國立臺灣大學電機資訊學院資訊網路與多媒體研究所

碩士論文

Graduate Institute of Networking and Multimedia

College of Electrical Engineering & Computer Science

National Taiwan University

Master's Thesis

# 以骨頭長度修正增強三維人體骨架預測

# Enhancing 3D Human Pose Estimation with Bone Length Adjustment

許智翔

Chih-Hsiang Hsu

指導教授: 張智星 博士

Advisor: Jyh-Shing Roger Jang, Ph.D.

中華民國 113 年 7 月

July, 2024

doi:10.6342/NTU202402242

# 國立臺灣大學碩士學位論文
# 口試委員會審定書
## MASTER'S THESIS ACCEPTANCE CERTIFICATE
## NATIONAL TAIWAN UNIVERSITY

以修正骨頭長度增強三維人體骨架預測

Enhancing 3D Human Pose Estimation with Bone Length Adjustment

本論文係許智翔（學號 R11944034）在國立臺灣大學資訊網路與多媒體研究所完成之碩士學位論文，於民國 113 年 7 月 10 日承下列考試委員審查通過及口試及格，特此證明。

The undersigned, appointed by the Graduate Institute of Networking and Multimedia on 10 July 2024 have examined a Master's Thesis entitled above presented by HSU, CHIH-HSIANG (student ID: R11944034) candidate and hereby certify that it is worthy of acceptance.

口試委員 Oral examination committee:

張智星

（指導教授 Advisor）

陳祝嵩　林仁俊

系（所）主管 Director: 鄭卜壬



doi:10.6342/NTU202402242

# Acknowledgements

在就讀碩士的時間裡，我最感謝我的指導教授張智星老師，每次我在研究上遇到困難時，您總能鞭辟入裡地給出一些可行的方向，引導我最終完成了這篇論文。接著要感謝陳祝嵩教授及林俊仁老師，您們對我的研究提出了很多新的觀點，讓我又對這個題目有了不同的方向。再來要感謝運動分析組的葉揚昀，很感謝你聽我分享了很多研究上無聊的細節。我也很感謝 MIRLAB 互相扶持的同學們，以及協助調整國網中心額度的王鈞右學長和王崇喆學長。我還要感謝一些非資工領域的朋友們，雖然不知道我的研究內容，但還是聽我描述研究上遇到的總總問題。最後我很感謝我的父母，支持我撐過最後漫長的寫論文時光。



doi:10.6342/NTU202402242



doi:10.6342/NTU202402242

# 摘要

現今在三維人體骨架預測的研究，主要集中於預測三維關節座標，而忽視了其他重要的物理限制，例如骨頭長度的一致性以及人體的對稱性。我們提出了骨頭長度的預測模型，模型使用循環神經網路的架構，捕捉全面的影片資訊，以達到準確的預測。為了使訓練更有效，我們合成了符合物理限制的骨頭長度資料，並提出了全新的資料增強方法。此外，我們提出了骨頭長度校正，在保持骨頭轉向的狀態下，把骨頭長度替換成我們的預測值。結果顯示，在經過骨頭長度校正後，現存的三維人體骨架預測模型都能有顯著的改善。我們更進一步使用預測出的骨頭長度，對人體骨架預測模型進行微調，也同樣能有很好的改善。我們的骨頭長度預測模型超越了過去的最佳結果，並且在 Human3.6M 資料集的多個評估方法上，校正與模型微調的方法都能有效地改善。

關鍵字：人體骨架預測、二維至三維抬升、電腦視覺、骨頭長度修正、循環神經網路



doi:10.6342/NTU202402242



doi:10.6342/NTU202402242

# Abstract

Current approaches to 3D human pose estimation primarily focus on regressing 3D joint locations, often neglecting critical physical constraints such as bone length consistency and body symmetry. This work introduces a recurrent neural network architecture designed to capture holistic information across entire video sequences, enabling accurate prediction of bone lengths. To enhance training effectiveness, we propose a novel augmentation strategy using synthetic bone lengths that adhere to physical constraints. Moreover, we present a bone length adjustment method that preserves bone orientations while substituting bone lengths with predicted values. Our results demonstrate that existing 3D human pose estimation models can be significantly enhanced through this adjustment process. Furthermore, we fine-tune human pose estimation models using inferred bone lengths, observing notable improvements. Our bone length prediction model surpasses the previous best results, and our adjustment and fine-tuning method enhance performance across several metrics on the Human3.6M dataset.



doi:10.6342/NTU202402242

**Keywords:** Human pose estimation, 2D-to-3D lifting, computer vision, bone length adjustment, recurrent neural network



doi:10.6342/NTU202402242

# Contents





doi:10.6342/NTU202402242

doi:10.6342/NTU202402242

doi:10.6342/NTU202402242



doi:10.6342/NTU202402242

# List of Figures





doi:10.6342/NTU202402242

doi:10.6342/NTU202402242

# List of Tables





doi:10.6342/NTU202402242



doi:10.6342/NTU202402242

# Denotation

| | |
|---|---|
| HPE | 人體骨架預測（Human Pose Estimation） |
| RNN | 循環神經網路（Recurrent Neural Network） |
| Bi-RNN | 雙向循環神經網路（Bi-directional Recurrent Neural Network） |
| LSTM | 長短期記憶模型 (Long Short-Term Memory) |
| GRU | 閘門循環單元（Gated Recurrent Unit） |
| Bi-GRU | 雙向閘門循環單元（Bi-directional Gated Recurrent Unit） |
| CNN | 卷積神經網路（Convolutional Neural Network） |
| MoCap | 動態捕捉（Motion Capture） |
| MPJPE | 平均關節位置誤差（Mean Per Joint Position Error） |
| CPN | 串聯金字塔網路（Cascaded Pyramid Network） |
| FPS | 影格速率（Frame Per Second） |



doi:10.6342/NTU202402242



doi:10.6342/NTU202402242

# Chapter 1 Introduction

In this chapter, we provide a brief introduction to this thesis. Section 1.1 introduces the human pose estimation and the 2D-to-3D task. Section 1.2 discusses the motivation behind this thesis. Section 1.3 describes our research topic and key contributions. Finally, Section 1.4 outlines the contents of the subsequent chapters.

## 1.1 3D Human Pose Estimation

3D human pose estimation aims to localize the 3D positions of human joints from monocular images or videos, holding significant implications for applications such as human-computer interaction, sports analysis, and medical diagnostics, due to its capacity to capture human motion. Presently, two-stage approaches dominate in 3D human pose estimation. These methods initially detect 2D keypoints from input images or videos and subsequently lift these 2D keypoints into 3D space, which is known as the 2D-to-3D lifting task.

The 2D-to-3D lifting task faces inherent challenges due to depth ambiguity: multiple 3D poses can project to the same 2D keypoints. A simple example is the flipping ambiguity described by [18], where finite possible 3D poses arise by flipping each bone forwards or backwards. This ambiguity intensifies when bone lengths are unknown or


doi:10.6342/NTU202402242

when 2D keypoints are inaccurate. Recognizing the importance of temporal information in resolving depth ambiguity, recent studies have employed recurrent neural networks (RNNs) [8, 12], temporal convolutions [2, 15], and transformer-based models [13, 24, 25] to extract temporal features.

## 1.2 Motivation

Despite recent advancements, many existing methods overlook the natural structure of human poses. Studies have shown that focusing solely on minimizing per-joint errors independently neglects overall pose coherence. Addressing this issue, bone-based representations have been proposed [20]. Chen *et al.* [2] introduced a method to decompose human poses into bone lengths and directions, simplifying the pose estimation task. However, integrating physical constraints such as bone length consistency and body symmetry remains a challenge, with significant bone length errors observed in existing works.

We evaluated bone lengths with several state-of-the-art lifting models and found that they do not predict accurate and consistent bone lengths. Although Chen *et al*. [2] trained a bone length prediction network in their model and achieved high accuracy in bone length prediction, bone length consistency is still neglected. As shown in Figure 1.1, the right forearm length changes over time.

## 1.3 Research Topic and Contribution

Inspired by previous work [2], we propose RNN-based models to predict bone lengths. Our models leverage global information from all frames of a video, rather than short se-


doi:10.6342/NTU202402242

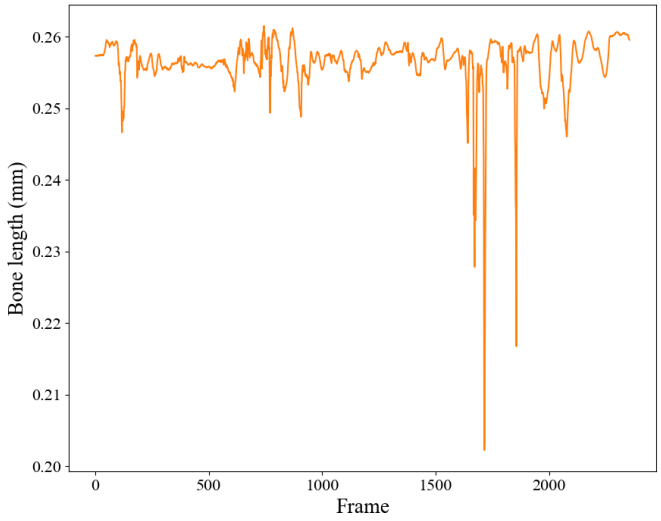


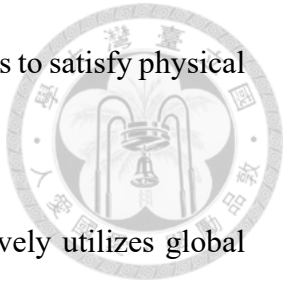

Figure 1.1: Evaluating the variation in right forearm length over time with Chen *et al*. [2] on S9 Direction 1 in Human3.6M test set.

quences. To enhance training effectiveness, we introduce a novel training time augmentation method using synthetic bone lengths generated by the statistical body shape model called SMPL [14]. This augmentation ensures that predicted bone lengths respect symmetry constraints and adhere to natural human body proportions. Unlike the method in [2] that randomly adjust bone lengths, potentially distorting body proportions, our method prioritizes realistic and accurate predictions.

We propose a novel adjustment method to enhance current state-of-the-art 2D-to-3D lifting models. This method preserves bone directions while replacing bone lengths, ensuring that the adjusted poses maintain anatomical correctness and achieve more precise joint positions. Finally, we fine-tune existing 2D-to-3D lifting models using bone length information. This fine-tuning process further improves model performance and can be applied to any 2D-to-3D lifting model, demonstrating its versatility and effectiveness.

The contributions of this work are fourfold:


doi:10.6342/NTU202402242



- We propose a new data augmentation with synthetic bone lengths to satisfy physical constraints.
- We propose a novel bone length prediction model that effectively utilizes global information.
- We introduce a bone length adjustment method that enhances existing 2D-to-3D lifting models, ensuring realistic body shapes and accurate joint positions.
- We demonstrate the efficacy of fine-tuning existing models with predicted bone lengths, thereby improving their performance in 3D human pose estimation tasks.

## 1.4 Chapter Overview

In this chapter, we provide a brief introduction to our work. In Chapter 2, we will discuss related work. Chapter 3 elaborates on our proposed methods. The experimental setup is introduced in Chapter 4, followed by a discussion of the experimental results in Chapter 5. Finally, Chapter 6 concludes the findings and points out the limitations of this thesis.


doi:10.6342/NTU202402242

# Chapter 2 Related Work

In this chapter, we will introduce previous studies related to our work. We discuss the related work on 3D human pose estimation in Section 2.1 and the related work on Recurrent Neural Network (RNN) in Section 2.2.

## 2.1 3D Human Pose Estimation

We split this section into three parts: overview, 2D keypoint detection, and 2D-to-3D lifting. In the overview, we introduce related work on solving 3D human pose estimation (Section 2.1.1), covering 2D keypoint detection (Section 2.1.2) and the 2D-to-3D lifting task (Section 2.1.3).

### 2.1.1 Overview

3D human pose estimation using deep learning methods can be categorized into two main approaches: the end-to-end approach and the two-stage approach.

The end-to-end approach predicts 3D human poses directly from RGB images. These methods [10, 11, 22] heavily rely on parametric 3D human shape models, such as SMPL [14]. They estimate parameterized pose, shape, and translation, which are then decoded


doi:10.6342/NTU202402242

via the SMPL model to obtain the human pose and shape.

In contrast, the two-stage approach first detects 2D positions of joints in an image, known as 2D keypoints. Techniques such as convolutional neural networks (CNNs) generate heatmaps that indicate the probability of joint locations [3, 19]. In the second stage, these 2D keypoints are used to estimate the 3D pose of the human figure, which is called 2D-to-3D lifting task. Currently, the two-stage approach tends to be more accurate than the end-to-end approach. End-to-end methods struggle with a lack of diversity in video data, especially variations in background, due to the complex requirements of 3D pose datasets, such as motion capture systems and high-speed cameras. This issue is mitigated in the two-stage approach since 2D keypoints can be manually labeled, and modern 2D keypoint detectors achieve high precision.

