance. To further evaluate effectiveness and efficiency, we construct a simple game scenario, as shown in Figure 1, to demonstrate its practical application.

The main contributions of our method are listed as follows:

- We propose a novel framework, CLIP-Hand, for 3D hand pose estimation. Our method effectively leverages high-resolution features and the power of CLIP to improve performance.
- Under the framework, the HFA, MHF, MTED, CM, and PJ modules are proposed to boost the performance.
- Experiments on three challenging datasets, FreiHAND, RHD, and Dexter+Object, demonstrate that our method achieves comparable impressive results.

The rest of the paper is organized as follows. Section 2 briefly introduces related work with a highlight on recent work. Then, we give the details of the proposed model and the experimental results in section 3 and section 4, respectively. Finally, the conclusions are drawn in section 5.

2 Related Work

First, we review advanced methods for 3D hand pose estimation. Then, we introduce CLIP's applications in 3D hand pose estimation, highlighting their relevance to our work.

2.1 3D Hand Pose Estimation

In recent years, deep neural network-based methods [18–21] have become the dominant approach for 3D hand pose estimation. These methods can be broadly categorized into two types based on their input modality: depth-map-based methods [22, 23] and RGB-image-based methods [24]. While depth map-based approaches provide useful geometric information, their range is typically constrained, and they are sensitive to environmental factors. In contrast, RGB images offer high spatial resolution and are easily accessible, making them ideal for downstream applications. As a result, RGB-based methods have attracted the most attention in the field of 3D hand pose estimation due to their faster processing speeds and improved real-time performance. Recent work in this area has focused on overcoming challenges such as occlusion and on improving the accuracy and efficiency of 3D pose estimation. For instance, [25] and [26] address occlusion issues by exploiting the spatial relationships between hand keypoints or mesh vertices, improving the robustness of 3D hand pose estimation from RGB images. Additionally, methods leveraging Transformer encoders [15, 27] have been introduced to enhance the precision of regressed 3D joint and mesh vertex coordinates while also improving the efficiency of both model training and inference. Kim et al. [28] propose a feature sampling strategy to generate more realistic and natural human and hand poses, further advancing the realism of the generated 3D poses.

More recently, several works have sought to push the boundaries of 3D hand pose estimation. For interacting-hand scenarios, Jiang et al. [29] extend anchor-to-joint encoding to handle inter-hand occlusions and complex articulations. Cheng et al. [30] leverage diffusion models over image–point cloud representations, improving robustness to noise and uncertain

depth cues. In egocentric vision, Liu et al. [31] enhance depth reasoning by adapting single-view networks to dual-view settings. These advances reflect an ongoing trend toward unified architectures that integrate local geometry and global context for high-fidelity 3D hand reconstruction. Additionally, AHRNet [32] introduces a novel framework that integrates an attention mechanism with heatmap regression to effectively and efficiently predict 3D joint locations and reconstruct the hand mesh.

Several recent works have also expanded the scope of hand pose estimation through novel approaches. For example, Zhao et al. [33] investigate the synthetic-to-real domain gap in 3D hand pose estimation and propose a new data synthesis pipeline to better align synthetic data with real-world scenarios. Wang et al. [34] introduce UniHOPE, a unified method for both hand-only and hand-object pose estimation, leveraging dynamic feature fusion modules to improve accuracy in complex hand-object interactions. Potamias et al. [35] present WiLoR, an end-to-end 3D hand localization and reconstruction method that tackles challenges in multi-hand reconstruction under varying real-world conditions such as occlusion and lighting variations.

2.2 CLIP in 3D Hand Pose Estimation

The CLIP model, which bridges the gap between computer vision and natural language processing, has gained significant attention in recent years. Its first application to hand pose estimation is explored by Lee et al. [36], who utilize text descriptions to extract text features and integrate them with the image features to predict 3D hand pose. However, due to the discrete nature of the hand joint in 3D space, defining effective text prompts that capture pose-aware features poses significant challenges. To address this, Guo et al. [37] propose a novel text description method that successfully links irregular joint position labels with text prompts, ensuring consistent alignment between pose-aware features and their corresponding textual representations. Further advancements include the work of Zhang et al. [38], who proposed the CLAMP animal pose estimation network. This approach enhances the connection between text and animal images by adapting the pre-trained language model to incorporate spatial and feature perception, thereby improving animal pose estimation.

However, it is worth noting that CLIP also has certain limitations when applied to fine-grained 3D hand pose estimation. Since its feature space is learned from general natural images, it may struggle under unusual lighting, severe occlusions, or highly unconventional hand shapes that fall outside its training distribution. To mitigate these issues, our method combines CLIP-based semantic embeddings with the structural constraints of the 3D hand pose estimation network. Inspired by [32], this paper employs powerful CLIP technology to enhance feature extraction, thereby improving the performance of 3D hand pose estimation and robustness under complex real-world conditions.

3 Methodology

3.1 Overview of CLIP-Hand

The goal of this task is to estimate the 3D joint positions $P \in \mathbb{R}^{J\times 3}$ of the human hand in the camera coordinate system, given an input RGB image $I \in \mathbb{R}^{H\times W}$, where H and W represent the height and width of the input image, respectively, and $J = 21$ denotes the total number of hand joints.

Accurately estimating the 3D positions of hand joints, particularly their relative depth, is challenging due to occlusions, hand pose variations, and depth ambiguity in RGB images. Hand pose estimation can generally be approached using two main techniques: regression-based methods and detection-based methods. Regression-based methods directly predict the 3D coordinates of hand joints from the input image by extracting relevant features and mapping them to 3D space. However, these methods often struggle to handle complex hand movements and joint interdependencies. In contrast, detection-based methods use convolutional neural networks (CNNs) to generate heatmaps indicating hand joint locations, followed by post-processing to extract joint positions. While detection-based methods can achieve high precision, they may struggle to accurately estimate depth in 3D space, particularly under occlusions or extreme hand poses. In this paper, we propose a novel method, CLIP-Hand, that combines the strengths of both regression-based and detection-based techniques to achieve more accurate and robust 3D hand pose estimation. The architecture of our proposed method, as illustrated

