# HumMUSS: Human Motion Understanding using State Space Models

Arnab Mondal
Mila & Apple
arnab.mondal@mila.quebec

Stefano Alletto
Apple
salletto@apple.com

Denis Tome
Apple
d_tome@apple.com

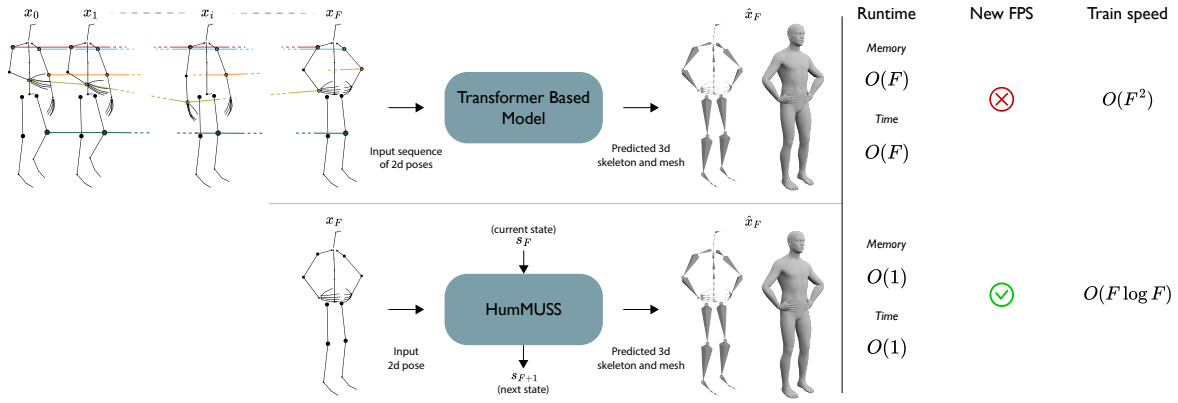


Figure 1. HumMUSS vs. Transformer-Based Models for sequential prediction of 3D poses and human meshes from 2D keypoint videos. **Top**: Transformer-based models attend to a history of 2D poses/keypoints to predict the current frame's output. **Bottom**: HumMUSS, being a stateful model, efficiently utilizes only the current frame and its current state for predictions, ensuring constant memory and time complexity. HumMUSS also generalizes to new frame rates and enhances the training speed without compromising the prediction accuracy.

## Abstract

*Understanding human motion from video is essential for a range of applications, including pose estimation, mesh recovery and action recognition. While state-of-the-art methods predominantly rely on transformer-based architectures, these approaches have limitations in practical scenarios. Transformers are slower when sequentially predicting on a continuous stream of frames in real-time, and do not generalize to new frame rates. In light of these constraints, we propose a novel attention-free spatiotemporal model for human motion understanding building upon recent advancements in state space models.*

*Our model not only matches the performance of transformer-based models in various motion understanding tasks but also brings added benefits like adaptability to different video frame rates and enhanced training speed when working with longer sequences of keypoints. Moreover, the proposed model supports both offline and real-time applications. For real-time sequential prediction, our model is both memory efficient and several times faster than transformer-based approaches while maintaining their high accuracy.*

## 1. Introduction

Understanding human motion is crucial for various computer vision applications, including body keypoints tracking [16, 32, 84, 109], human mesh estimation [13, 102] or action recognition [19, 107]. While recent advancements have enabled a plethora of real-world applications focusing on real-time requirements and mobile deployment [5, 98], this field is dominated by Transformer-based architectures [46, 98, 109]. Although these models attain remarkable accuracy, they face practical challenges: processing long video sequences with Transformers can be slow, and they do not generalize well to unseen frame rates. Moreover, Transformers are highly inefficient in terms of memory and speed for real-time sequential prediction [54, 69, 83] compared to state-based sequence models.

In this work, we propose to look beyond attention-based architectures for learning motion representations and leverage recent advancements in State Space Models (SSMs) [25, 26, 28, 62, 91] to address the practical limitations of Transformers. In particular, we propose a novel attention-free spatiotemporal architecture, named HumMUSS, which utilizes SSMs to learn human motion representations. HumMUSS consists of several stacked blocks, each containing

two streams of alternating spatial and temporal Gated Diagonal SSM blocks (GDSSM), designed to efficiently learn rich spatiotemporal features. HumMUSS inherits the advantages of DSSM, such as faster training and inference for longer sequences and O(1) time and memory complexity for real-time sequential inference. Moreover, it also can generalize to unseen and variable frame rates due to SSMs' inherent continuous time formulation. This is especially relevant for any real-time on-device applications where load and thermal conditions can drastically change the camera's sampling rate.