### 2.1.2 2D Keypoint Detection

There are two common approaches for 2D keypoint detection: the top-down approach and the bottom-up approach. The top-down approach first localizes the bounding box of a single subject and then detects keypoints within the bounding box [3, 19]. In contrast, the bottom-up approach directly detects all keypoints in the image, which may contain multiple subjects [6]. While the top-down approach has higher time complexity, it generally achieves superior performance compared to the bottom-up approach.

Since a human joint, like the wrist, cannot be accurately represented by a single pixel in an image, detected keypoints only approximate the positions of joints. Recent works address this by predicting heatmaps that represent the probability distributions of joints in an image. The pixel points with the highest probability are considered the positions of the


doi:10.6342/NTU202402242

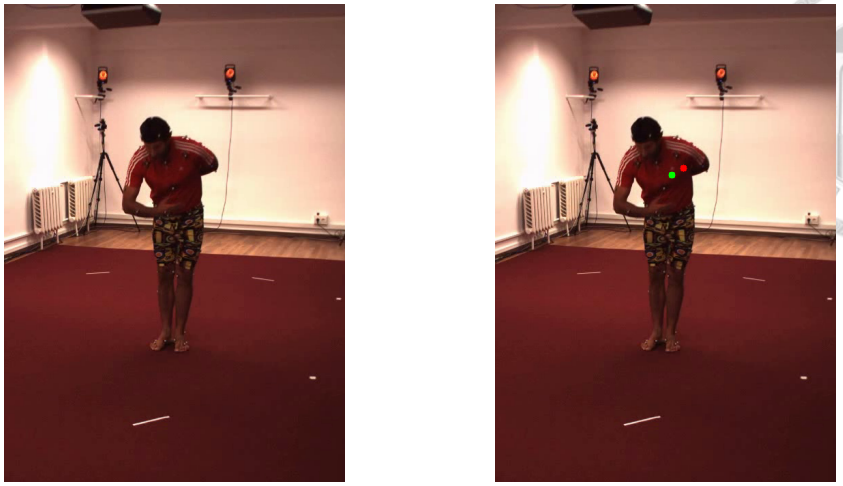
keypoints.

### 2.1.3 2D-to-3D Lifting

The 2D-to-3D lifting task is an ill-posed problem that involves predicting the additional depth dimension. Models that address this task are known as lifting models. Recent works have utilized temporal information by predicting the human pose of the central frame from a sequence of 2D keypoints in a video. The context information provides clues about the movement of the target, making the predicted poses more robust to noise. Hossain and Little [8] designed a sequence-to-sequence model based on Long Short-Term Memory (LSTM) to obtain temporally consistent 3D poses. Pavllo *et al*. [15] proposed a model based on dilated temporal convolution to capture long-term information and reduce computational overhead compared to LSTM models. Chen *et al*. [2] demonstrated that dividing the task into bone length and bone direction prediction yields better results. Zheng *et al*. [25] used transformer-based models to encode both spatial and temporal information. Li *et al*. [13] further applied transformer-based models to generate multiple plausible pose hypotheses and aggregate hypothesis features to estimate human poses. After the denoising diffusion models have emerged, a probalistic method based on diffusion models is applied to human pose estimation, *i.e*. refining predicted poses [7] or generating several hypotheses [16, 17, 23].

However, several challenges in the 2D-to-3D lifting task, such as self-occlusion and the accuracy of bone lengths, remain inadequately addressed. In the following sections, we will elaborate on these difficulties and the corresponding solutions.


doi:10.6342/NTU202402242



(a) (b)

Figure 2.1: Illustration of self-occlusion. (b) the model predicted keypoint of the wrist is labeled by a red point and the correct keypoint position is labeled by a green point.

**Self-occlusion.** As shown in Figure 2.1, the left hand is occluded by his torso, leading to inaccurate keypoint detection. Although the self-occlusion is unpreventable by the limited view from a single camera, the same joint is not always occluded in a video, enabling mitigating the noise by inferring with a sequence of keypoints that is sufficiently long.

Chen *et al*. [2] addresses this issue by incorporating the 2D keypoint visibility score, evaluated using the predicted heatmaps from keypoint detection models. The visibility score indicates the confidence level for each keypoint's visibility. When a joint is difficult to locate, such as when it is occluded or blurred, the visibility score is lower, suggesting that the predicted keypoint is more likely to be inaccurate. This score provides crucial information for lifting models to assess the reliability of keypoints, thereby enhancing robustness. Additionally, the visibility score can reveal relative depth. For instance, in Figure 2.1, the left wrist keypoint is occluded by the torso. Estimating the depth of the left wrist relative to the chest is challenging when pictorial information is lost. If the left wrist has a lower visibility score compared to the chest, it indicates that the wrist is likely behind the body


doi:10.6342/NTU202402242

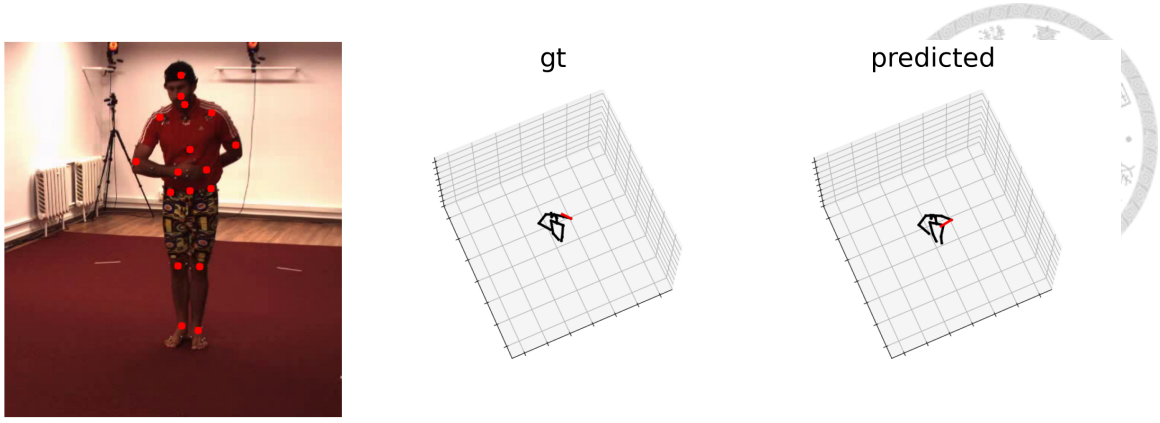



Figure 2.2: The inference result of [2] on a self-occluded video. Left: the target frame. Middle: groundtruth 3D pose. Right: predicted 3D pose.

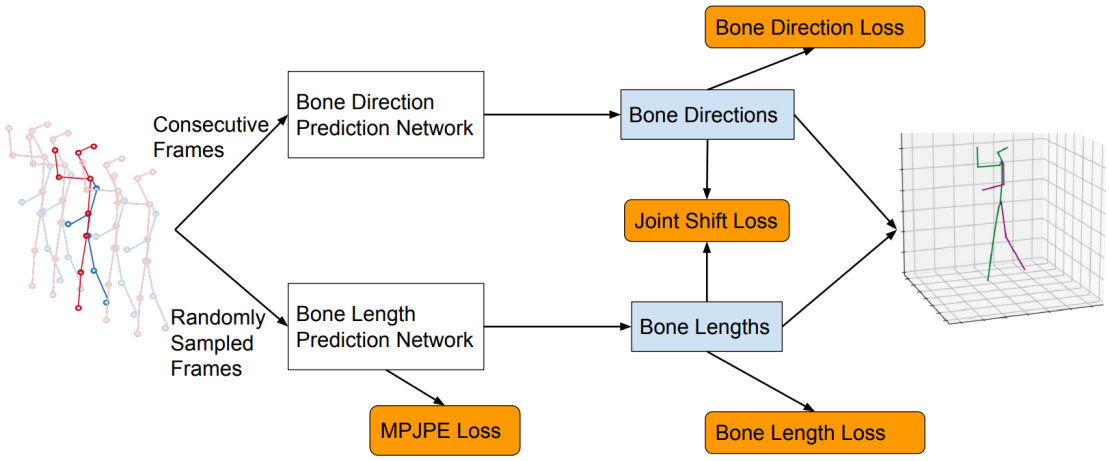



Figure 2.3: The framework of Anatomy3D [2], illustrating the bone direction prediction network and the bone length prediction network.

However, the visibility score does not always accurately reflect visibility confidence. Figure 2.2 shows a failed case where the visibility of the left wrist is the highest among all keypoints, resulting in an incorrect prediction of the left wrist being in front of the body.

**Anatomy3D.** Chen *et al*. [2] proposed an anatomy-aware approach called Anatomy3D, which simplifies the 2D-to-3D lifting task by dividing it into bone length prediction and bone direction prediction, as illustrated in Figure 2.3. The bone direction prediction network uses consecutive local frames as input as most lifting models do. The bone length network leverages the global information by randomly sampling frames across the entire video


doi:10.6342/NTU202402242

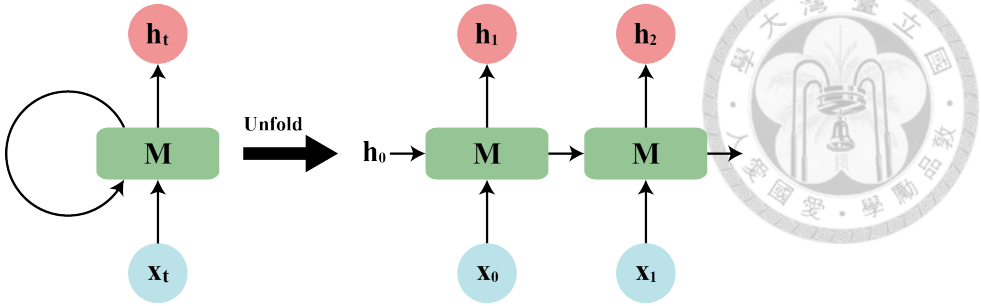



Figure 2.4: The illustration of the Recurrent Neural Network (RNN) with an unfolded workflow. $\mathbf{x}_t$: the input. $M$: the RNN model. $\mathbf{h}_t$: the hidden state.

To address the limited diversity in bone lengths in the Human3.6M dataset's training set, which contains only five subjects, they introduced a training-time augmentation for the bone length prediction task. During this augmentation process, a new set of bone lengths is randomly generated and used to replace the bone lengths of the given ground truth 3D poses, while preserving the bone directions. Additionally, the trajectory of a sequence of 3D poses is randomly shifted. Finally, the augmented 3D poses are projected onto the camera plane, resulting in the input 2D keypoints.

## 2.2 Recurrent Neural Network (RNN)

The Recurrent Neural Network (RNN) is designed for processing sequential data. Unlike traditional neural networks, an RNN has connections that form directed cycles, allowing information to persist. This makes the RNN suitable for tasks where context from previous inputs is crucial, such as time series prediction, language modeling, and speech recognition.

In an RNN, a hidden state is maintained to capture temporal information from previous inputs. At step $t$, an RNN takes an input $\mathbf{x}_t$ together with the last hidden state $\mathbf{h}_{t-1}$ to evaluate a new hidden state $\mathbf{h}_t$. This hidden state $\mathbf{h}_t$ is then passed to the next step. The


doi:10.6342/NTU202402242

update process can be written as:

$$\mathbf{h}_t = W_h \cdot \mathbf{x}_t + U_h \cdot \mathbf{h}_{t-1} + \mathbf{b}_h$$

where $W_h$, $U_h$, and $\mathbf{b}_h$ are learnable parameters.

Next, we will introduce the variations of RNN models: bi-directional RNN (Bi-RNN), long short-term memory (LSTM), and gated recurrent unit (GRU).

### 2.2.1 Bi-directional RNN (Bi-RNN)

While the RNN is designed to capture information from previous inputs, the Bi-directional RNN (Bi-RNN) adds another layer to process inputs in reverse order, known as the backward RNN. The outputs from the forward and backward RNNs capture both past and future information, which can be particularly helpful for tasks like language modeling where the context to the right of the target word is important.