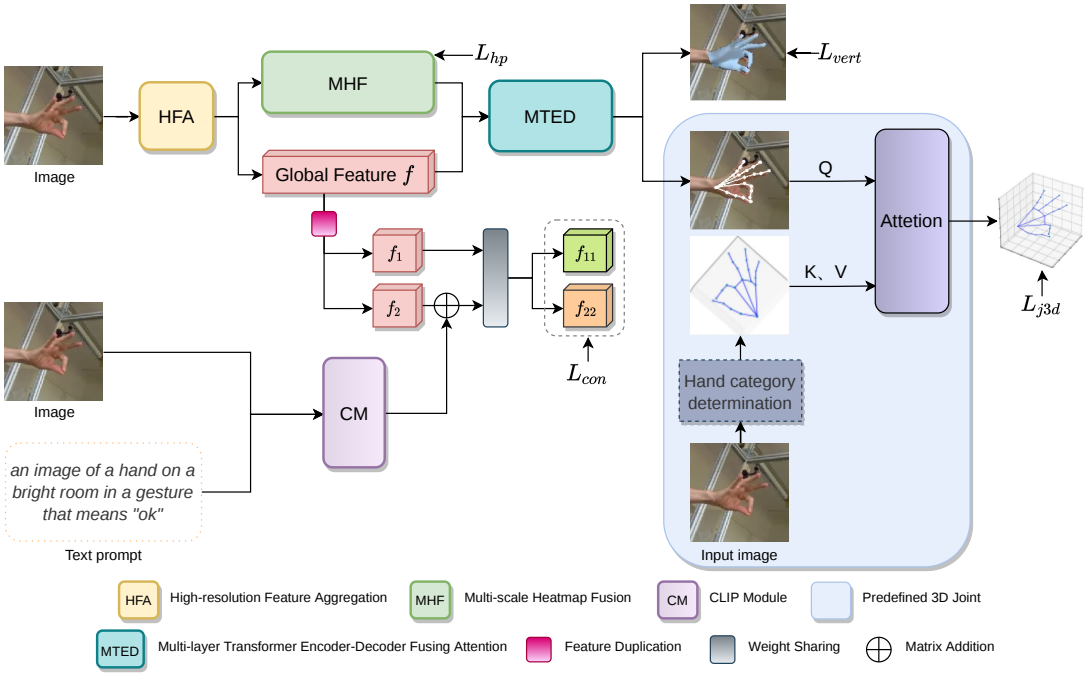

Fig. 2 Overview of the proposed CLIP-Hand, which mainly consists of (1) High-resolution Feature Aggregation (HFA), responsible for extracting high-resolution feature maps; (2) Multi-scale Heatmap Fusion (MHF), concentrating on learning vertex, joint and attention map features; (3) Multi-layer Transformer Encoder-Decoder Fusing Attention (MTED), leveraging attention to regress 3D hand vertices and joints; (4) CLIP Module (CM), used to obtain more plausible feature representations; (5) the Predefined 3D Joint (PJ) module, focusing on boosting performance. Best viewed in color.

in Figure 2, consists of four modules: HFA, MHF, MTED, and CM. Meanwhile, we adopt the PJ module to boost the performance.

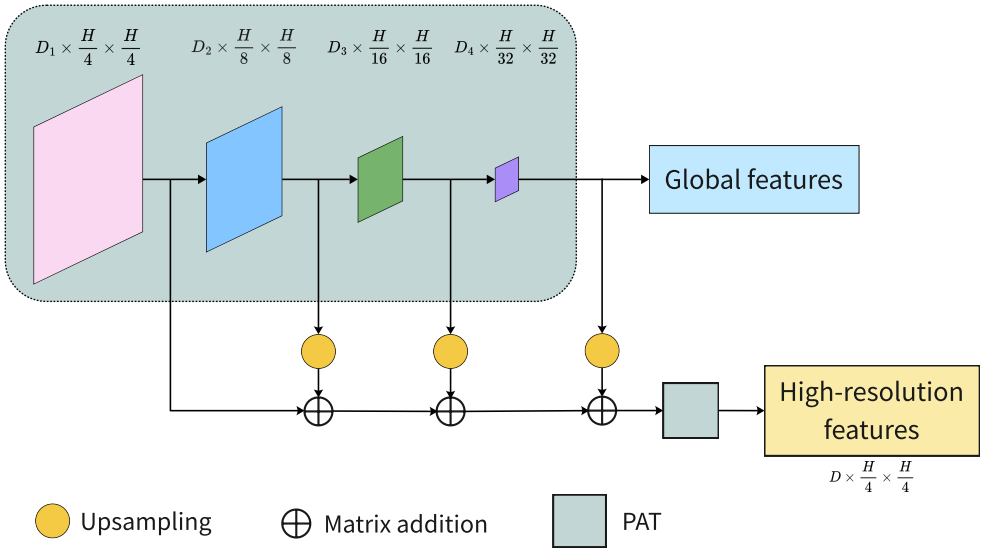

Fig. 3 The structure of HFA module, which is designed to preserve high-resolution feature maps by aggregating features from multiple layers with varying resolutions.

3.2 High-resolution Feature Aggregation (HFA) Module

Although frameworks like the Swin Transformer [39] are capable of extracting effective feature representations, the resolution of these features tends to decrease as the network depth increases. To preserve high-resolution feature maps, it becomes necessary to aggregate features from multiple layers with varying resolutions. As illustrated in Figure 3, we adopt the

approach proposed by Zheng et al. [40] to effectively combine features from different layers and obtain high-resolution feature maps.

The process of high-resolution feature aggregation begins by upsampling the image feature $f_2(D_2 \times \frac{H}{8} \times \frac{H}{8})$ from the second stage to obtain $f_2'(D_2 \times \frac{H}{4} \times \frac{H}{4})$. This upsampled feature is then concatenated with the image feature $f_1(D_1 \times \frac{H}{4} \times \frac{H}{4})$ from the first stage to produce the combined feature $f_{12}(D_1 \times \frac{H}{4} \times \frac{H}{4})$. The same procedure is repeated to generate the feature $f_{123}(D_1 \times \frac{H}{4} \times \frac{H}{4})$. Next, the feature f_{123} is concatenated with $f_4'(D_1 \times \frac{H}{4} \times \frac{H}{4})$ and passed through a PAT block [40] for dimensionality reduction and compression of the two-dimensional image features. Finally, the high-resolution feature $f_{1234}(D \times \frac{H}{4} \times \frac{H}{4})$ is obtained, where D_i denotes the channel of the image feature.