HumMUSS achieves performance on par with state-of-the-art methods on several motion understanding tasks, while providing all the aforementioned practical benefits. Additionally, we introduce a fully causal version of HumMUSS, designed to predict each current time-step using only past and current frames, with no foresight into future frames. In this causal setting, we demonstrate that HumMUSS not only outperforms current state-of-the-art models but is also several times faster and memory efficient. This is essential for real-time applications where low latency is critical.

Our contributions can be summarized as follows

- We introduce HumMUSS, a novel attention-free spatiotemporal architecture using SSMs, to process human motion. To the best of our knowledge, this is the first attempt to bring SSM-based architectures to human motion understanding tasks.
- We demonstrate HumMUSS improves both training and inference speed compared to state of the art methods such as MotionBERT [109] for long motion videos. Additionally, being a state-based model, it makes sequential prediction several times faster, for example $3.8\times$ for a context length of $\approx 243$ frames.
- We show that HumMUSS, being a continuous-time model, seamlessly generalizes to dynamic frame rates during inference with minimal performance degradation.
- We empirically demonstrate that our pre-trained HumMUSS achieves competitive accuracy for various tasks such as 3D pose estimation, human mesh recovery and action recognition, proving its viability as a superior alternative to transformers for a broad range of motion understanding tasks.

## 2. Related Work

**3D Pose Estimation** Various approaches for Human 3D Pose Estimation are currently implemented using either a monocular (single-view) or a multi-view configuration. In the multi-view scenarios [16, 32, 71, 72, 86, 104], multiple camera perspectives are utilized to enhance the accuracy of the extracted poses by exploiting geometric information from the camera placement. Both configurations can be further classified into two broader categories: *i)* direct approaches, which predict 3d joints directly from input frames, and *ii)* two-step solutions, which use 2d joint positions as a basis to estimate the 3d poses.

Direct 3D estimation techniques [65, 66, 80, 108] determine the spatial positions of 3D joints from video frames without intermediary stages, enabling real-time application with reduced computational overhead. In comparison to that, other 2D-3D lifting techniques have adopted an initial step of utilizing a readily available precise off-the-shelf 2D pose estimators [64, 79, 92], and then lift the 2D coordinates into 3D [47, 59, 68, 84, 85, 103, 105, 106, 106, 109].

**Human Mesh Recovery** Given a frame or video input, the problem of 3D human mesh recovery falls into two main categories: parametric and non-parametric methods. Parametric methods learn to estimate the parameters of a human body model, like SMPL [56], while non-parametric approaches directly regress the 3D coordinates of human mesh vertices. Parametric models [6, 20, 27, 33, 34, 39, 41, 67, 82, 102] utilizing the inherent body structure encoded in models such as SMPL [56], estimate shape and pose parameters, which often lead to more anatomically accurate results. However, as the 3d position of vertices are generated from shape and pose parameters, such methods can sometimes limit the representation of complex and diverse body types and movements. In contrast, non-parametric models [13, 42, 50, 51, 63] capture more nuanced details and variations in human forms and postures, offering adaptability to a wider range of complex data. However, they may yield results that lack interpretability due to the absence of explicit parameters, which can obscure the rationale behind certain predictions.

**Skeleton-based Action Recognition** Skeleton-based action recognition focuses on identifying the spatial and temporal dynamics between human body joints to classify the action being performed in a video. Recent methods approach this challenge by employing multi-task learning or using multimodal inputs [1, 57], yet the predominant strategy involves learning from pre-established 2D or 3D poses [10, 18, 19, 22, 77, 107]. Among these methods, some of the most relevant approaches rely on one three main paradigms: 3D CNNs, GCNs or transformers. [18] utilizes 3D CNNs alongside 3D volumetric heatmaps to learn a representation that is robust to noise in the input pose. [107] proposes an additional feature refinement phase to mitigate some intrinsic limitations of graph-based pose modeling with GCNs. Lastly, [109] leverages a Transformer, pre-trained on a pretext 3D pose estimation task, to classify actions from 2D pose data.