### 2.2.2 Long Short-Term Memory (LSTM)

RNNs often struggle to maintain long-range dependencies and suffer from the vanishing gradient problem. LSTM networks address these issues by introducing an additional cell state and three gates: the input gate, the forget gate, and the output gate. These gates regulate the flow of information that should be passed to the next step, effectively maintaining important long-range dependencies and mitigating gradient vanishing.


doi:10.6342/NTU202402242

### 2.2.3 Gated Recurrent Unit (GRU)

Gated Recurrent Units (GRUs) [5] simplify the LSTM by combining the input gate and the forget gate, reducing computational complexity. Although GRUs simplify the model structure, there is no definitive conclusion that either the LSTM or the GRU performs better overall; their effectiveness can vary depending on the specific task and dataset.


doi:10.6342/NTU202402242

# Chapter 3 Methods

In our work, we aim to predict bone lengths that are accurate, consistent, and reasonable in body proportions. Our work consists of four main components: data augmentation, bone length prediction, bone length adjustment, and fine-tuning. Section 3.1 details our bone length augmentation method, Section 3.2 discusses our model design for predicting bone lengths, Section 3.3 explains how we apply our bone length prediction model to enhance 2D-to-3D lifting models, and Section 3.4 describes our fine-tuning methods.

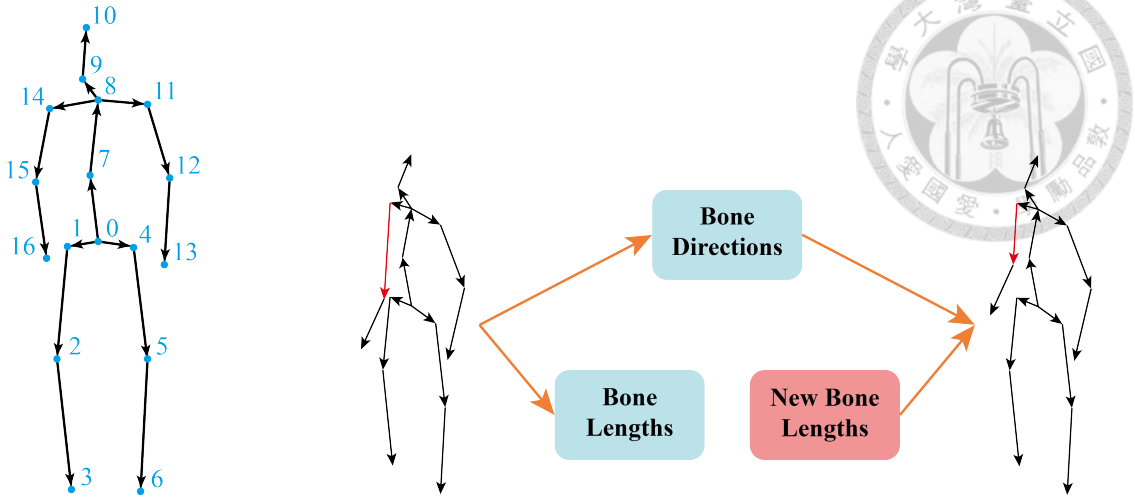
## 3.1 Bone Length Augmentation

In this section, we detail the bone length augmentation method in three parts. We first elaborate the data augmentation process (Section 3.1.1. Then we introduce the augmentation with random bone lengths (Section 3.1.2) and synthetic bone length (Section 3.1.3).

### 3.1.1 Augmentation Process

We represent a human pose $P = [\boldsymbol{p}_0 \cdots \boldsymbol{p}_{J-1}]^T \in \mathbb{R}^{J\times 3}$ with $J$ 3D joint positions as a tree structure, as shown in Figure 3.1 (a). The root joint is positioned on the pelvis and labeled as joint $0$. For each joint $\boldsymbol{p}_i \in \mathbb{R}^3$, its parent is defined as the joint closer to the


doi:10.6342/NTU202402242




(a) (b)

Figure 3.1: (a) The representation of a human pose with joint labels. (b) The overview of bone length replacement, which involves decomposing the pose into bone directions and bone lengths, and then substituting the original bone lengths with new ones.

root (*e.g.*, joint 0 is the parent of joint 1). The pose can be decomposed into bone lengths $L = [l_1 \cdots l_{J-1}]^T \in \mathbb{R}^{(J-1)\times 1}$ and bone directions $D = [\boldsymbol{d}_1 \cdots \boldsymbol{d}_{J-1}]^T \in \mathbb{R}^{(J-1)\times 3}$ using the following equations:

$$\begin{aligned} l_i &= \|\boldsymbol{p}_i - \boldsymbol{p}_{\text{parent}(i)}\|_2, \quad i = 1, \ldots, J-1 \\ \boldsymbol{d}_i &= \frac{\boldsymbol{p}_i - \boldsymbol{p}_{\text{parent}(i)}}{l_i}, \quad i = 1, \ldots, J-1 \end{aligned} \tag{3.1}$$

Here, vertices (joints) are labeled from 0 to $J-1$ and the edges (bones) are labeled from 1 to $J-1$. Given $L$ and $D$, the original pose $P$ can be reconstructed.

In the augmentation process, we first decompose a pose $P$ into bone lengths $L$ and bone directions $D$. We then use new bone lengths $L' = \left[l'_1 \cdots l'_{J-1}\right]^T$ and the original bone directions $D$ to reconstruct a new pose $\tilde{P} = [\tilde{\boldsymbol{p}}_1 \cdots \tilde{\boldsymbol{p}}_{J-1}]^T$. The bone length replacement process is illustrated in Figure 3.1 (b). A random shift $\boldsymbol{s} \in \mathbb{R}^3$ is added to the poses to enhance the augmentation. The final result is the augmented pose $P' = \left[\boldsymbol{p}'_1 \cdots \boldsymbol{p}'_{J-1}\right]^T$.


doi:10.6342/NTU202402242

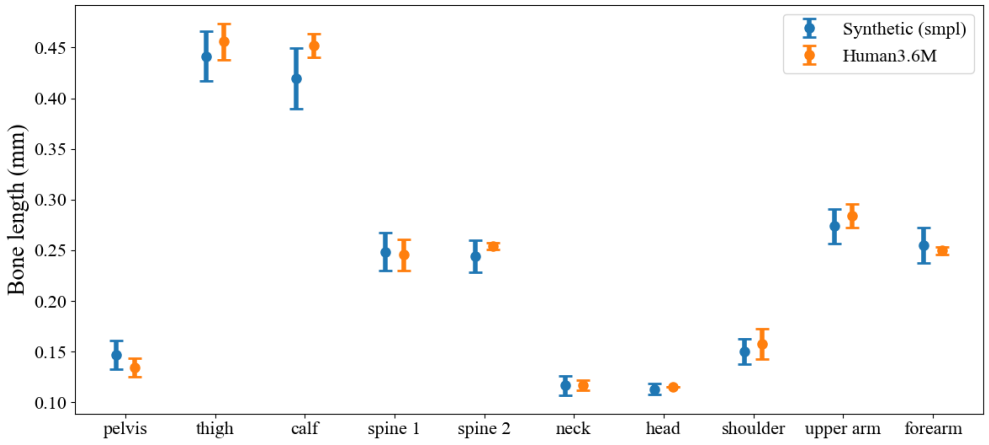



Figure 3.2: This error bar plot shows the means and the standard deviations of bone lengths in the Human3.6M dataset. Each mean value is represented by a dot, and the associated standard deviation is shown by the bars, indicating the variability around the mean bone lengths.

$$\begin{aligned} \boldsymbol{s} &\sim \mathcal{N}(\mu = 0, \sigma = 0.5) \\ \boldsymbol{p}'_i &= \tilde{\boldsymbol{p}}_i + \boldsymbol{s}, \quad i = 1, \ldots, J-1 \end{aligned} \tag{3.2}$$

For a sequence of poses, the same random shift should be applied to preserve smoothness in the trajectory. Since we use 2D keypoints as the model input, we project the pose $P'$ onto the 2D camera plane considering the camera intrinsic matrix (focal length and principal point), and both radial and tangential nonlinear lens distortion. Our model learns to predict $L'$ from the projected 2D keypoints. The key to the augmentation process is generating reasonable bone lengths $L'$.

### 3.1.2 Random Bone Lengths

Before introducing our augmentation method, we briefly discuss the approach used in [2]. They randomly adjust bone lengths $L$ based on the average bone lengths $\bar{L} =$


doi:10.6342/NTU202402242

$\left[\bar{l}'_1 \cdots \bar{l}'_{J-1}\right]^T \in \mathbb{R}^{(J-1)\times 1}$ in the batch.

$$
\begin{aligned}
&l'_i = l_i + r_i \bar{l}_i, \quad r_i \sim \mathcal{U}(-0.3, 0.3), \quad i = 1, \ldots, J-1 \\
&l'_i = l'_j, \quad \text{if they are the same body part on different side.}
\end{aligned}
\tag{3.3}
$$

The proportion varies between $-30\%$ to $30\%$, which can lead to $L'$ deviating from natural human anatomical structures. For example, this method could generate an unnaturally long forearm combined with a short upper arm. Additionally, $L'$ might lack symmetry because each bone is adjusted independently by different random values. In our experiments, we ensure symmetry by applying identical random adjustments to corresponding bones on both sides of the body.

As shown in Figure 3.2, the variability of each bone length is different. For instance, subjects in the Human3.6M dataset have similar lengths of forearms but differ in lengths of upper arms. Intuitively, we may randomly adjust the bone lengths by applying a normal distribution:

$$
\begin{aligned}
&l'_i \sim \mathcal{N}(l_i, \sigma_i), \quad i = 1, \ldots, J-1 \\
&l'_i = l'_j, \quad \text{if they are the same body part on different side.}
\end{aligned}
\tag{3.4}
$$

where the mean value is the $i$-th original length and $\sigma_i$ denotes the standard deviation of the $i$-th bone length in the Human3.6M dataset. We also maintain the symmetry in this case.



doi:10.6342/NTU202402242

### 3.1.3 Synthetic Bone Lengths

SMPL [14] is a model that generates 3D human meshes from parameters. We use SMPL to randomly generate human meshes and then evaluate the bone lengths from these meshes. This ensures that the bone lengths are symmetric and reasonable, reflecting natural body shapes. To evaluate bone lengths from meshes, we apply the joint regression matrix $\mathcal{J}$ introduced in [4] to mesh coordinates $M$:

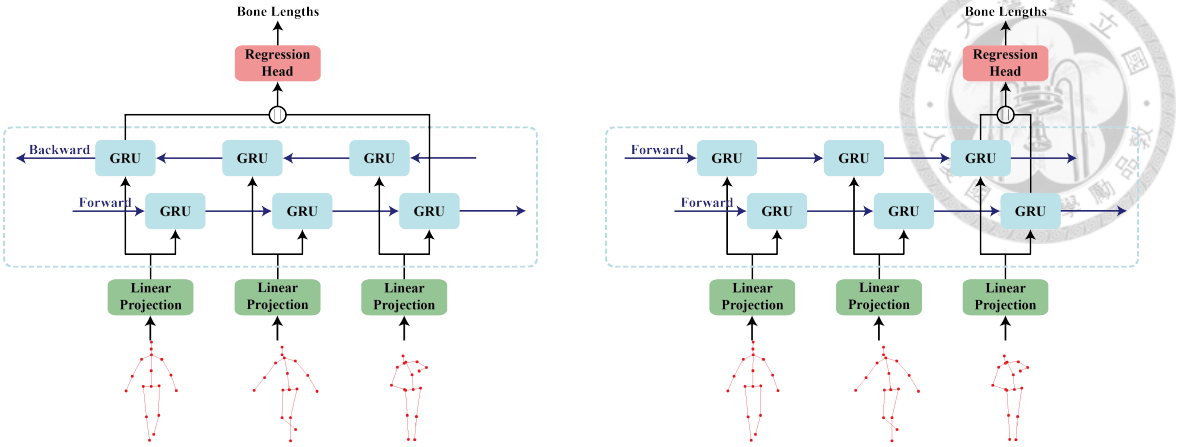
$$\tilde{L} = \mathcal{J}M \tag{3.5}$$

The 3D poses in the Human3.6M dataset are recorded using a marker-based motion capture system, where the position of each joint depends on the placement of the markers. Consequently, a single joint regression matrix cannot accurately describe the positions of the joints. When using a single joint regression matrix, the distribution of the regressed bone lengths differs from that of the Human3.6M dataset, as shown in Figure 3.2, leading to poor predictive ability. To mitigate the difference in data distribution, we align the mean value of the regressed bone lengths with the mean value in the Human3.6M dataset. After the alignment, we obtain the augmented bone lengths $L'$.

## 3.2 Bone Length Model

The structures of our models are illustrated in Figure 3.3. Our primary model, depicted in Figure 3.3 (a), utilizes a single-layer bidirectional gated recurrent unit (Bi-GRU) [5]. This Bi-GRU model processes the entire sequence of 2D keypoints from a given video, leveraging both past and future information for improved prediction accuracy.


doi:10.6342/NTU202402242




(a) Bi-GRU model (b) GRU model

Figure 3.3: The structures of our bone length prediction models. The input length is 3 for illustration.