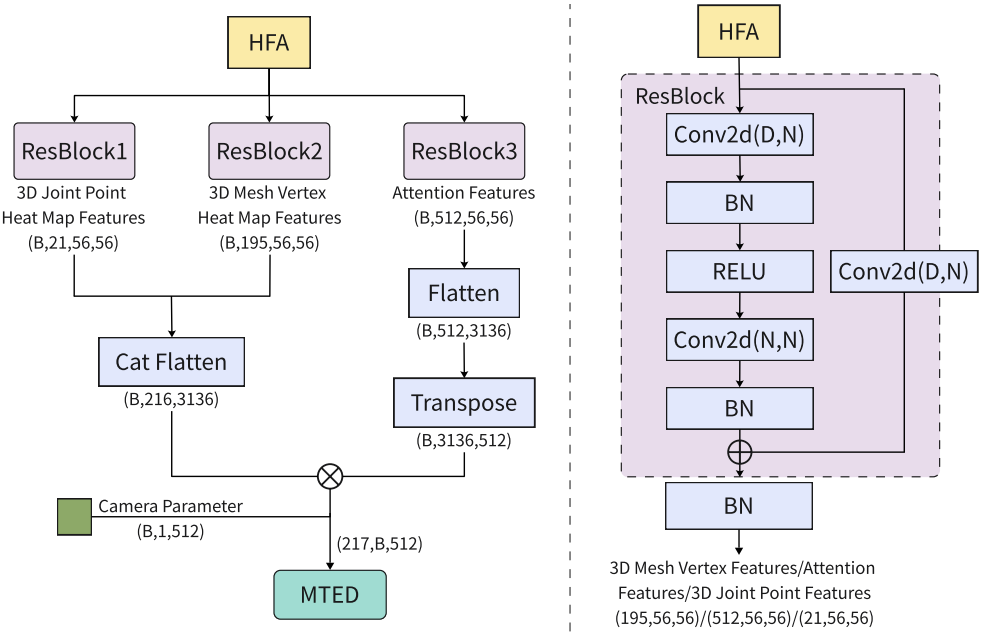

Fig. 4 The structure of MHF module, which integrates 3D joint point, mesh vertex heatmap and attention features via ResBlocks.

3.3 Multi-scale Heatmap Fusion (MHF) Module

Through the Hand Feature Aggregation (HFA) module, we obtain high-resolution features f_{1234}, which are subsequently passed to the vertex heatmap branch, joint heatmap branch, and attention feature branch. The structures of these branches are illustrated in Figure 4. To process the high-resolution features, we employ the ResBlock module [41]. Initially, the high-resolution feature $f_{1234} \in D \times \frac{H}{4} \times \frac{H}{4}$ (with $D = 64$) is input into the ResBlock module. Through the ResBlock network, we obtain the vertex heatmap features ($B \times N_1 \times \frac{H}{4} \times \frac{H}{4}$ with $N_1 = 195$), joint heatmap features ($B \times N_2 \times \frac{H}{4} \times \frac{H}{4}$ with $N_2 = 21$), and attention map features ($B \times N_3 \times \frac{H}{4} \times \frac{H}{4}$ with $N_3 = 512$).

Subsequently, the vertex heatmap and joint heatmap features are passed through a Softmax layer and matrix-multiplied with the transposed attention map features. This process yields the grid vertex heatmap features $(195, B, 512)$ and joint features $(21, B, 512)$ with attention, which serve as the grid vertex queries and joint queries for the subsequent MTED module. Simultaneously, a feature with dimensions $(1, B, 512)$ is obtained by fully connecting the global features, which is used as the initial weak perspective camera parameters (Cam Queries). Finally, these features are concatenated to form a feature vector with dimensions $(217, B, 512)$, which serves as the query input for the subsequent MTED module.

3.4 Multi-layer Transformer Encoder-Decoder (MTED) Module

Inspired by the progressive dimension reduction scheme in [15] and the novel transformer encoder-decoder architecture in [16], we propose a progressive dimension reduction transformer encoder-decoder architecture for regressing the final 3D hand vertices and joints. In our design,

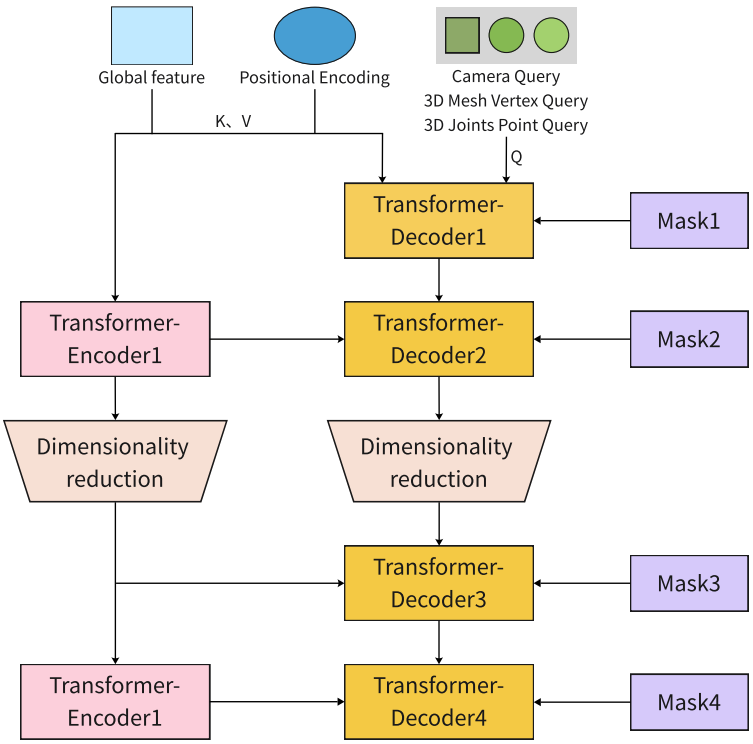

Fig. 5 Structure of the MTED module: This module integrates global features and positional encoding to inform the decoding process. It optimizes pose estimation accuracy through cross-attention mechanisms and progressive masking.

the global features serve as the Key and Value in the decoder, while the features f_s are used as queries. By incorporating the cross-attention module into the decoder, we can effectively capture the non-local relationships between hand joints and mesh vertices. As a result, we obtain the joint features $X_j \in \mathbb{R}^{N_2 \times 2D}$ and vertex features $X_v \in \mathbb{R}^{N_1 \times 2D}$.

Additionally, we utilize the progressive attention masking scheme proposed in [28], which places greater emphasis on vertices and joints within specific distance thresholds during learning. As shown in Figure 5, four masks are used in our approach, with distance thresholds set to 7, 5, 3, and 1, respectively. These thresholds define the local connections between vertices, progressively reducing the range to facilitate the model's consideration of local relationships between adjacent vertices.

3.5 CLIP Module (CM)

While the three modules discussed above demonstrate strong performance, the complexity and self-occlusion inherent in hand poses can sometimes hinder the extraction of sufficient information from images alone. To enhance the model's performance, we incorporate a CLIP module, which improves pose estimation accuracy by leveraging textual descriptions. Specifically, the RGB image and the corresponding text prompt (please refer to 4.1) are input into the image encoder and text encoder, respectively, to extract the hand image features f_{img} and the corresponding text features f_{text}. These features are then concatenated to form the fusion feature f_{clip}.

To effectively integrate the fusion feature f_{clip} from the CLIP module into the hand pose estimation network, we adopt a contrastive learning approach. Specifically, we duplicate the global features into two copies: f_1 and f_2. The fusion feature f_{clip} is added to the global feature f_2, and the resulting global features f_{11} and f_{22} are obtained by sharing the weights of f_1 and f_2 after adding the fusion feature, as shown in Figure 2. For the positive and negative samples, we adopt [42] to generate them. A contrastive learning loss is then applied to supervise the global feature representation in the network, thereby improving the model's ability to extract more accurate image features.