**Transformers for Human Modeling** The Transformer architecture [87], initially developed for language model-

ing, has demonstrated significant advancements in various computer vision tasks. Particularly in human understanding tasks, the Transformer has shown notable improvements in the accuracy of human pose and shape estimation [39, 50, 75, 89, 100]. Several approaches utilize the attention mechanism to effectively capture spatial relationship of the joints [39, 50, 51] and their temporal dynamics [89, 100], or a combination of both using spatiotemporal attention based architectures [47, 48, 103, 105, 109]. These methods indicate the versatility and effectiveness of Transformers in modeling complex human bodies and their motion data.

**State Space Models** Recent studies [25, 26, 28, 62] have explored State Space Models (SSMs) as an effective alternative to Transformers for modeling long sequences. In particular, the use of a diagonal state matrix has been crucial in enhancing the training speed of these models, thus making them faster than Transformers, especially for processing longer sequences. Moreover, certain initializations of diagonal values [25] are capable of replicating convolution kernels that are designed for long-range memory. This aligns with the principles of the HiPPO theory [23] and explains the comparable performance of Diagonal SSMs to Structured State Space Models (S4) [26] as indicated by [28].

**Multiplicative Gating** Various types of neural architectures, including Multi-Layer Perceptrons (MLPs), CNNs, and Transformers have benefited from the integration of gating units [17, 74]. One popular form of these gating units is the Gated Linear Unit (GLU), which has proven particularly effective in CNN-based Natural Language Processing (NLP) applications [17]. Recent studies emphasize the utility of gating units for simplifying network routing and suggests that multiplicative gating can act as a surrogate to recapture some of the intricate interactions found in attention-based mechanisms. For instance, Hua *et al*. [30] demonstrate that linear-time attention models can achieve better performance with the use of optimized gating mechanisms. Similarly, [62] and [91] show that incorporating multiplicative gating mechanism enhances SSMs for NLP tasks.

# 3. Background

## 3.1. State Space Models

State space models (SSM) provide a continuous time learnable framework for mapping between continuous-time scalar inputs and outputs. Given an input $u(t)$ and output $y(t)$, SSM is described by differential equations involving a continuous-time state vector $x(t)$ and its derivative $x'(t)$, parameterized by matrices $A \in \mathbb{R}^{N\times N}$, $B \in \mathbb{R}^{N\times 1}$, $C \in \mathbb{R}^{1\times N}$ and $D \in \mathbb{R}$.

$$x'(t) = Ax(t) + Bu(t), \quad y(t) = Cx(t) + Du(t).$$

In discrete-time, with a parameterized sample time $\Delta$, these equations transition into recursive formulations

$$x_k = \bar{A}x_{k-1} + \bar{B}u_k, \quad y_k = \bar{C}x_k + \bar{D}u_k. \tag{1}$$

where $\bar{A} = e^{A\Delta}$, $\bar{B} = (e^{A\Delta} - I)A^{-1}B$, $\bar{C} = C$ and $\bar{D} = D$ using zero order hold (zoh) discretization [31].

The linear nature of SSMs allows the output sequence to be computed directly by unrolling the recursion in time

$$y_k = \sum_{j=0}^{k} \bar{C}\bar{A}^j\bar{B} \cdot u_{k-j}.$$

A significant advantage of this structure is the potential for parallel computation, facilitated by the discrete convolution of the input sequence $u$ with the precomputed SSM kernel $K = (\bar{C}\bar{B}, \bar{C}\bar{A}\bar{B}, \ldots, \bar{C}\bar{A}^{L-1}\bar{B})$, denoted by $y = K *_c u$. While the naive approach to this computation requires $O(L^2)$ multiplications, it can be done in $O(L\log(L))$ time using the Fast Fourier Transform (FFT) [7]. SSMs can conveniently switch from their convolutional to recursive formulation in Eq. (1) when properties like auto-regressive decoding are desirable.