However, due to its reliance on future data, the Bi-GRU model is not suitable for real-time online processing. To address this limitation, we also developed a GRU model, shown in Figure 3.3 (b), which updates bone lengths by processing the input keypoints frame by frame, making it suitable for online applications. In this section, we specifically introduce the Bi-GRU model.

During training, we slice the sequences of 2D keypoints into fixed-size segments for convenience. The input sequence of 2D keypoints is denoted by $X = [\boldsymbol{x}_0 \cdots \boldsymbol{x}_N] \in \mathbb{R}^{N\times(J\times 2)}$, where $N$ is the sequence length, $J$ is the number of joints, and $\boldsymbol{x}_t \in \mathbb{R}^{2J}$ is the flattened vector of 2D keypoints at frame $t$. A linear projection layer maps each $\boldsymbol{x}_t$ to a higher dimension $c$.

$$\boldsymbol{x}'_t = W_p \boldsymbol{x}_t + \boldsymbol{b}_p. \tag{3.6}$$

where $W_p$ is the weight matrix and $\boldsymbol{b}_p$ is the bias vector in the linear projection layer.

The projected vectors $\boldsymbol{x}'_t \in \mathbb{R}^c$ are then input to the GRU. The forward process at frame $t$ can be written as

$$\boldsymbol{h}_t = \mathrm{GRU}(X'_t,\ \boldsymbol{h}_{t-1}) \tag{3.7}$$


doi:10.6342/NTU202402242

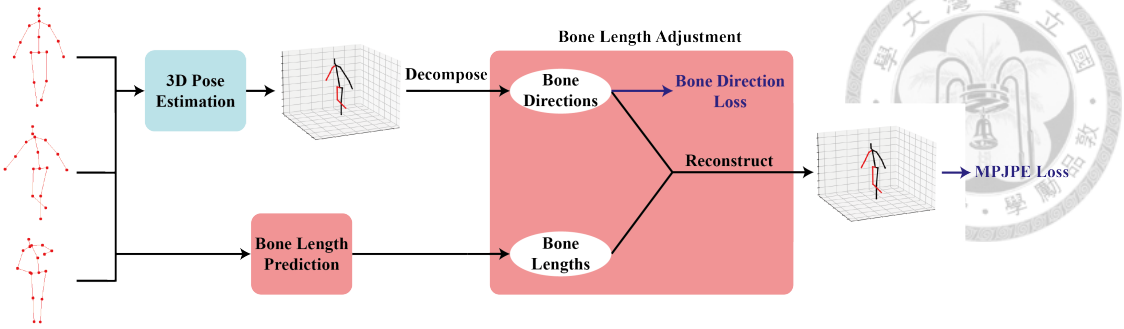



Figure 3.4: The overview of bone length adjustment. The 3D pose estimation is based on existing 2D-to-3D lifting models. The blue part is based on existing lifting models. Only the parameters in blue part are fine-tuned.

where $\boldsymbol{h}_t \in \mathbb{R}^{c'}$ is the hidden state at frame $t$ with hidden size $c'$, and the initial hidden state $\boldsymbol{h}_0$ is a zero vector. The backward process is similar but processes $X'_t$ in reverse order. We concatenate the final hidden states from the forward process and backward processes to obtain $\boldsymbol{h} \in \mathbb{R}^{2c'}$. The bone lengths $L \in \mathbb{R}^{(J-1)\times 1}$ are then regressed from $\boldsymbol{h}$ using the weight matrix $W_R$.

$$L = W_R \boldsymbol{h} \tag{3.8}$$

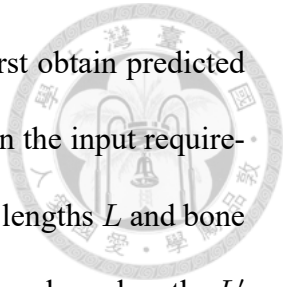
Our goal is to minimize the difference between the predicted bone lengths $L$ and the groundtruth bone lengths $\hat{L}$. The loss function is defined by the mean absolute error:

$$\mathcal{L}_L = \frac{1}{J-1} \sum_{i=1}^{J-1} \|l_i - \hat{l}_i\|_1 \tag{3.9}$$

## 3.3 Bone Length Adjustment

Figure 3.4 provides an overview of our bone length adjustment method. This technique is applied to the human poses predicted by existing 2D-to-3D lifting models. The bone length adjustment involves replacing the bone lengths of the human poses with our predicted bone lengths, as illustrated in Figure 3.1 (b).


doi:10.6342/NTU202402242



Given a sequence of 2D keypoints $X$ and a lifting model, we first obtain predicted poses $P$ from the lifting model. The sequence $X$ is segmented to fit in the input requirements of the lifting model. We then decompose the poses $P$ into bone lengths $L$ and bone directions $D$. Concurrently, we use the entire sequence $X$ to predict new bone lengths $L'$ with our model. By combining the bone directions $D$ from the lifting model and the bone lengths $L'$ from our model, we generate the reconstructed poses $P'$. This process refines the poses, ensuring a more realistic body structure.

To evaluate the adjustment process, we use the Mean Per Joint Position Error (MPJPE) to measure the error between the reconstructed pose $P'$ and the groundtruth pose $\hat{P}$:

$$\mathcal{L}_P = \frac{1}{J}\sum_{i=0}^{J-1} \|\boldsymbol{p}'_i - \hat{\boldsymbol{p}}_i\|_2 \tag{3.10}$$

## 3.4 Fine-tuning

Fine-tuning the entire model (both the bone length model and the lifting model) is very challenging. Therefore, we considered first fine-tune the lifting model while keeping the parameters in the bone length model fixed. In this section, we introduce two fine-tuning methods: fine-tuning solely on the lifting model (Section 3.4.1 and fine-tuning the entire model (Section 3.4.2.

### 3.4.1 Fine-tuning on Lifting Model

we decided to fix the parameters of the length model for two main reasons. First, bone directions are more challenging to learn compared to bone lengths, leading to er-


doi:10.6342/NTU202402242

roneous bone directions. For instance, as discussed in Section 2.1.3, the predicted bone direction may point forward, when it should point backward. With these failed directions, the model struggles to find suitable bone lengths that the reconstructed poses are close to the groundtruth poses, thereby reducing learning efficiency in bone length prediction.

Second, the input keypoints differ between the lifting model and the bone length model. In the two-stage estimation, 2D-to-3D lifting follows keypoint detection, so previous works use predicted keypoints as input. However, when applying data augmentation, keypoints are obtained by projecting augmented poses, making them groundtruth keypoints. This discrepancy makes it unsuitable to fine-tune the lifting model and the bone length model simultaneously.

In our adjustment process, bone lengths can also enhance the lifting models' ability to predict bone directions. We propose a fine-tuning method based on our adjustment process. Since the lifting models are trained with predicted 2D keypoints, we can not apply data augmentation that generates groundtruth keypoints. To prevent overfitting, the bone length prediction model is fixed during this process. We fix the weights of our bone length model and fine-tune the lifting models by minimizing the error in the predicted bone directions and the MPJPE of the reconstructed pose $P'$. The direction loss is defined as:

$$\mathcal{L}_D = \frac{1}{J-1} \sum_{i=1}^{J-1} \|\boldsymbol{d}_i - \hat{\boldsymbol{d}}_i\|_2 \tag{3.11}$$

where $\boldsymbol{d}_i$ is the predicted direction and $\hat{\boldsymbol{d}}_i$ is the groundtruth direction of the $i$-th bone. The total loss combines both the direction loss and the position error loss:


doi:10.6342/NTU202402242

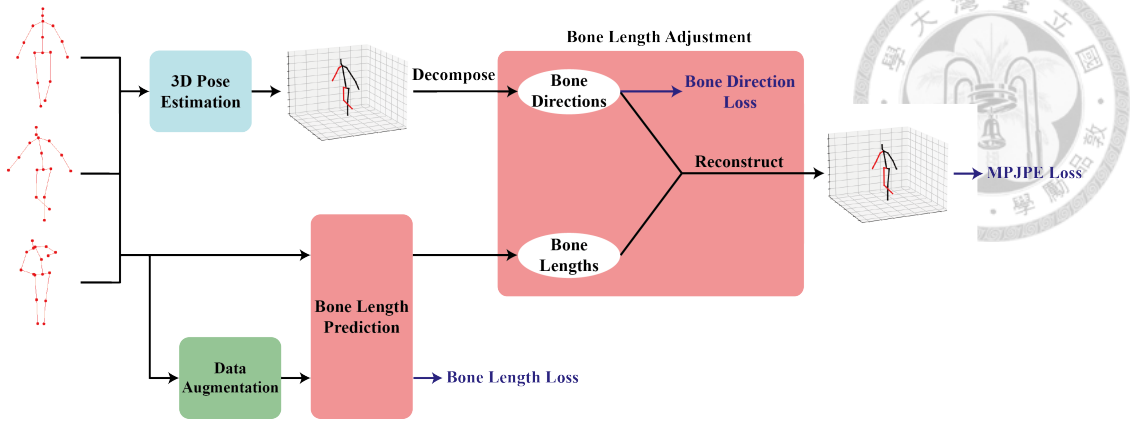



Figure 3.5: The overview of the fine-tuning method on the entire model. The blue part is based on existing lifting models.

$$\mathcal{L} = \mathcal{L}_D + \mathcal{L}_P \tag{3.12}$$

### 3.4.2 Fine-tuning on the Entire Model

To achieve a comprehensive evaluation of our bone length model, we fine-tune the entire adjustment process, including the bone length prediction component. We incorporate data augmentation in the fine-tuning process, as illustrated in Figure 3.5.

Given a sequence of 2D keypoints $X$, we generate augmented keypoints $X'$ and the corresponding bone lengths $\hat{L}'$. In the lifting model branch, we first predict the 3D pose using the lifting model and then decompose it into bone directions $D$. The direction loss is the same in 3.11.

In the bone length model branch, we predict the target bone lengths using the keypoints $X$, resulting $L$. For the augmented data, we predict $L'$ from the augmented keypoints $X'$. To prevent overfitting, we only evaluate the bone length loss between $L'$ and the augmented data $\hat{L}'$:


doi:10.6342/NTU202402242

$$\mathcal{L}_{L_{aug}} = \frac{1}{J-1} \sum_{i=1}^{J-1} \|l'_i - \hat{l}'_i\|_1 \tag{3.13}$$

We reconstruct the final pose $P'$ with the bone directions $D$ and the bone lengths $L$. The MPJPE loss is evaluated to fine-tune both the lifting model and the bone length model. The total loss function is given by:

$$\mathcal{L} = \mathcal{L}_D + \mathcal{L}_P + \mathcal{L}_{L_{aug}} \tag{3.14}$$



doi:10.6342/NTU202402242



doi:10.6342/NTU202402242

# Chapter 4 Experimental setup

In this chapter, we first discuss the dataset in Section 4.1 and the evaluation metrics in Section 4.2. We then provide implementation details, including the environment in Section 4.3 and parameter settings in Section 4.4. Section 4.5 outlines the experiments conducted. The results of these experiments are discussed in Chapter 5.

## 4.1 Human3.6M Dataset

The Human3.6M dataset [1, 9] is the widely used Motion Capture (MoCap) dataset in the field of human pose estimation. It contains 3.6 million frames featuring 11 actors (5 females and 6 males) performing 15 diverse actions, such as walking, taking photos, and sitting. The dataset includes high-resolution video recorded by four synchronized cameras operating at 50 Hz, providing diverse perspectives for each action. Seven subjects are annotated with 3D poses, captured using a high-speed MoCap system. Following the standard protocol in prior works [2, 13, 15, 24, 25], we train our model on five subjects (S1, S5, S6, S7, S8) and test on two subjects (S9, S11), using a 17-joint skeleton.


doi:10.6342/NTU202402242

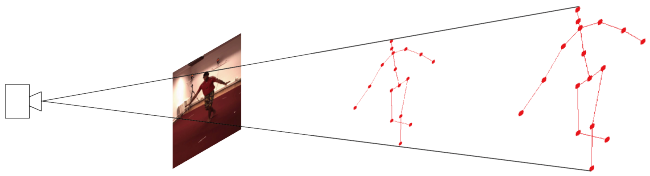


Figure 4.1: Two poses with similar body shapes having identical keypoints

## 4.2 Evaluation Metrics

For bone length evaluation, we use the bone length error as described by Equation 3.9, comparing the predicted to the groundtruth lengths.

For human poses, we use two protocols: Protocol 1 measures the Mean Per Joint Position Error (MPJPE), the average Euclidean distance between the predicted and groundtruth joint positions, and Protocol 2 (P-MPJPE) reports the error after applying Procrustes analysis, which aligns the predicted poses with the groundtruth in terms of translating, rotating, and uniform scaling.