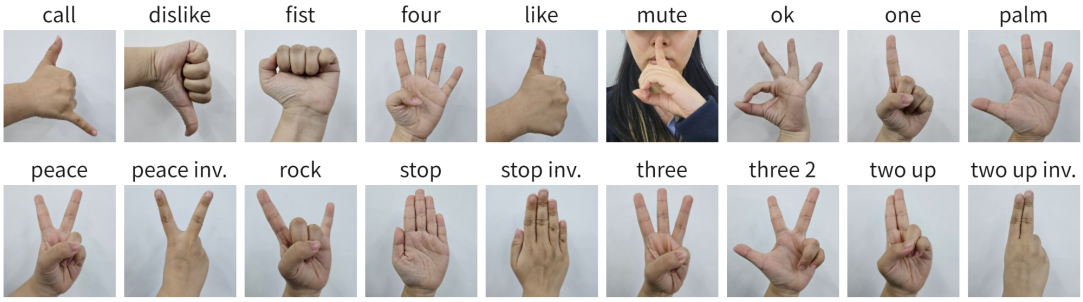

Fig. 6 Illustration of the predefined hand poses.

3.6 Predefined 3D Joint (PJ) module

To enhance model performance, we introduce a Predefined 3D Joint (PJ) module that integrates 19 predefined hand gestures [43], as shown in Figure 6. These gestures, derived from the Light-HaGRID dataset containing about 120,000 RGB hand images across 18 gesture types (e.g., *fist, OK, like, peace, palm, call, stop*) plus an additional "no-gesture" class, represent common daily actions. Although they cannot fully cover the continuous hand-pose space, they serve as representative discrete samples providing semantic and geometric diversity, defining meaningful directions in the high-dimensional pose space, and enabling the network to interpolate between poses.

We trained a YOLOv5 network on the Light-HaGRID dataset to obtain accurate hand detection boxes and gesture category predictions. Each image is annotated with gesture class and bounding box information in JSON format, with about 7,000 samples per class. After extending the gesture categories to 19, the trained YOLOv5 model achieved a final mAP@0.5 of 96.4% on the validation set and was then used to classify hand gestures in the FreiHAND dataset, providing reliable categorical cues for subsequent feature alignment.

In the PJ module, the predefined 3D hand joint templates (each of size 21×3) correspond to the 19 gesture categories. The same rotation and scaling transformations applied during data augmentation are also applied to the predefined templates to ensure geometric consistency. The augmented joint templates $X_{\text{de\_j\_3d}}$ and predicted joint points $X_{\text{pred\_j\_3d}}$ are fed into an attention module, where the templates serve as Key and Value vectors and the predicted joints act as the Query. Through this attention-based mechanism, the module learns adaptive soft weights between the predicted and reference joints, dynamically fusing multiple templates rather than memorizing a single one. This design enables efficient spatial correlation modeling and stabilizes optimization. Additionally, the contrastive learning objective encourages discrimination between structurally similar but semantically distinct gestures, preventing feature collapse and improving robustness.

The PJ module thus combines semantic coherence and geometric validity, demonstrating strong generalization across unseen subjects, viewpoints, and lighting conditions. The YOLOv5-based classification ensures semantic consistency (e.g., similar gestures such as *fist* and *half-fist* share close feature representations), while the 19 templates provide a standardized geometric scaffold that maintains structural plausibility under occlusion or rotation. Experiments on RHD and Dexter+Object verify that the module maintains high accuracy and robustness under challenging conditions, including occlusion, background clutter, and illumination variation.

3.7 Loss function

The proposed 3D hand pose estimation method based on attention and different scale heatmaps is trained on five loss functions, namely, the hand 3D mesh vertex loss L_{vert}, the hand 3D joint point loss L_{j3d}, the hand 2D joint point loss L_{j2d}, the heatmap loss L_{hp}, and the contrastive learning loss L_{hp}. The L_{vert} employs the L_1 loss function to quantify the discrepancy between the predicted vertices and the ground truth vertices. The L_{j3d} utilizes the mean squared error (MSE) loss function to measure the difference between the predicted joints and the ground truth joints, as well as the disparity between the 3D joints regressed from the predicted vertices and the ground truth joints. The L_{j2d} applies the L_1 loss function to compute the difference

between the projected 2D joint coordinates, obtained by applying camera intrinsic parameters to the predicted vertices, and the corresponding ground truth 2D joint coordinates. The L_{hp} measures the difference between the predicted heatmap and the ground truth heatmap. Furthermore, contrastive loss L_{con} NT-XentLoss [44] is employed to enhance the learning of generalizable features, thereby improving the robustness of hand pose estimation.

The final loss of the proposed 3D hand pose estimation network based on different-scale heatmaps is given by Equation 1.

$$L_{total} = W_{vert} \times L_{vert} + W_{j3d} \times L_{j3d} + W_{j2d} \times L_{j2d} + L_{hp} + W_{con} \times L_{con} \tag{1}$$

where W_{vert}, W_{j3d}, W_{j2d}, W_{con} are the balance factors of each loss term, which are set to 1000, 1000, 100, and 10, respectively.

The weighting strategy is empirically tuned to balance gradients across heterogeneous objectives. Since the contrastive loss (10^1–10^2) is orders of magnitude larger than geometric losses (below 10^{-4}), higher coefficients are assigned to geometric terms to prevent semantic dominance and preserve spatial accuracy. This configuration ensures stable convergence and robust performance on FreiHAND, RHD, and Dexter+Object.

4 Experiments

4.1 Datasets

The FreiHAND. The FreiHAND dataset [45] is a real-world 3D hand dataset with images with a green screen background, which facilitates changing the background for network training. The dataset contains 130,240 training images and 3,960 test images from the multi-view setup, providing an image resolution of 224×224. They provide 21 joint position annotation information, hand mask, and camera parameter matrix.

RHD dataset. The RHD dataset [13] is a synthetic hand image dataset, which contains 2020 character models and 39 actions with a total of 43,986 frames (41,258 for training and 2,728 for evaluation). The dataset has an image resolution of 320×320 and provides depth segmentation masks for 21 hand joints and location annotation information for both 2D and 3D hand joints. The training and test sets of this dataset do not have the same features or actions and cover more rare poses. Each RGB image of this dataset has a corresponding pose label and mask label.