## 3.2. Diagonal State Spaces

An efficient adaptation of the SSM framework is the incorporation of a diagonal state matrix, greatly facilitating the computation of the SSM kernel $K$ [24, 28]. As shown by Gu *et al*. [25], a diagonal state matrix $A$ represented as $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_N)$ can approximate the HiPPO parameterization [23] of the transition matrix A that yields stable training regime with long sequences. Further simplifications are introduced with the vector $B$ being expressed as $B = (1)_{N\times 1}$. Under these conditions, the DSSM model is characterized by learnable parameters $\Lambda_{re}, \Lambda_{im} \in \mathbb{R}^N$, $C \in \mathbb{C}^N$, and $\Delta_{\log} \in \mathbb{R}$. The diagonal elements of $A$ are then computed through the relationship $-\exp(\Lambda_{re}) + i \cdot \Lambda_{im}$, where $i = \sqrt{-1}$ and $\Delta$ is deduced as $\exp(\Delta_{log}) \in \mathbb{R}^{>0}$. The kernel $K$ can be computed as

$$K = (C \odot \begin{bmatrix} \left(e^{\lambda_1\Delta} - 1\right)/\lambda_1 \\ \vdots \\ \left(e^{\lambda_N\Delta} - 1\right)/\lambda_N \end{bmatrix})^T \cdot \exp(P) \tag{2}$$

where $\odot$ is element-wise multiplication and the elements of matrix $P \in \mathbb{C}^{N\times L}$ are defined as $P_{j,k} = \lambda_j k\Delta$. In practice, to get a real valued kernel K, the diagonal elements are assumed to appear in complex conjugate pairs and their corresponding parameters in $C$ are tied together. Hence, the

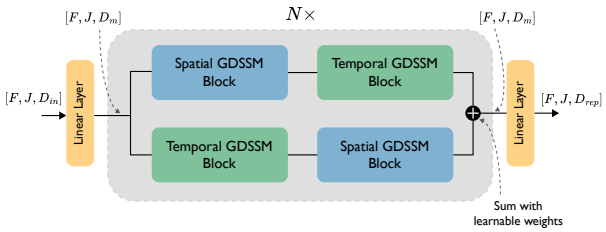


Figure 2. HumMUSS model architecture

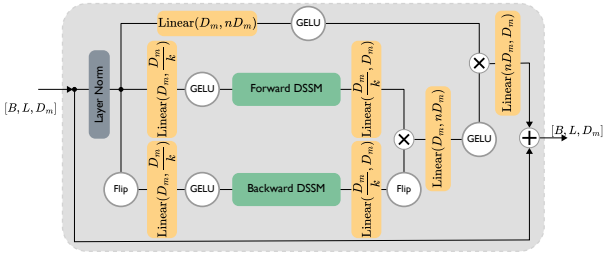


Figure 3. Bidirectional Gated DSSM Block

dimension of state space is effectively set to $N/2$ and the final kernel is obtained by taking the real part of $2K$. We provide more details on the initialization of DSSM parameters in the supplementary material.

This framework establishes a linear mapping for 1-D sequences. When extending to sequences comprising $H$-dimensional vectors, individual state space models are applied to each of the $H$ dimensions. Specifically, a DSSM layer takes a sequence of length $L$, denoted as $\mathbf{u} \in \mathbb{R}^{H\times L}$, and yields an output $\mathbf{y} \in \mathbb{R}^{H\times L}$. For each feature dimension $h = [1, \ldots, H]$, a kernel $K_h \in \mathbb{R}^L$ is computed. The corresponding output $y_h \in \mathbb{R}^L$ for this feature is obtained using the convolution of input $u_h \in \mathbb{R}^L$ and kernel $K_h$. This can be done for a batch of samples leading to a linear DSSM layer that can map from $\mathbf{u} \in \mathbb{R}^{B\times L\times H}$ to $\mathbf{y} \in \mathbb{R}^{B\times L\times H}$ and is denoted by $y = \text{DSSM}(u)$[1]. Considering a batch size of $B$, sequence length $L$, and hidden dimension $H$, the computation time for the kernels in the DSS layer scales as $O(NHL)$, whereas the discrete convolution demands a time complexity of $O(BHL\log(L))$.

## 4. Method

HumMUSS is a general purpose attention-free architecture that takes spatiotemporal human motion data as input and outputs representations corresponding to each spatial and temporal location. Let the input to HumMUSS be a video of joints $u \in \mathbb{R}^{B\times F\times J\times D_{in}}$ where $B$ is the batch size, $F$ is the number of frames, $J$ is the number of joints and $D_{in}$ is the dimension of the input which is typically 3 for the 2D joint positions and scalar joint confidence. HumMUSS learns to model the underlying continuous signal resulting from evolution of joint positions and their interactions with each other to produce a spatiotemporal representation $r \in \mathbb{R}^{B\times F\times J\times D_{rep}}$. As shown in Figure 2, HumMUSS features a sequence of spatiotemporal blocks, each incorporating two alternating streams of Spatial and Temporal Gated Diagonal State Space Model (GDSSM) blocks. It uses a lifting layer to transform the input to the model dimension $D_m$ and a final layer to transform the output embeddings in $D_m$ to required representation dimension $D_{rep}$.