Since the distance between the subject and the camera is unknown, poses after scaling and translating can be projected to identical keypoints, as shown in Figure 4.1. Although the two poses in the figure appear similar in body shape, the MPJPE between them is large. By applying Procrustes analysis, the poses become aligned, enabling us to better evaluate the accuracy of the predicted body structure.


doi:10.6342/NTU202402242

## 4.3 Environment

Our experiments are tested on two different devices. We trained all of the models on TWCC. For the inference speed, we test on our local device. On both devices, we test under a single GPU. The settings are as following:

- TWCC
  - CPU: Intel(R) Xeon(R) Platinum 8280 CPU @ 2.70GHz
  - RAM: 90 GB
  - GPU: NVIDIA Tesla V100 SXM2
- Local device
  - CPU: 12th Gen Intel(R) Core(TM) i5-12400 @ 2.50 GHz
  - RAM: 32 GB
  - GPU: NVIDIA GeForce RTX 3060 Ti

## 4.4 Parameter Settings

We discuss the parameter settings for bone length prediction in Section 4.4.1 and for fine-tuning in Section 4.4.2.

### 4.4.1 Bone Length Prediction

We align the mean value of our synthetic bone lengths with the Human3.6M dataset. Our study evaluates five different methods for generating bone lengths, detailed in Section


doi:10.6342/NTU202402242

3.1:

- **Random Uniform Distribution**: Randomly generated bone lengths from a uniform distribution.
- **Random Normal Distribution (Training Set Std)**: Bone lengths generated from a normal distribution using standard deviations from the training set.
- **Random Normal Distribution (Human3.6M Std)**: Bone lengths generated from a normal distribution using standard deviations from both the training and test sets.
- **Synthetic Aligned with Training Set**: Synthetic bone lengths aligned with the mean values in the training set.
- **Synthetic Aligned with Human3.6M**: Synthetic bone lengths aligned with the mean values in both the training and test sets.

During training, we use a sequence length $N = 512$ and utilize the entire sequence during testing. The projected dimension $c$ is set to $256$, and the hidden state dimension $c'$ is set to $512$. We train our models using the Adam optimizer with an exponentially decaying learning rate schedule. The initial learning rate is set to $0.0001$, and it decays by a factor of $0.95$ each epoch. The batch size is set to $256$. We train our model with groundtruth 2D keypoints which are projected from synthetic poses and test our model with 2D keypoints predicted by the Cascaded Pyramid Network (CPN) [3].

### 4.4.2 Fine-tuning

For fine-tuning, we select Pavllo *et al*. [15] as our fine-tuning target. We configure the sequence length $N$ to $243$, and apply horizontal flip augmentation during both training


doi:10.6342/NTU202402242

and testing, following their settings. We fine-tune the model using the Adam optimizer and a batch-normalization momentum set to the final state $0.001$. Similarly, we employ an exponentially decaying learning rate schedule, starting at $0.00004$ with a decay factor of $0.95$ per epoch. The batch size for fine-tuning is set to $1024$, consistent with their work. We utilize 2D keypoints predicted by CPN for both training and testing phases. Finally, the horizontal flip augmentation is applied at train and test time, following previous works [2, 15, 25].

## 4.5 Roadmap of Experiments

We conducted six experiments in this thesis:

- Bone Length Prediction

  We compare the results on both the GRU model and the Bi-GRU model with random and synthetic augmentations. We compare our best results to previous works.

- Bone Length Adjustment

  We apply the adjustment to several existing lifting models using bone lengths predicted by our GRU model and Bi-GRU model. We compare the results before and after the adjustment.

- Fine-tuning

  We report two different settings: fine-tune the lifting model and fine-tune the entire model. We compare the results before and after fine-tuning.

- Inference Speed

  We test the inference speed in real-time processing and compare the results of our


doi:10.6342/NTU202402242

GRU model to previous works.

- Abaltion Study

  We test our model with different settings and compare the results.



doi:10.6342/NTU202402242

# Chapter 5 Results

In this chapter, we discuss the results for the bone length prediction model in Section 5.1, the bone length adjustment in Section 5.2, and the fine-tuning in Section 5.3. We then present the inference speed of our bone length model and bone length adjustment in Section 5.4. Finally, we conduct an ablation study and present the results in Section 5.5.

## 5.1 Bone Length Prediction Model

Figure 5.1 presents the outcomes of our Bi-GRU bone length model evaluation. Utilizing synthetic bone lengths during training time augmentation yields the lowest overall bone length error among all methods evaluated. The random uniform distribution method fail to generate bone lengths that adhere to natural human body proportions, resulting in poor performance. Conversely, synthetic methods demonstrate superior performance over random methods, even when not using the mean values in the test set.

Table 5.1 shows the comparison of bone lengths. For the lifting models, We use the off-the-shelf pretrained models to evaluate the 3D poses. The error is evaluated by decomposing the predicted poses into bone lengths. We report the results on our device, which might be slightly different from what they claimed. For the diffusion-based methods [16, 17, 23] that generates multiple hypotheses, we report the results with the deter-


doi:10.6342/NTU202402242

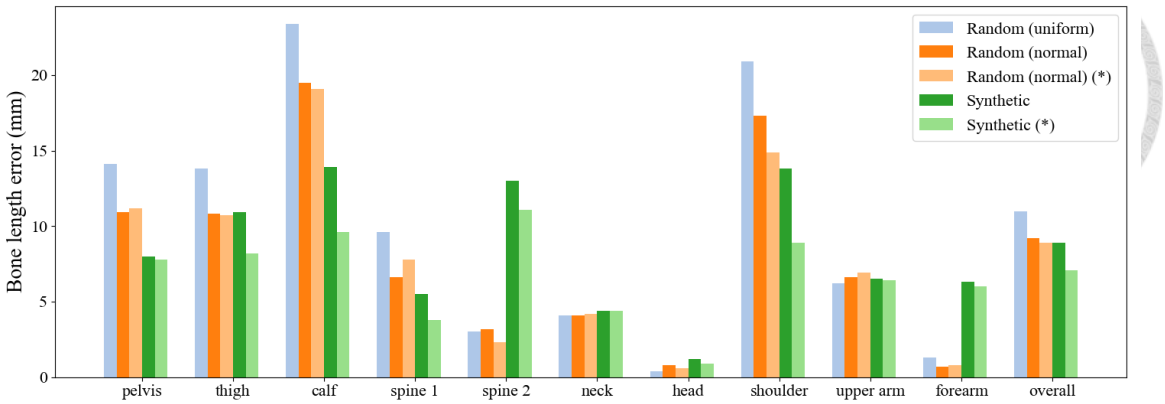



Figure 5.1: The average bone length error comparison across all frames of the test set in Human3.6M. (∗) involving statistics in the test set.

Table 5.1: Quantitative comparison of bone length error. Best in bold and second best underlined. (∗) including the mean values in the test set. (†) bone length model.

| | | Bone length error ↓ (mm) |
|---|---|---|
| Pavllo *et al*. [15] (T=243) | CVPR'19 | 12.3 |
| Chen *et al*. [2] (T=50) (†) | TCSVT'21 | 10.3 |
| Chen *et al*. [2] (T=243) | TCSVT'21 | 8.9 |
| Zheng *et al*. [25] (T=81) | ICCV'21 | 10.8 |
| Li *et al*. [13] (T=351) | CVPR'22 | 10.3 |
| Zhang *et al*. [24] (T=243) | CVPR'22 | 11.0 |
| Gong *et al*. [7] (T=243) | CVPR'23 | <u>8.5</u> |
| Shan *et al*. [17] (T=243) | ICCV'23 | 10.6 |
| Peng *et al*. [16] (T=243) | CVPR'24 | 10.9 |
| Xu *et al*. [23] (T=243) | CVPR'24 | 12.2 |
| Ours, GRU (synthetic) (T=all frames) (*)(†) | | **7.1** |
| Ours, Bi-GRU (synthetic) (T=all frames) (*)(†) | | **7.1** |
| Ours, Bi-GRU (synthetic) (T=all frames) (†) | | 8.9 |

ministic joint-level aggregation. Our model achieves the state-of-the-art result when not using the mean values in the test set. Additionally, the GRU model designed for online processing performs comparably to the Bi-GRU model. The synthetic method incorporating the test set statistics notably outperforms all other results. Given that the training set comprises data from only five subjects, the statistics may not fully represent broader variations, leading to significant disparities between using and not using the mean values in the test set. With a more comprehensive dataset, our approach could potentially circumvent such limitations.


doi:10.6342/NTU202402242

Table 5.2: Action-wise bone length error on Human3.6M with our Bi-GRU model. Best in bold. Unit: millimeter

| Model | Dir. | Disc. | Eat | Greet | Phone | Photo | Pose | Purch. | Sit | SitD. | Smoke | Wait | WalkD. | Walk | WalkT. | Avg |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Pavllo *et al.* [15] | 13.2 | 11.9 | 10.7 | 11.3 | 12.2 | 13.7 | 11.3 | 11.7 | 14.1 | 16.2 | 12.8 | 11.1 | 13.1 | 10.4 | 10.3 | 12.3 |
| Chen *et al.* [2] | 9.4 | 9.3 | 8.9 | 8.2 | 8.4 | 8.5 | 8.5 | 8.6 | 10.1 | 10.1 | 8.9 | 8.1 | 9 | 8.6 | 8.9 | 8.9 |
| Zheng *et al.* [25] | 11.1 | 10.8 | 9.9 | 10.0 | 10.6 | 12 | 10.1 | 10.6 | 11.7 | 13.1 | 11.1 | 10.2 | 11.5 | 9.2 | 9.4 | 10.8 |
| Li *et al.* [13] | 10.6 | 10.2 | 9.4 | 9.8 | 10.5 | 11.9 | 9.4 | 9.8 | 12.6 | 13.7 | 10.7 | 9.3 | 10.3 | 8.2 | 8.2 | 10.3 |
| Zhang *et al.* [24] | 11.4 | 11.2 | 10.2 | 10.8 | 10.8 | 12.1 | 10 | 10.8 | 12.1 | 14.9 | 11.2 | 10.4 | 11.5 | 8.9 | 8.8 | 11.0 |
| Gong *et al.* [7] | 8.8 | 8.2 | 8.1 | 8.5 | 8.0 | 9.6 | 7.7 | 8.3 | 9.4 | 11.9 | 8.7 | 7.5 | 8.9 | **7.0** | 7.2 | 8.5 |
| Shan *et al.* [17] | 11.3 | 10.6 | 10.2 | 10.3 | 10.2 | 11.7 | 9.8 | 10.3 | 11.6 | 13.6 | 11.0 | 10.1 | 10.3 | 8.8 | 8.9 | 10.6 |
| Peng *et al.* [16] | 11.1 | 10.7 | 10.2 | 10.2 | 10.5 | 12.3 | 9.9 | 11.0 | 12.0 | 14.7 | 10.8 | 10.0 | 11.3 | 9.6 | 9.6 | 10.9 |
| Xu *et al.* [23] | 13.5 | 12.7 | 11.4 | 12.1 | 11.7 | 13.4 | 11.2 | 11.6 | 13.1 | 15.8 | 12.5 | 11.7 | 12.0 | 10.1 | 10.7 | 12.2 |
| Ours, GRU | **7.5** | 7.3 | 7.5 | **6.7** | **7.1** | **6.6** | **6.6** | **6.7** | **7.1** | 7.6 | 7.5 | **6.3** | **7.2** | 7.5 | 7.4 | **7.1** |
| Ours, Bi-GRU | **7.5** | **7.2** | **7.2** | 7.1 | 7.3 | 7 | 6.7 | 6.8 | 7.2 | **6.9** | **7.3** | 6.7 | 7.3 | 7.7 | **7.1** | **7.1** |

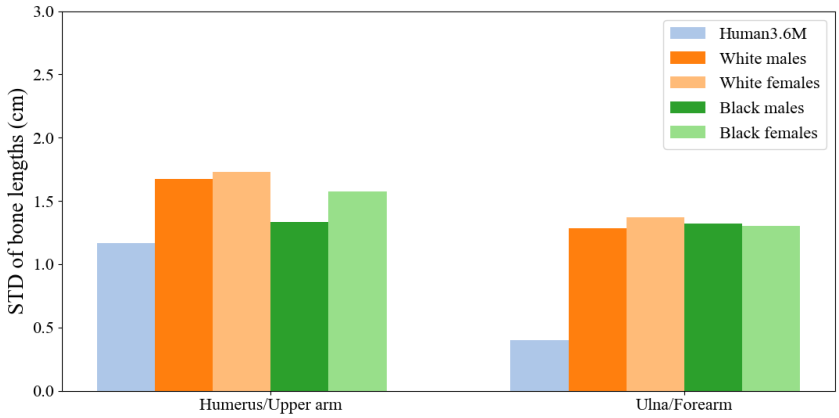



Figure 5.2: Comparison between the standard deviation of real bone lengths and bone lengths in Human3.6M.