Dexter+Object dataset. The Dexter+Object dataset [46] is designed to facilitate the evaluation of hand pose estimation and hand-object interaction analysis, and consists of six sequences recorded from two subjects, encompassing a total of 3,014 frames with a resolution of 640 × 320 pixels. These frames capture complex hand-object interactions involving cuboid objects, providing a unique and valuable resource for investigating the dynamics of hand movements during physical interactions. Given its emphasis on interactive scenarios, the dataset is particularly critical for assessing the performance of hand pose estimation algorithms in real-world, dynamic contexts.

Text dataset. In this paper, we introduce the phrase "An image/picture of hand with color background" to describe the various backgrounds associated with hand RGB images to form text-image pairs. This phrase serves as a template for generating text prompts that capture the diverse backgrounds in our dataset used for training and testing. The combination of text prompts for different backgrounds is presented in Table 1, which illustrates how different color backgrounds are associated with hand images. This approach provides a structured representation of hand images across diverse environments, facilitating evaluation of the model's robustness under varying background conditions.

To further enrich semantic information, we incorporate gesture category predictions and construct the final prompt format as "an image of a hand on a {color} background in a gesture that means {class}."Here, class corresponds to one of the 19 predefined gesture labels (e.g., "peace," "fist," "like," "OK," "palm"), and color denotes the sampled background type. This text–image pairing strategy allows the CLIP encoder to learn both visual and semantic correspondences, thereby improving robustness to variations in gesture type and background complexity. Representative examples of the constructed prompts are provided in Table 2 for clarity and reproducibility.

Table 1 Background text generation for hand RGB images of the FreiHAND dataset.

Image	Hand	Color		Background
a cropped image of an image of a cropped photo of a picture of one a photo of a photo of right	hand with right hand with	mustard bright green white olive orange blue gray pink coral	lime dark purple silver black red yellow beige brown wine	room background

Table 2 Image–text paired samples constructed from the FreiHAND dataset.

Image	Text
	an image of a hand on a bright room in a gesture that means “peace inv.”
	an image of a hand on a mountain background in a gesture that means “one”
	an image of a hand on a green background in a gesture that means “palm”
	an image of a hand on a mountain background in a gesture that means “two up inv.”

4.2 Implementation details

Parameter settings. All the experiments of our CLIP-Hand are trained using the AdamW optimizer [47] with a batch size of 8 on Geforce 4090 GPUs by PyTorch. The initial learning rate is set to 10^{-4}. We train our model for 120 epochs and lower the learning rate by a factor of 10 after each 40 epochs. Even with random seeds set, we observe that the model's performance remains erratic. Therefore, we present the average performance across three trials to overcome fluctuations in this paper.

Model size and running time. We record the model parameters saved by PyTorch for METRO [15], Fast METRO [16] and our CLIP-Hand. The size of METRO and Fast METRO are 230.4M and 153.0M, respectively, and our model is 76.3M. Hence, the proposed modules do not yield substantial parameter increases over the compared models. For training time, METRO and Fast METRO take around 1.05 s and 0.90 s per iteration, and our CLIP-Hand is 0.83 s, while for inference time, METRO and Fast METRO take around 51 ms and 46 ms per iteration under batch size 1, and our CLIP-Hand is about 41 ms.

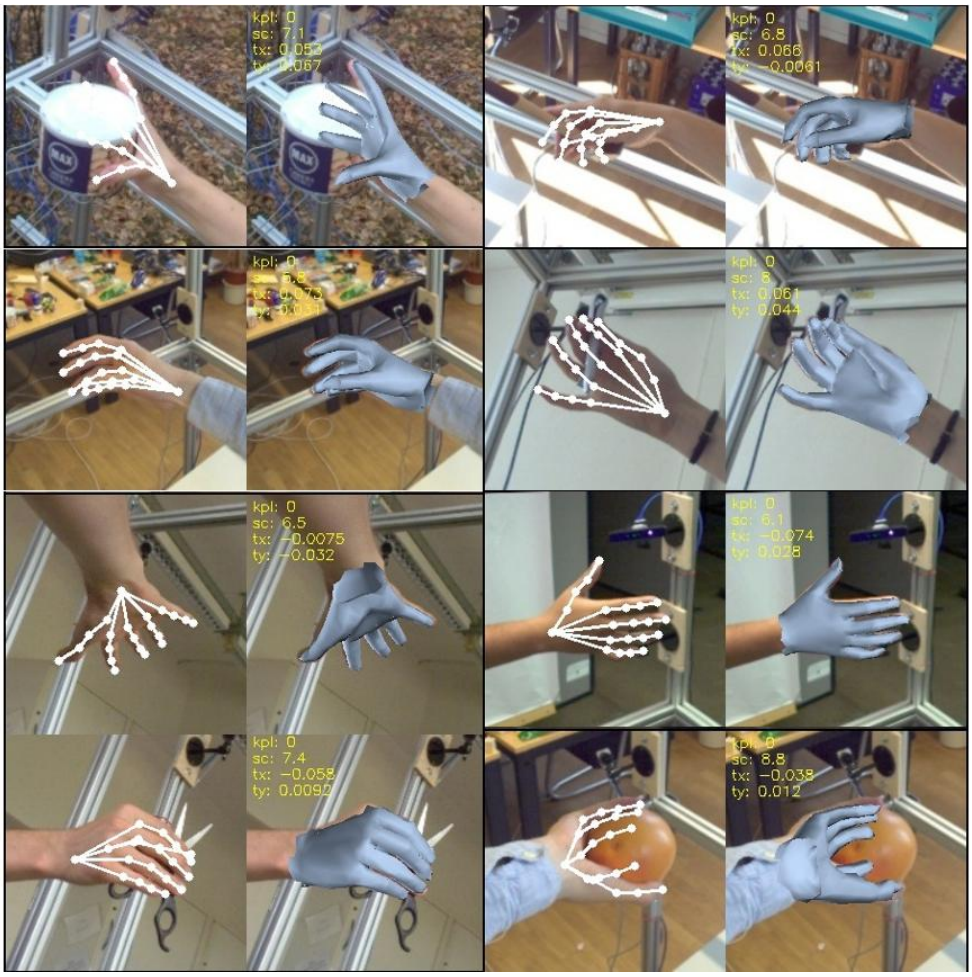

Fig. 7 Visualizations of our predicted 3D pose and mesh results on the FreiHand dataset.

4.3 Comparison with state-of-the-art methods

Experiment on FreiHAND dataset. To demonstrate the effectiveness of our model, we make comparisons with state-of-the-art methods and report the results on the FreiHand dataset in terms of FPS, Params, FLOPs, PA-MPJPE, and F@15, as shown in Table 3. Similar to [40], we increase the number of blocks to boost the performance.

Table 3 Quantitative comparisons with state-of-the-arts on the FreiHand dataset.