Table 5.2 presents the action-wise comparison of bone lengths. The actions ”Walk” and ”Walk Together” are dynamic, providing sufficient information for lifting models to predict accurate 3D poses. For all models except for [2], we observe a significant performance gap between these dynamic actions and others. Conversely, [2] trained a bone length prediction network, resulting in consistent performance across all actions. Our models exhibit similar consistent performance, demonstrating their robustness across different types of actions.

Examining Figure 3.2, we observe that the standard deviations of certain bones, such as spine 2, neck, head, and forearm in the Human3.6M dataset are exceptionally small. As shown in Figure 5.2, anthropometric research [21] indicates similar standard deviations


doi:10.6342/NTU202402242

for lengths of the humerus (upper arm) and ulna (forearm). However, in Human3.6M, standard deviations for the lengths of the upper arm and forearm differ significantly. Additionally, the standard deviation of the upper arm closely matches the anthropometric result for the humerus, while the standard deviation of the lower arm is much smaller than the anthropometric result for the ulna. These discrepancies may arise from limitations in the transformation process from MoCap raw data to human poses, potentially influenced by constraints on body shape or inaccuracies in marker placement within the Human3.6M dataset.

Even with dataset constraints, our synthetic method consistently outperforms random methods and the lifting models. We select the model trained with synthetic bone lengths using the mean value of the entire datset as our final model.

## 5.2 Bone Length Adjustment

Table 5.3 illustrates the reconstruction error of existing lifting models before and after applying our adjustment method. The results reported are tested on our device. For the models generating multi-hypotheses [16, 17, 23], we use the deterministic joint-level aggregation to obtain the final poses. Across all tested models, we observe consistent performance improvements under both protocol 1 (MPJPE) and protocol 2 (P-MPJPE) after adjustment with both GRU and Bi-GRU models. Our Bi-GRU model outperforms the GRU model, showing the advantage of utilizing future information. The degree of improvement correlates with the bone length error inherent in the original models. Models with larger initial bone length errors, such as Pavllo *et al*. [15], demonstrate significant enhancement, achieving a 3% reduction in Protocol 1 error with the adjustment. In contrast,


doi:10.6342/NTU202402242

Table 5.3: Quantitative comparison of the adjustment process on reconstruction error evaluated on Human3.6M under MPJPE and P-MPJPE. Best results of the same base model are in bold.

| Base model | | Bone length error ↓ (mm) | Bone length model | MPJPE ↓ (mm) | P-MPJPE ↓ (mm) |
|---|---|---|---|---|---|
| Pavllo *et al*. [15] | CVPR'19 | 12.3 | ✗ | 46.8 | 36.5 |
| | | | GRU | 45.6 | 36.1 |
| | | | Bi-GRU | **45.2** | **35.8** |
| Chen *et al*. [2] | TCSVT'21 | 8.9 | ✗ | 44.2 | 35.0 |
| | | | GRU | 44.0 | 34.8 |
| | | | Bi-GRU | **43.5** | **34.5** |
| Zheng *et al*. [25] | ICCV'21 | 10.8 | ✗ | 44.3 | 34.6 |
| | | | GRU | 43.3 | 34.1 |
| | | | Bi-GRU | **42.9** | **33.8** |
| Li *et al*. [13] | CVPR'22 | 10.3 | ✗ | 43.0 | 34.5 |
| | | | GRU | 42.5 | 34.0 |
| | | | Bi-GRU | **42.2** | **33.7** |
| Zhang *et al*. [24] | CVPR'22 | 11.0 | ✗ | 40.9 | 32.7 |
| | | | GRU | 40.6 | 32.5 |
| | | | Bi-GRU | **40.2** | **32.2** |
| Gong *et al*. [7] | CVPR'23 | 8.5 | ✗ | 39.5 | 31.2 |
| | | | GRU | 39.4 | 31.1 |
| | | | Bi-GRU | **39.0** | **30.8** |
| Shan *et al*. [17] | ICCV'23 | 10.6 | ✗ | 39.6 | 31.7 |
| | | | GRU | 38.9 | 31.2 |
| | | | Bi-GRU | **38.6** | **31.0** |
| Peng *et al*. [16] | CVPR'24 | 10.9 | ✗ | 40.2 | 32.2 |
| | | | GRU | 39.9 | 31.9 |
| | | | Bi-GRU | **39.4** | **31.5** |
| Xu *et al*. [23] | CVPR'24 | 12.2 | ✗ | 40.2 | 32.9 |
| | | | GRU | 40.1 | 32.5 |
| | | | Bi-GRU | **39.6** | **32.2** |



doi:10.6342/NTU202402242

Table 5.4: Action-wise reconstruction error on Human3.6M before and after adjustment with our Bi-GRU model. The top table shows the result under protocol 1. The bottom table shows the result under protocol 2. Best in bold. Red for better results before the adjustment. Unit: millimeter

| Protocol 1 | Dir. | Disc. | Eat | Greet | Phone | Photo | Pose | Purch. | Sit | SitD. | Smoke | Wait | WalkD. | Walk | WalkT. | Avg |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Pavllo *et al.* [15] | 45.2 | 46.7 | 43.3 | 45.6 | 48.1 | 55.1 | 44.6 | 44.3 | 57.3 | 65.8 | 47.1 | 44.0 | 49.0 | 32.8 | 33.9 | 46.8 |
| Adjusted | **41.9** | **45.2** | **42.1** | **44.3** | **46.2** | **53.5** | **43.4** | **42.1** | **55.6** | **64.1** | **45.5** | **42.3** | **47.1** | **32.0** | **32.7** | **45.2** |
| Chen *et al.* [2] | 41.5 | 43.8 | 39.8 | 43.1 | 46.2 | 52.5 | 42.2 | 41.8 | 54.1 | 60.7 | 45.5 | 41.6 | 46.0 | 31.4 | 32.4 | 44.2 |
| Adjusted | **40.0** | **43.0** | **39.6** | **42.8** | **45.9** | **52.5** | **41.4** | **41.0** | **53.1** | **60.0** | **44.9** | **40.9** | **45.2** | **31.3** | **31.7** | **43.5** |
| Zheng *et al.* [25] | 41.5 | 44.8 | 39.8 | 42.5 | 46.5 | 51.6 | 42.1 | 42.0 | 53.3 | 60.7 | 45.5 | 43.3 | 46.1 | 31.8 | 32.2 | 44.3 |
| Adjusted | **38.9** | **43.7** | **38.9** | **41.3** | **45.0** | **50.2** | **41.1** | **39.8** | **52.2** | **59.5** | **44.1** | **41.9** | **44.3** | **31.2** | **31.5** | **42.9** |
| Li *et al.* [13] | 39.2 | 43.1 | 40.1 | 40.9 | 45.0 | 51.2 | 40.6 | 41.3 | 53.6 | 60.4 | 43.7 | 41.1 | 43.9 | **29.8** | **30.6** | 43.0 |
| Adjusted | **37.7** | **42.5** | **39.2** | **40.3** | **43.7** | **50.3** | **40.1** | **40.3** | **52.2** | **59.3** | **42.8** | **40.4** | **43.1** | 30.1 | 30.7 | **42.2** |
| Zhang *et al.* [24] | 37.9 | 40.9 | 38.5 | 39.6 | 41.9 | 49.4 | 39.5 | 40.1 | 51.5 | 55.4 | 42.0 | 39.7 | 41.2 | **27.8** | **28.1** | 40.9 |
| Adjusted | **36.2** | **40.4** | **37.6** | **38.5** | **41.6** | **49.1** | **38.7** | **38.3** | **51.2** | **54.2** | **41.4** | **38.8** | **40.6** | 28.1 | 28.3 | **40.2** |
| Gong *et al.* [7] | 35.6 | 39.5 | 36.9 | 38.2 | 40.6 | 47.6 | 38.4 | 38.5 | 50.6 | 53.2 | 40.8 | 38.1 | 40.1 | **26.9** | **27.1** | 39.5 |
| Adjusted | **34.9** | **39.2** | **36.2** | **37.6** | **40.2** | **47.1** | **38.1** | **37.5** | **49.4** | **52.3** | **40.1** | **37.7** | **39.9** | 27.9 | 27.7 | **39.0** |
| Shan *et al.* [17] | 37.5 | 39.7 | 36.2 | 37.9 | 41.1 | 47.7 | 38.6 | 38.1 | 50.0 | 52.4 | 41.1 | 39.0 | 39.9 | **27.2** | 27.3 | 39.6 |
| Adjusted | **35.7** | **38.6** | **35.2** | **36.3** | **40.5** | **46.7** | **37.6** | **36.9** | **49.1** | **51.7** | **39.9** | **37.8** | **39.4** | 27.5 | **26.9** | **38.6** |
| Peng *et al.* [16] | 37.7 | 39.9 | 36.5 | 37.8 | 41.7 | 47.5 | 38.3 | 39.8 | 52.4 | 55.6 | 41.2 | 40.0 | 39.9 | **26.7** | 27.4 | 40.2 |
| Adjusted | **35.6** | **39.3** | **35.5** | **36.8** | **41.5** | **47.0** | **37.2** | **38.2** | **51.9** | **55.3** | **40.5** | **38.9** | **39.6** | **26.7** | **27.2** | **39.4** |
| Xu *et al.* [23] | 39.6 | 41.2 | 36.3 | 38.5 | **41.6** | 45.3 | 38.1 | 38.5 | 51.6 | 54.4 | 41.8 | 40.2 | 40.3 | **27.8** | **28.2** | 40.2 |
| Adjusted | **37.3** | **40.7** | **35.6** | **37.7** | **41.6** | **45.1** | **37.6** | **37.8** | **50.7** | **54.1** | **40.7** | **39.1** | **40.1** | 28.1 | 28.3 | **39.6** |

| Protocol 2 | Dir. | Disc. | Eat | Greet | Phone | Photo | Pose | Purch. | Sit | SitD. | Smoke | Wait | WalkD. | Walk | WalkT. | Avg |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Pavllo *et al.* [15] | 34.1 | 36.1 | 34.4 | 37.2 | 36.4 | 42.2 | 34.4 | 33.6 | 45.0 | 52.5 | 37.4 | 33.8 | 37.8 | 25.6 | 27.3 | 36.5 |
| Adjusted | **32.8** | **35.2** | **33.1** | **36.2** | **35.3** | **42** | **33.8** | **33.1** | **44.1** | **52.0** | **36.5** | **33.1** | **37.3** | **25.0** | **27.0** | **35.8** |
| Chen *et al.* [2] | 33.0 | 35.3 | 32.6 | 35.4 | 35.8 | 40.4 | 32.9 | 32.5 | 42.3 | 49.7 | 36.9 | 32.5 | 36.1 | 25.0 | 26.3 | 35.1 |
| Adjusted | **31.6** | **34.3** | **32.0** | **34.6** | **35.1** | **40.3** | **32.2** | **32.2** | **42.1** | **49.2** | **36.2** | **31.8** | **35.5** | **24.3** | **25.6** | **34.5** |
| Zheng *et al.* [25] | 32.5 | 34.8 | 32.6 | 34.6 | 35.3 | 39.5 | 32.2 | 32 | 42.8 | 48.5 | 36.5 | 32.4 | 35.3 | 24.5 | 26.0 | 34.6 |
| Adjusted | **30.9** | **33.9** | **31.4** | **33.4** | **34.4** | **39.2** | **31.4** | **31.3** | **42.0** | **48.1** | **35.5** | **31.7** | **34.7** | **23.6** | **25.2** | **33.8** |
| Li *et al.* [13] | 31.7 | 34.9 | 32.8 | 33.9 | 35.3 | 39.6 | 31.9 | 32.3 | 43.6 | 49.0 | 36.3 | 32.6 | 34.5 | 23.8 | 25.1 | 34.5 |
| Adjusted | **30.5** | **34** | **31.6** | **32.9** | **34.1** | **39.2** | **31.1** | **31.9** | **42.5** | **48.6** | **35.3** | **31.9** | **33.9** | **23.2** | **24.7** | **33.7** |
| Zhang *et al.* [24] | 31.1 | 33.3 | 31.3 | 32.1 | 32.9 | **38.7** | 30.7 | 31.2 | 42.5 | 44.6 | 34.1 | 30.7 | 32.8 | 21.9 | 23.0 | 32.7 |
| Adjusted | **30.0** | **32.8** | **30.5** | **31.2** | **32.7** | 38.8 | **30.1** | **30.5** | **42.0** | **44.3** | **33.5** | **30.1** | **32.4** | **21.5** | **22.6** | **32.2** |
| Gong *et al.* [7] | 28.9 | 31.6 | 29.7 | 30.6 | 31.4 | 37.1 | 29.5 | 29.6 | 41.2 | 42.8 | 32.5 | 29.3 | 31.6 | **20.7** | 21.7 | 31.2 |
| Adjusted | **28.3** | **31.0** | **29.0** | **30.0** | **31.0** | **37.0** | **29.3** | **29.3** | **40.4** | **42.5** | **31.9** | **28.9** | **31.1** | **20.7** | **21.6** | **30.8** |
| Shan *et al.* [17] | 30.7 | 32.6 | 29.9 | 31.1 | 31.7 | 37.1 | 30.0 | 29.8 | 40.6 | 42.9 | 33.2 | 30.4 | 31.7 | 21.6 | 22.5 | 31.7 |
| Adjusted | **29.2** | **31.6** | **29.1** | **29.9** | **31.3** | **37.0** | **29.1** | **29.2** | **40.1** | **42.7** | **32.5** | **29.4** | **31.1** | **20.9** | **21.7** | **31.0** |
| Peng *et al.* [16] | 30.7 | 32.7 | 30.1 | 31.2 | 32.1 | 37.4 | 30.2 | 31.3 | 42.1 | 45.5 | 33.7 | 30.6 | 31.7 | 21.5 | 22.7 | 32.2 |
| Adjusted | **29.3** | **31.8** | **29.2** | **30.0** | **31.8** | **36.9** | **29.3** | **30.5** | **41.5** | **45.4** | **33.1** | **29.6** | **31.3** | **20.7** | **21.9** | **31.5** |
| Xu *et al.* [23] | 32.2 | 34.3 | 30.7 | 32.3 | 33.3 | 36.1 | 30.1 | 31.6 | 41.9 | **45.3** | 34.6 | 31.9 | 32.6 | 22.5 | 23.7 | 32.9 |
| Adjusted | **30.8** | **33.2** | **29.8** | **31.3** | **33.2** | **36.2** | **29.5** | **30.9** | **41.3** | **45.3** | **33.8** | **30.8** | **32.1** | **21.8** | **22.9** | **32.2** |



doi:10.6342/NTU202402242

Table 5.5: Reconstruction error on Human3.6M before and after adjustment and fine-tuning with our Bi-GRU model fixed. The top table shows the result under protocol 1. The bottom table shows the result under protocol 2. Best in bold. Unit: millimeter

| Protocol 1 | Dir. | Disc. | Eat | Greet | Phone | Photo | Pose | Purch. | Sit | SitD. | Smoke | Wait | WalkD. | Walk | WalkT. | Avg |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Pavllo *et al*. [15] | 45.2 | 46.7 | 43.3 | 45.6 | 48.1 | 55.1 | 44.6 | 44.3 | 57.3 | 65.8 | 47.1 | 44.0 | 49.0 | 32.8 | 33.9 | 46.8 |
| Adjusted | 41.9 | **45.2** | 42.1 | 44.3 | 46.1 | **53.5** | 43.4 | **42.1** | 55.5 | 64.0 | 45.5 | 42.3 | 47.1 | 32.1 | 32.9 | 45.2 |
| Fine-tuned | **41.5** | **45.2** | **41.9** | **44.0** | **45.9** | 53.6 | **43.3** | 42.3 | **55.3** | **63.8** | **45.3** | **42.2** | **47.0** | **31.8** | **32.4** | **45.0** |

| Protocol 2 | Dir. | Disc. | Eat | Greet | Phone | Photo | Pose | Purch. | Sit | SitD. | Smoke | Wait | WalkD. | Walk | WalkT. | Avg |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Pavllo *et al*. [15] | 34.1 | 36.1 | 34.4 | 37.2 | 36.4 | 42.2 | 34.4 | 33.6 | 45.0 | 52.5 | 37.4 | 33.8 | 37.8 | 25.6 | 27.3 | 36.5 |
| Adjusted | 32.8 | **35.1** | **33.1** | 36.2 | **35.2** | 42.0 | 33.8 | **33.1** | **44.1** | **52.0** | 36.5 | **33.0** | **37.3** | 25.0 | 27.0 | 35.8 |
| Fine-tuned | **32.7** | 35.3 | **33.1** | **35.9** | 35.3 | **41.7** | **33.7** | 33.3 | **44.1** | **52.0** | **36.4** | **33.0** | **37.3** | **24.9** | **26.6** | **35.7** |

models like Chen *et al*. [2], which exhibit smaller bone length errors due to effective bone length prediction, show more modest improvements of around 1% under Protocol 1. Our adjustment effectively rectifies pose errors for all models under Protocol 2, which undergoes rigid alignment like scaling. This indicates that our predicted bone lengths possess better body proportions.

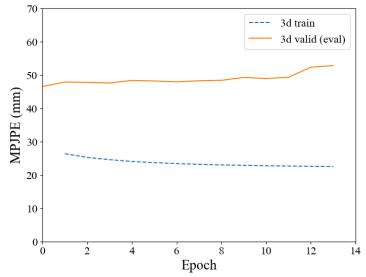
Table 5.4 presents the action-wise comparison. Our adjustment consistently improves all models across all actions. For the actions ”Walk” and ”Walk Together”, we observe a decrease in performance for several models [7, 13, 17, 23, 24]. As discussed in Section 5.1, these models can predict accurate poses for dynamic actions. However, the MPJPE only measures the error in joint positions. If our model can predict bone lengths more accurately than these models, the MPJPE may still be larger due to errors in bone directions. The improvement under protocol 2 (P-MPJPE) supports our assumption and demonstrates that our model achieves better body proportions.

## 5.3 Fine-tuning

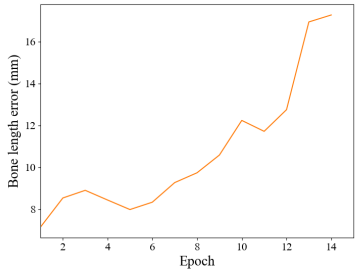
Table 5.5 details the results of our fine-tuning process with the bone length model fixed. We select Pavllo *et al*. [15] as the target lifting model and fine-tune it. We fo-


doi:10.6342/NTU202402242

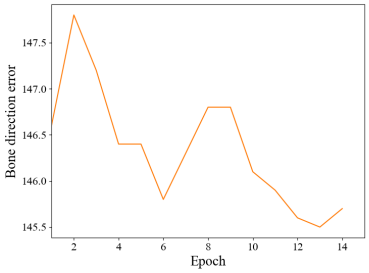
cus on comparing the result after adjustment and the result after fine-tuning. While the lifting model already performs well, fine-tuning demonstrates incremental improvements, particularly noticeable in dynamic actions like ”Walk” (0.3 mm) and ”Walk Together” (0.5 mm). This highlights the effectiveness of leveraging bone length cues to refine the model’s predictions.




(a) The MPJPE on the test set




(b) The bone length error on the test set

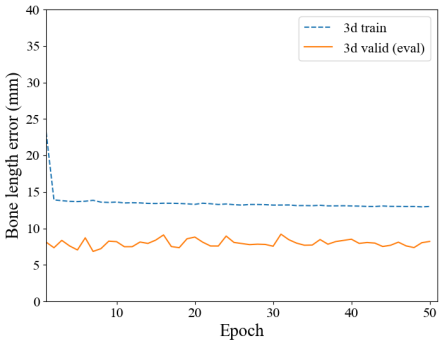



(c) The bone direction error on the test set

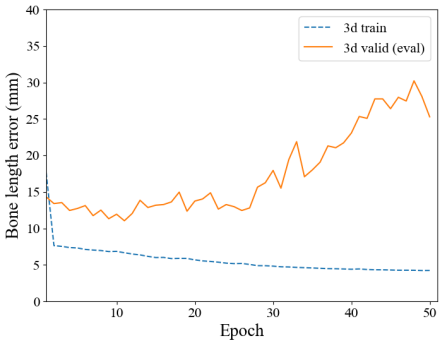
Figure 5.3: Training curves of fine-tuning the entire model using model-predicted keypoints as input.

Figure 5.3 illustrates the training curves for fine-tuning the entire model using model-predicted keypoints. Initially, the model’s MPJPE is 46.5 mm. However, performance deteriorates immediately after a single epoch, with the MPJPE increasing to 47.9 mm, a 3% rise. After a few epochs, overfitting becomes apparent. We observe a rapid gain on the bone length error, while the bone direction error slightly decreases. As discussed in Section 3.4, the bone length model may be affected by erroneous bone directions. Flipping ambiguity is observed in predicted poses in both the training and test sets, leading to the bone length model’s failure.


doi:10.6342/NTU202402242




(a) Trained with data augmentation




(b) Trained without data augmentation

Figure 5.4: Training curves of the bone length model with and without data augmentation. (a) synthetic bone lengths using the mean values in the test set.

Additionally, the bone length model is more likely to overfit since the non-augmented data used for predicting final poses accounts for half of the inputs to the bone length model. As shown in Figure 5.4, the bone length model does not overfit when data augmentation is applied, while it overfits after several epochs when data augmentation is not applied. This indicates the effectiveness of data augmentation in preventing overfitting and improving the robustness of the model.

To sum up, the model overfits when fine-tuning the bone length model due to two main reasons. First, the predicted bone directions misguide the length model. Second, the non-augmented data used for fine-tuning the length model exacerbates this issue.

## 5.4 Inference Speed

In Table 5.6, we evaluate the inference efficiency across three scenarios: (1) the lifting models alone, (2) the lifting models with our adjustment process, and (3) the bone length models.

We measure the frames per second (FPS) for these models during real-time online


doi:10.6342/NTU202402242

Table 5.6: Comparison on Parameters, frame per second (FPS), and MPJPE. The evaluation is performed without test-time augmentation.

| Model | Frames | Parameters (M) | FPS | MPJPE (mm) |
|---|---|---|---|---|
| Pavllo *et al*. [15] | 243 | 16.95 | 958 | 46.8 |
| Chen *et al*. [2] | 243 | 59.18 | 197 | 44.2 |
| Zheng *et al*. [25] | 81 | 9.60 | 379 | 44.3 |
| Pavllo *et al*. [15] (with adjustment) | 243 | 19.34 | 435 | 45.6 |
| Chen *et al*. [2] (with adjustment) | 243 | 61.57 | 154 | 44.0 |
| Zheng *et al*. [25] (with adjustment) | 81 | 11.99 | 252 | 43.3 |
| Chen *et al*. [2] (bone length model) | - | 8.56 | 715 | - |
| Ours, GRU model | - | 2.39 | 2097 | - |

processing, where each model predicts a single frame at a time. The horizontal flip augmentation is not applied in the evaluation. We repeat the inference step 10,000 times, simulating a test on a 10,000-frame video, using a single GeForce GTX 3060 Ti GPU. As our Bi-GRU model is unsuitable for online processing, we test using our GRU model instead. After applying our adjustment process, the MPJPE loss improves significantly with minimal overhead in model size and computation time. Although [15] with adjustment runs at half the FPS compared to without adjustment, it remains faster than the other models listed. Additionally, the complete human pose estimation includes 2D keypoint detection and 2D-to-3D lifting. The FPS of most 2D keypoint detection models is lower than 100. Thus our approach will not be the bottleneck.

For the bone length model, our approach updates bone length values faster than [2]. Our model requires only the input of the new frame at each step, as past information is stored in the hidden state, whereas [2] needs to randomly select 50 frames from previous inputs to predict bone lengths. The FPS is limited by our adjustment process that we decompose poses into bone directions and reconstruct poses with inferred bone lengths.

In summary, our adjustment and fine-tuning methodologies enhance the robustness and accuracy of existing 3D lifting models, demonstrating their efficacy in improving


doi:10.6342/NTU202402242

Table 5.7: Abaltion study on different architecture parameters in the bone length prediction model. Best in bold.

| Model | Layer | Number of units | Bidirectional | Length error (mm) | MPJPE (mm) | P-MPJPE (mm) |
|---|---|---|---|---|---|---|
| GRU | 1 | 1 | ✗ | 7.9 | 46.0 | 36.6 |
| GRU | 2 | 1 | ✗ | 8.0 | 45.9 | 36.6 |
| GRU | 3 | 1 | ✗ | 7.7 | 45.9 | 36.7 |
| GRU | 1 | 2 | ✗ | 7.4 | 45.8 | 36.1 |
| GRU | 1 | 3 | ✗ | 7.5 | 46.3 | 36.3 |
| GRU | 1 | 1 | ✓ | 7.1 | **45.2** | **35.8** |
| GRU | 2 | 1 | ✓ | 7.4 | 45.4 | 35.9 |
| GRU | 3 | 1 | ✓ | 7.6 | 46.1 | 36.3 |
| GRU | 1 | 2 | ✓ | 7.1 | 45.5 | 36.6 |
| GRU | 1 | 3 | ✓ | **6.7** | 45.9 | 36.5 |
| LSTM | 1 | 1 | ✓ | **7.1** | 45.5 | **35.8** |

pose estimation across different evaluation protocols and dynamic scenarios. In real-time online processing, our adjustments achieve competitive results with minimal efficiency overhead.