Methods	FPS	Params	FLOPs	PA-MPJPE	F@15mm
Hasson [48]	–	–	–	–	0.908
MANO CNN [45]	–	–	–	–	0.935
Pose2Mesh [49]	8	–	–	7.7	0.969
I2LMeshNet [50]	–	–	–	7.4	0.973
METRO [15]	19.55	183.80M	41.47G	6.8	0.981
FastMETRO [16]	21.88	133.90M	30.56G	6.5	0.982
FastViT [51]	84	–	–	6.6	0.981
Our CLIP-Hand-cls-S12	42.36	30.64M	34.01G	7.0	0.979
Our CLIP-Hand-cls-S24	30.20	40.20M	34.89G	6.3	0.984

In the comparison results, the last two rows correspond to the large and small models of our method, with the PAT module having layer sizes (2, 2, 6, 2) for the small model and (4, 4, 12, 4) for the large model. Our CLIP-Hand-cls-S24 model, the large version, achieves a PA-MPJPE of 6.3. Compared to other methods, CLIP-Hand achieves equivalent performance while delivering significantly higher frame rate (FPS). Furthermore, compared with METRO and FastMETRO, CLIP-Hand has fewer parameters, shorter training iterations, and faster inference per iteration, making it both more efficient and effective.

Figure 7 visualizes the predicted results of the CLIP-Hand model on the FreiHAND dataset. It demonstrates that the proposed method achieves convincing accuracy in predicting the 3D joint points of the hand and the 3D mesh vertices.

Experiment on the RHD and Dexter+Object datasets. To verify the accuracy of the proposed CLIP-Hand, we also evaluate our method on the RHD and Dexter+Object datasets, and make comparisons with previous methods. The 3DPCK evaluation index within the threshold of 0-0.5cm, namely PCK@0.5, is leveraged to conduct quantitative evaluations, and the results are shown in Table 4. It can be seen from Table 4 that the proposed CLIP-Hand method outperforms other methods on RHD and Dexter+Object datasets.

Table 4 Quantitative comparisons with state-of-the-arts on RHD and Dexter+Object.

Methods	RHD AUC of PCK@0.5↑	Dextere+Object AUC of PCK@0.5↑
Baek et al.[52]	0.926	0.650
Zhang et al. [53]	0.901	0.825
Rong et al.[54]	0.934	-
Cui-GCN [55]	0.933	0.77
Gu et al.[56]	0.936	-
Hao et. al. [57]	0.92	-
Our CLIP-Hand	0.941	0.938

4.4 Ablation studies

We conduct ablation studies on the FreiHand dataset using the large model (CLIP-Hand-cls-s24, termed as s24), as presented in Table 5. The results of the ablation study provide insights into the incremental contributions of each module to the performance of the CLIP-Hand model. The evaluation metrics include PA-MPJPE, which measures the average distance between predicted and actual hand joint positions, and F@15mm, which represents the proportion of correctly estimated poses in terms of the joint errors being within a defined margin.

Starting with the baseline model (s24), we observe a PA-MPJPE of 7.24 mm and an F@15mm of 0.978. As modules are sequentially added, both metrics show a general trend of improvement. Specifically, *a.* adding the MTED module yields a slight improvement, with the PA-MPJPE reducing to 7.06 mm and F@15mm increasing to 0.979; *b.* incorporating the HFA module further enhances performance, resulting in a PA-MPJPE of 7.13 mm and F@15mm of 0.979; *e.* the inclusion of the MTED module alongside the HFA and MHF combination leads to continued improvement, with a PA-MPJPE of 6.74 mm and F@15mm of 0.981; and *g.* the full combination of all modules achieves the best performance, with the lowest PA-MPJPE of 6.32 mm and the highest F@15mm of 0.984.

Table 5 Ablation experiments on FreiHand dataset for different modules of CLIP-hand.

	Methods	PA-MPJPE	F@15mm
a.	Baseline(s24)	7.24	0.978
b.	Baseline(s24)+MTED	7.06	0.979
c.	Baseline(s24)+HFA	7.13	0.979
d.	Baseline(s24)+HFA+MHF	6.86	0.980
e.	Baseline(s24)+HFA+MHF+MTED	6.74	0.981
f.	Baseline(s24)+HFA+MHF+MTED+CM	6.58	0.982
g.	Baseline(s24)+HFA+MHF+MTED+CM+PJ	6.32	0.984

To provide a deeper understanding of the proposed three modules, we present qualitative ablation studies with and without these modules on the FreiHand dataset in Figure 8. From left to right, (a) gives the result of row *b* in Table 5, (b) is the result of row *e* in Table 5, (c) is the result of row *f* in Table 5, and (d) is our CLIP-Hand.

To further verify the effectiveness of our method, we conduct additional experiments and analyses on scenes with complex backgrounds and varying illumination conditions to better understand its applicability. For the detailed results, please refer to Figure 9.

Fig. 8 Visualization of the CLIP-Hand model: From left to right, (a) presents the result of row *b* in Table 5, (b) shows the result of row *e* in Table 5, (c) depicts the result of row *f* in Table 5, and (d) is the result of our CLIP-Hand model. We highlight the contributions of various modules and showcase the model's performance across different configurations.

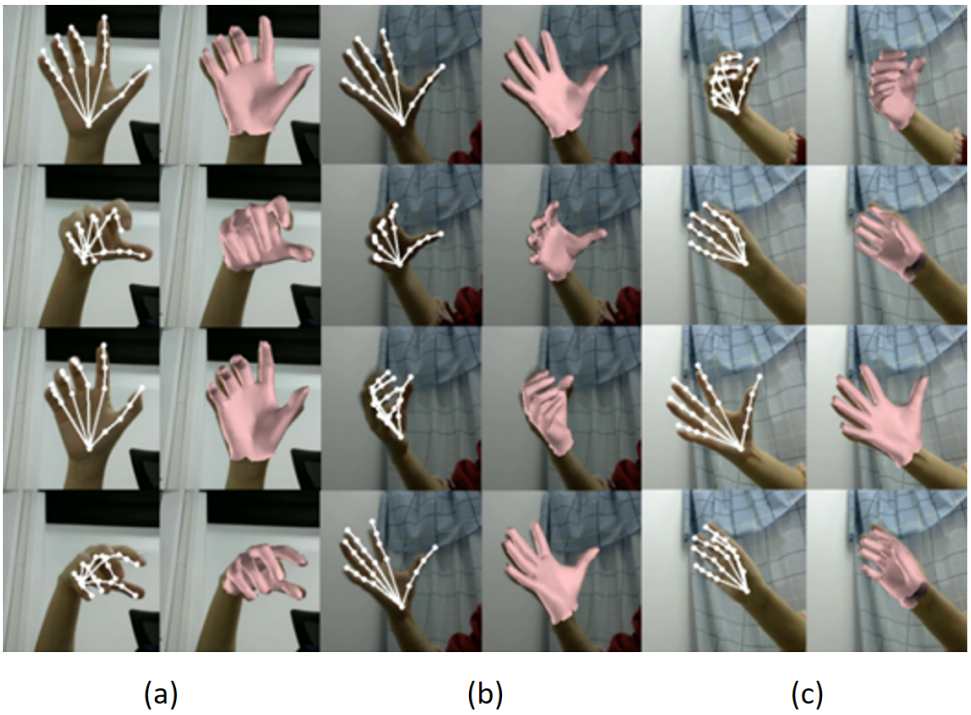

Fig. 9 Visualization of RGB hand images in a lab environment. (a) shows RGB hand images captured in a lab environment; (b) presents hand images collected after altering the background and reducing the lighting; (c) depicts hand images collected with enhanced lighting. The results demonstrate that our method maintains strong performance even on hand images that are out of distribution in the training set.