## 5.5 Ablation Study

We perform extensive ablation experiments on Human3.6M under bone length error, protocol 1 (MPJPE), and protocol 2 (P-MPJPE). We use the prediction of [15] and apply the bone length adjustment on the poses. All the errors are evaluated on these adjusted poses.

We conduct the ablation study on the bone length model, fine-tuning, and inference process. The results are presented in Section 5.5.1, Section 5.5.2, and Section 5.5.3, respectively.

### 5.5.1 Bone Length Model

We compare different model architectures for the bone length prediction model. For data augmentation, we use synthetic bone lengths aligned with the mean values of both


doi:10.6342/NTU202402242

Table 5.8: Abaltion study on the bone length model. MPJPE is applied as a loss function in all cases.

| Bone length model | Bone length loss | Bone direction loss | MPJPE (mm) | P-MPJPE (mm) |
|---|---|---|---|---|
| ✗ | ✗ | ✗ | 46.8 | 36.5 |
| ✗ | ✓ | ✗ | 47.5 | 37.2 |
| ✗ | ✗ | ✓ | 48 | 37.1 |
| ✓ | ✗ | ✓ | **45.0** | **35.7** |

the training set and test set of Human3.6M. We apply GRU and LSTM as the RNN units. The number of layers determines how many layers are in a single unit, while the number of units indicates how many units are in the model. For example, there is one unit with a single layer in Figure 3.3 (a), and there are two units with a single layer in Figure 3.3 (b). The hidden states of the last layer are concatenated and input to the regression head.

The results in Table 5.7 demonstrate that the best performance is achieved using bidirectional GRU units, highlighting the importance of leveraging future information. Although the model with three Bi-GRU units achieves the lowest bone length error, it does not perform well in bone length adjustment. We observe that models with higher complexity tend to have worse performance in bone length adjustment. Notably, the best GRU model has the same size as the best Bi-GRU model (a single Bi-GRU is equivalent to two GRUs). Moreover, the performance is similar between GRU and LSTM. Given the lower computational complexity of GRU, we decided to use GRU instead of LSTM.

### 5.5.2 Fine-tuning

In our work, we fine-tune the lifting model [15] with bone length information. We compare to the result of fine-tuning the same lifting model without the bone length prediction model. For not using the bone length model, we apply the loss function directly to poses predicted by the lifting model. The loss including bone length loss, bone direction loss, and MPJPE.


doi:10.6342/NTU202402242

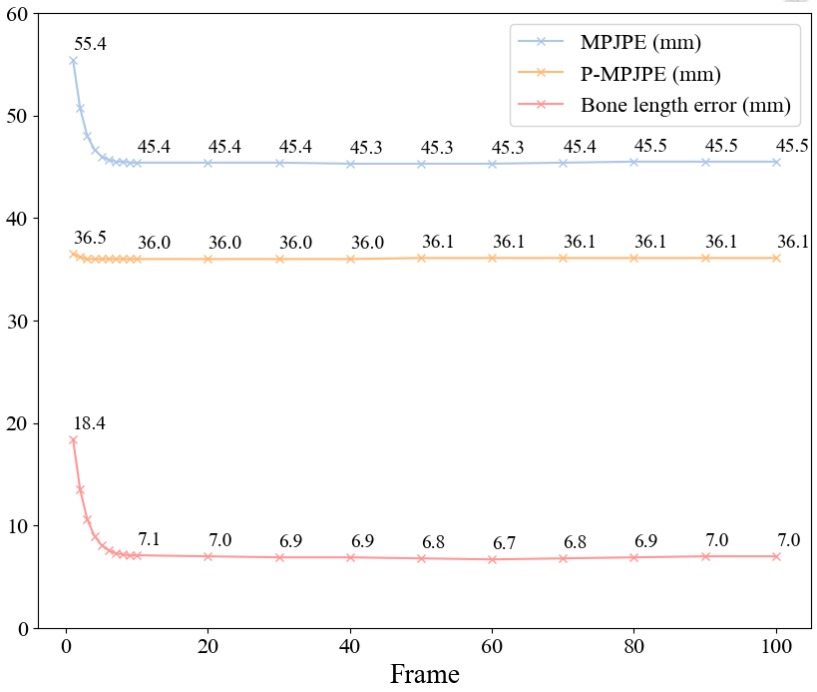


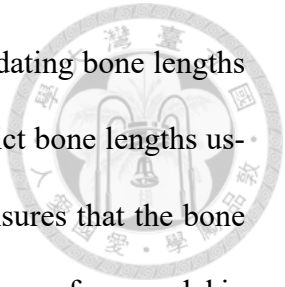

Figure 5.5: Comparison of using different input sequence lengths in our GRU model on the test set of Human3.6M.

The results in Table 5.8 illustrate the impact of incorporating our bone length prediction model. Interestingly, the model performs worse when fine-tuning with either the bone length loss or the bone direction loss. Since no data augmentation is applied, the bone length loss does not contribute much additional information. Regarding the bone direction loss, although the MPJPE is higher compared to using the bone length loss, the P-MPJPE is improved. This discrepancy suggests that the model may prioritize learning bone directions over accurately predicting joint positions. After incorporating our bone length model, we observe a significant improvement in the results. This demonstrates the efficacy of our approach in enhancing 3D human pose estimation.

### 5.5.3 Inference

In real-world applications, human pose estimation is often used for online processing. Since the bone lengths of the subject remain constant throughout the video, continuous up-


doi:10.6342/NTU202402242



dates of bone lengths are unnecessary. Additionally, continuously updating bone lengths can compromise their consistency. Therefore, it is preferable to predict bone lengths using a short sequence and then stop updating them. This approach ensures that the bone lengths remain consistent throughout the video. To evaluate the efficiency of our model in predicting precise bone lengths, we tested it by starting from the beginning of the videos and updating the bone lengths with our GRU model. These predicted bone lengths were then used to adjust the poses predicted by [15]. The results, shown in Figure 5.5, indicate that the errors rapidly converge in fewer than 10 frames. This demonstrates that our model can predict precise bone lengths using a short input sequence, thereby imposing minimal efficiency overhead when applied to existing models.


doi:10.6342/NTU202402242

# Chapter 6 Conclusions and Future Work

In this thesis, we introduced a novel approach for enhancing 3D human pose estimation by integrating precise bone length prediction and adjustment methods. For bone length prediction, we developed a GRU-based model along with a novel data augmentation technique involving synthetic bone lengths.

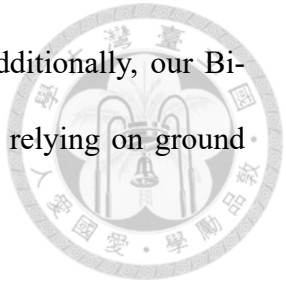
In this section, we conclude our findings in Section 6.1 and outline future work in Section 6.2.

## 6.1 Conclusions

The main observations are listed as follows:

- In Section 5.1, the comparison between different data augmentation approaches shows that the synthetic bone lengths effectively improve the performance of our bone length prediction model.
- The comparison between bone length errors in Section 5.1 reveals that both our GRU model and Bi-GRU model, when using synthetic data for augmentation, achieve


doi:10.6342/NTU202402242



the lowest bone length errors on the Human3.6M dataset. Additionally, our Bi-GRU model attains state-of-the-art performance even without relying on ground truth mean values of bone lengths.

- In Section 5.1, we observe that the Human3.6M dataset has inaccuracies in the joint positions of the wrists, indicating that they are not accurately located.

- For the experiments in Section 5.2, our bone length adjustment technique refines the poses generated by existing 2D-to-3D lifting models, which significantly reduces MPJPE and P-MPJPE errors, particularly in models with higher initial bone length errors. This demonstrates that our model effectively learns better body proportions from synthetic bone lengths.

- The results in Section 5.3 indicate that our fine-tuning process further improves pose estimation accuracy, especially for dynamic actions. However, when we attempt to fine-tune the bone length model simultaneously, the model overfits immediately.

- In the experiments in Section 5.4, our GRU-model achieves high FPS in real-time processing, illustrating the small overhead of our adjustment process.

- The ablation study on inference process in Section 5.5 shows that our bone length prediction model can achieve high precision within 10 frames.

Overall, our approach effectively enhances the anatomical accuracy of 3D human pose predictions, demonstrating significant improvements in error metrics and robustness across various models and activities.


doi:10.6342/NTU202402242

## 6.2 Future Work

We observe several limitations in our work that we are unable to overcome in a short time. Therefore, we will address them as future work.

- To improve the quality of Human3.6M, we can create a new MoCap dataset with a wider variety of bone lengths and a consistent definition of joint positions across different subjects.
- We can fine-tune more lifting models and test our methods on different Mocap datasets to show the generality of our methods.
- We failed in fine-tuning the entire model since flipping ambiguity in bone directions misguide the bone length model. Solving flipping ambiguity not only improves the performance of lifting models, but also provides a chance to fine-tune flipping models and our bone length model simultaneously.


doi:10.6342/NTU202402242



doi:10.6342/NTU202402242

# References


[1] C. S. Catalin Ionescu, Fuxin Li. Latent structured models for human pose estimation. In *International Conference on Computer Vision*, 2011.

[2] T. Chen, C. Fang, X. Shen, Y. Zhu, Z. Chen, and J. Luo. Anatomy-aware 3d human pose estimation with bone-based pose decomposition. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(1):198–209, 2021.

[3] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7103–7112, 2018.

[4] H. Choi, G. Moon, and K. M. Lee. Pose2mesh: Graph convolutional network for 3d human pose and mesh recovery from a 2d human pose. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 769–787. Springer, 2020.

[5] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[6] Z. Geng, K. Sun, B. Xiao, Z. Zhang, and J. Wang. Bottom-up human pose estimation




doi:10.6342/NTU202402242

via disentangled keypoint regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14676–14686, 2021.

[7] J. Gong, L. G. Foo, Z. Fan, Q. Ke, H. Rahmani, and J. Liu. Diffpose: Toward more reliable 3d pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13041–13051, 2023.

[8] M. R. I. Hossain and J. J. Little. Exploiting temporal information for 3d human pose estimation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 68–84, 2018.

[9] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, jul 2014.

[10] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7122–7131, 2018.

[11] M. Kocabas, N. Athanasiou, and M. J. Black. Vibe: Video inference for human body pose and shape estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5253–5263, 2020.

[12] K. Lee, I. Lee, and S. Lee. Propagating lstm: 3d pose estimation based on joint interdependency. In *Proceedings of the European conference on computer vision (ECCV)*, pages 119–135, 2018.

[13] W. Li, H. Liu, H. Tang, P. Wang, and L. Van Gool. Mhformer: Multi-hypothesis



doi:10.6342/NTU202402242

transformer for 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13147–13156, 2022.

[14] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, Oct. 2015.

[15] D. Pavllo, C. Feichtenhofer, D. Grangier, and M. Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7753–7762, 2019.

[16] J. Peng, Y. Zhou, and P. Mok. Ktpformer: Kinematics and trajectory prior knowledge-enhanced transformer for 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1123–1132, 2024.

[17] W. Shan, Z. Liu, X. Zhang, Z. Wang, K. Han, S. Wang, S. Ma, and W. Gao. Diffusion-based 3d human pose estimation with multi-hypothesis aggregation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14761–14771, 2023.

[18] C. Sminchisescu and B. Triggs. Kinematic jump processes for monocular 3d human tracking. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 1, pages I–I. IEEE, 2003.

[19] K. Sun, B. Xiao, D. Liu, and J. Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5693–5703, 2019.



doi:10.6342/NTU202402242

[20] X. Sun, J. Shang, S. Liang, and Y. Wei. Compositional human pose regression. In *Proceedings of the IEEE international conference on computer vision*, pages 2602–2611, 2017.

[21] M. Trotter and G. C. Gleser. Estimation of stature from long bones of american whites and negroes. *American journal of physical anthropology*, 10(4):463–514, 1952.

[22] W.-L. Wei, J.-C. Lin, T.-L. Liu, and H.-Y. M. Liao. Capturing humans in motion: Temporal-attentive 3d human pose and shape estimation from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13211–13220, 2022.

[23] J. Xu, Y. Guo, and Y. Peng. Finepose: Fine-grained prompt-driven 3d human pose estimation via diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 561–570, 2024.

[24] J. Zhang, Z. Tu, J. Yang, Y. Chen, and J. Yuan. Mixste: Seq2seq mixed spatio-temporal encoder for 3d human pose estimation in video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13232–13242, 2022.

[25] C. Zheng, S. Zhu, M. Mendieta, T. Yang, C. Chen, and Z. Ding. 3d human pose estimation with spatial and temporal transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11656–11665, 2021.



doi:10.6342/NTU202402242