5 Conclusion

In this study, we have presented a comprehensive approach to the challenging task of 3D hand pose estimation and mesh recovery from single RGB images. Our proposed method, CLIP-Hand, leverages the strengths of high-resolution feature aggregation and contrastive language-image pre-trained models (CLIP) to enhance feature representations, thereby improving the prediction accuracy of hand meshing and joint points. By integrating a multi-layer Transformer encoder-decoder module and a predefined 3D joint module, our method optimizes the accuracy and generalization of 3D gesture pose estimation. Extensive experiments conducted on standard benchmarks have demonstrated the effectiveness of our approach. The

results indicate that our method achieves comparable accuracy and robustness to state-of-the-art techniques while exhibiting improved efficiency. This is particularly evident in the higher frame rate our method delivers, which is crucial for real-time applications in interactive virtual environments and human-computer interaction. Future work will focus on further enhancing the model's generalization capabilities and exploring additional applications in augmented reality, virtual reality, and robotics. While the current CLIP-Hand has demonstrated promising results, there is room for further enhancement, particularly in incorporating temporal information. In future work, we aim to integrate temporal features into our model to capitalize on the sequential nature of hand movements.

References

[1] Rehg, J.M., Kanade, T.: DigitEyes: Vision-based hand tracking for human-computer interaction. In: IEEE Workshop on MNAO, pp. 16–22 (1994)

[2] Farjadi, S.A., Akbarzadeh-T, M.-R., Ghiasi-Shirazi, K.: Rgb image-based hand pose estimation: A survey on deep learning perspective. In: International Symposium on Artificial Intelligence and Signal Processing, pp. 1–7 (2024)

[3] Kolotouros, N., Pavlakos, G., Daniilidis, K.: Convolutional mesh regression for single-image human shape reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4501–4510 (2019)

[4] Han, S., Liu, B., Cabezas, R., Twigg, C.D., Zhang, P., Petkau, J., Yu, T.-H., Tai, C.-J., Akbay, M., Wang, Z., et al.: MEgATrack: monochrome egocentric articulated hand-tracking for virtual reality. ACM Transactions on Graphics, 87–1 (2020)

[5] Liebers, J., Brockel, S., Gruenefeld, U., Schneegass, S.: Identifying users by their hand tracking data in augmented and virtual reality. International Journal of Human–Computer Interaction, 409–424 (2024)

[6] Gupta, S., Bagga, S., Sharma, D.K.: Hand gesture recognition for human computer interaction and its applications in virtual reality. Advanced Computational Intelligence Techniques for Virtual Reality in Healthcare, 85–105 (2020)

[7] Zenner, A., Krüger, A.: Estimating detection thresholds for desktop-scale hand redirection in virtual reality. In: IEEE VR, pp. 47–55 (2019)

[8] Malik, S., McDonald, C., Roth, G.: Hand tracking for interactive pattern-based augmented reality. In: Proceedings. International Symposium on Mixed and Augmented Reality, pp. 117–126 (2002)

[9] Hassan, E.K., Jamila Harbi, S.: 3d hand pose and shape estimation from single rgb image for augmented reality. Journal of Intelligent Systems and Internet of Things, 90–101 (2023)

[10] Handa, A., Van Wyk, K., Yang, W., Liang, J., Chao, Y.-W., Wan, Q., Birchfield, S., Ratliff, N., Fox, D.: DexPilot: Vision-based teleoperation of dexterous robotic hand-arm system. In: International Conference on Robotics and Automation, pp. 9164–9170 (2020)

[11] Rho, E., Lee, H., Lee, Y., Lee, K.-D., Mun, J., Kim, M., Kim, D., Park, H.-S., Jo, S.: Multiple hand posture rehabilitation system using vision-based intention detection and soft-robotic glove. IEEE Transactions on Industrial Informatics, 6499–6509 (2024)

[12] Gomez-Donoso, F., Orts-Escolano, S., Cazorla, M.: Accurate and efficient 3d hand pose regression for robot hand teleoperation using a monocular rgb camera. Expert Systems with Applications, 327–337 (2019)

[13] Zimmermann, C., Brox, T.: Learning to estimate 3d hand pose from single rgb images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4903–4911 (2017)

[14] Dosovitskiy, A., Beyer, L., Kolesnikov, e. Alexander: An image is worth 16×16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2020)

[15] Lin, K., Wang, L., Liu, Z.: End-to-end human pose and mesh reconstruction with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1954–1963 (2021)

[16] Cho, J., Youwang, K., Oh, T.-H.: Cross-attention of disentangled modalities for 3d human mesh recovery with transformers. In: Proceedings of the European Conference on Computer Vision, pp. 342–359 (2022)

[17] Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., *et al.*: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning, pp. 8748–8763 (2021)

[18] Yuan, S., Garcia-Hernando, G., Stenger, B., Moon, G., Chang, J.Y., Lee, K.M., Molchanov, P., Kautz, J., Honari, S., Ge, L., *et al.*: Depth-based 3d hand pose estimation: From current achievements to future goals. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2636–2645 (2018)

[19] Ren, P., Sun, H., Hao, J., Wang, J., Qi, Q., Liao, J.: Mining multi-view information: a strong self-supervised framework for depth-based 3d hand pose and mesh estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 20555–20565 (2022)

[20] Sheng, B., Li, P., Ali, R., Chen, C.P.: Improving video temporal consistency via broad learning system. IEEE Transactions on Cybernetics, 6662–6675 (2021)

[21] Ali, S.G., Wang, X., Li, P., Li, H., Yang, P., Jung, Y., Qin, J., Kim, J., Sheng, B.: Egdnet: an efficient glomerular detection network for multiple anomalous pathological feature in glomerulonephritis. The Visual Computer, 1–18 (2024)

[22] Moon, G., Chang, J.Y., Lee, K.M.: V2v-posenet: Voxel-to-voxel prediction network for accurate 3d hand and human pose estimation from a single depth map. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5079–5088 (2018)

[23] Cheng, J., Wan, Y., Zuo, D., Ma, C., Gu, J., Tan, P., Wang, H., Deng, X., Zhang, Y.: Efficient virtual view selection for 3d hand pose estimation. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 419–426 (2022)

[24] Iqbal, U., Molchanov, P., Gall, T.B.J., Kautz, J.: Hand pose estimation via latent 2.5 d heatmap regression. In: Proceedings of the European Conference on Computer Vision, pp. 118–134 (2018)

[25] Ge, L., Ren, Z., Li, Y., Xue, Z., Wang, Y., Cai, J., Yuan, J.: 3d hand shape and pose estimation from a single rgb image. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10833–10842 (2019)

[26] Guo, S., Rigall, E., Qi, L., Dong, X., Li, H., Dong, J.: Graph-based cnns with self-supervised module for 3d hand pose estimation from monocular rgb. IEEE Transactions on Circuits and Systems for Video Technology, 1514–1525 (2020)

[27] Cho, J., Youwang, K., Oh, T.-H.: Cross-attention of disentangled modalities for 3d human mesh recovery with transformers. In: Proceedings of the European Conference on Computer Vision, pp. 342–359 (2022)

[28] Kim, J., Gwon, M.-G., Park, H., Kwon, H., Um, G.-M., Kim, W.: Sampling is matter: Point-guided 3d human mesh reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12880–12889 (2023)

[29] Jiang, C., Xiao, Y., Wu, C., Zhang, M., Zheng, J., Cao, Z., Zhou, J.T.: A2j-transformer: Anchor-to-joint transformer network for 3d interacting hand pose estimation from a single rgb image. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8846–8855 (2023)

[30] Cheng, W., Tang, H., Van Gool, L., Ko, J.H.: Handdiff: 3d hand pose estimation with diffusion on image-point cloud. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2274–2284 (2024)

[31] Liu, R., Ohkawa, T., Zhang, M., Sato, Y.: Single-to-dual-view adaptation for egocentric 3d hand pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 677–686 (2024)

[32] Zhou, F., Shen, P., Dai, J., Jiang, N., Hu, Y., Lai, Y.-K., Rosin, P.L.: Ahrnet: Attention and heatmap-based regressor for hand pose estimation and mesh recovery. In: IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 3000–3004 (2024)

[33] Zhao, Z., Yang, L., Sun, P., Hui, P., Yao, A.: Analyzing the synthetic-to-real domain gap in 3d hand pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12255–12265 (2025)

[34] Wang, Y., Xu, H., Heng, P.-A., Fu, C.-W.: Unihope: A unified approach for hand-only and hand-object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12231–12241 (2025)

[35] Potamias, R.A., Zhang, J., Deng, J., Zafeiriou, S.: Wilor: End-to-end 3d hand localization and reconstruction in-the-wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12242–12254 (2025)

[36] Lee, S., Park, H., Kim, D.U., Kim, J., Boboev, M., Baek, S.: Image-free domain generalization via clip for 3d hand pose estimation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 2934–2944 (2023)

[37] Guo, S., Cai, Q., Qi, L., Dong, J.: Clip-hand3d: Exploiting 3d hand pose estimation via context-aware prompting. In: Proceedings of the ACM International Conference on Multimedia, pp. 4896–4907 (2023)

[38] Zhang, X., Wang, W., Chen, Z., Xu, Y., Zhang, J., Tao, D.: Clamp: Prompt-based contrastive learning for connecting language and animal pose. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 23272–23281 (2023)

[39] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 10012–10022 (2021)

[40] Zheng, C., Liu, X., Qi, G.-J., Chen, C.: POTTER: Pooling attention transformer for efficient human mesh recovery. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1611–1620 (2023)

[41] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)

[42] Spurr, A., Dahiya, A., Wang, X., Zhang, X., Hilliges, O.: Self-supervised 3d hand pose estimation from monocular rgb via contrastive learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 11230–11239 (2021)

[43] Kapitanov, A., Kvanchiani, K., Nagaev, A., Kraynov, R., Makhliarchuk, A.: Hagrid – hand gesture recognition image dataset. In: Proceedings of the IEEE/CVF Winter Conference

on Applications of Computer Vision, pp. 4572–4581 (2024)

[44] Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International Conference on Machine Learning, pp. 1597–1607 (2020)

[45] Zimmermann, C., Ceylan, D., Yang, J., Russell, B., Argus, M., Brox, T.: Freihand: A dataset for markerless capture of hand pose and shape from single rgb images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 813–822 (2019)

[46] Sridhar, S., Mueller, F., Zollhöfer, M., Casas, D., Oulasvirta, A., Theobalt, C.: Real-time joint tracking of a hand manipulating an object from rgb-d input. In: Proceedings of the European Conference on Computer Vision, pp. 294–310 (2016)

[47] Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: International Conference on Learning Representations (2018)

[48] Hasson, Y., Varol, G., Tzionas, D., Kalevatykh, I., Black, M.J., Laptev, I., Schmid, C.: Learning joint reconstruction of hands and manipulated objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11807–11816 (2019)

[49] Choi, H., Moon, G., Lee, K.M.: Pose2mesh: Graph convolutional network for 3d human pose and mesh recovery from a 2d human pose. In: Proceedings of the European Conference on Computer Vision, pp. 769–787 (2020)

[50] Moon, G., Lee, K.M.: I2l-meshnet: Image-to-lixel prediction network for accurate 3d human pose and mesh estimation from a single RGB image. In: Proceedings of the European Conference on Computer Vision, pp. 752–768 (2020)

[51] Vasu, P.K.A., Gabriel, J., Zhu, J., Tuzel, O., Ranjan, A.: Fastvit: A fast hybrid vision transformer using structural reparameterization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 5785–5795 (2023)

[52] Baek, S., Kim, K.I., Kim, T.-K.: Pushing the envelope for RGB-based dense 3d hand pose estimation via neural rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1067–1076 (2019)

[53] Zhang, X., Li, Q., Mo, H., Zhang, W., Zheng, W.: End-to-end hand mesh recovery from a monocular RGB image. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2354–2364 (2019)

[54] Rong, Y., Shiratori, T., Joo, H.: FrankMocap: A monocular 3d whole-body pose estimation system via regression and integration. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 1749–1759 (2021)

[55] Cui, Y., Li, M., Gao, Y., Gao, C., Wu, F., Wen, H., Li, J., Sang, N.: Camera distance helps 3d hand pose estimated from a single RGB image. Graphical Models, 101179 (2023)

[56] Gu, K., Yang, L., Mi, M.B., Yao, A.: Bias-compensated integral regression for human pose estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 10687–10702 (2023)

[57] Hao, Y., Wang, S., Kan, Z.: 2d hand pose estimation from a single rgb image through flow model. In: International Conference on Advanced Robotics and Mechatronics, pp. 675–680 (2024)