[1]The transpose operation of the last two dimension of the input and output is moved inside the DSSM layer.

In the subsequent sections, we first provide our general architecture of a bidirectional and uni-direction (causal) GDSSM blocks, and then show how to use them to build a spatiotemporal layer.

### 4.1. Bidirectional GDSSM Block

Given an input $x \in \mathbb{R}^{B\times L\times D_m}$ where $B$ is the batch size, $L$ is the length of the sequence, and $D_m$ is the model dimension, the Bidirectional GDSSM Blocks learns to aggregate information across the sequence dimension $L$. As illustrated in Figure 3, this block starts with a layer normalization and has three distinct pathways to process information. The first pathway processes information independently whereas the other two combines it forward and backward in the sequence dimension

$$
\begin{aligned}
x_{id} &= \sigma(\text{LayerNorm}(x)W_{id}) && \in \mathbb{R}^{B\times L\times nD_m} \\
x_f &= \text{DSSM}_f(\sigma(x_N W_f^1))W_f^2 && \in \mathbb{R}^{B\times L\times D_m} \\
x_b &= flip(\text{DSSM}_b(\sigma(flip(x_N)W_b^1))W_b^2) && \in \mathbb{R}^{B\times L\times D_m}
\end{aligned}
$$

with $W_{id} \in \mathbb{R}^{D_m\times nD_m}$, $W_f^1 - W_b^1 \in \mathbb{R}^{D_m\times \frac{D_m}{k}}$, $W_f^2 - W_b^2 \in \mathbb{R}^{\frac{D_m}{k}\times D_m}$ the learnable weight matrices; $flip(\cdot)$ denotes flipping operation along the sequence dimension and $\sigma(\cdot)$ denote GELU activation [29]. In this formulation, we reduce the dimension of the DSSM by a factor of $k$ to speed up kernel computation and combine different dimensions of the DSSM output by using weights $W_f^2, W_b^2$. Following [62] and [91], we combine the forward and backward aggregated information using multiplicative gating

$$x_{cb} = \sigma((x_f \odot x_b)W_{cb}) \quad \in \mathbb{R}^{B\times L\times nD_m}$$

where $W_{cb} \in \mathbb{R}^{D_m\times nD_m}$ and $\odot$ denotes a Hadamard Product. Finally, the block's output is computed by combining the independently processed information from the first pathway with the outputs of the other two pathways, and then adding a skip connection with the block's input. We use a dimension expansion factor of $n$ before using the multiplicative gating

$$x_{out} = x + (x_{cb} \odot x_{id})W_{out} \quad \in \mathbb{R}^{B\times L\times D_m}$$

where $W_{out} \in \mathbb{R}^{nD_m\times D_m}$ is used to bring the output of this multiplicative gate back to the model dimension.

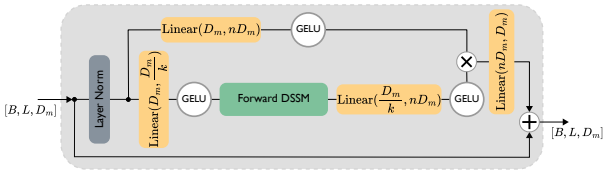


Figure 4. Unidirectional Gated DSSM Block

This provides an expressive non-linear bidirectional block to process a sequence of vectors and we denote it as $x_{out} = \text{BiGDSSM-Block}(x)$.

### 4.2. Unidirectional GDSSM Block

Similarly, for an input $x \in \mathbb{R}^{B\times L\times D_m}$, the unidirectional GDSSM Blocks learns to combine information along the sequence dimension $L$ but only in forward direction. As shown in Figure 4, this block also starts with layer normalization and has two main pathways to process information. The first pathway processes information independently whereas the other one combines it forward in the sequence dimension

$$\begin{aligned} x_{id} &= \sigma(\text{LayerNorm}(x)W_{id}) && \in \mathbb{R}^{B\times L\times nD_m} \\ x_f &= \text{DSSM}_f(\sigma(x_N W_f^1))W_f^2 && \in \mathbb{R}^{B\times L\times D_m} \end{aligned}$$

where $W_{id} \in \mathbb{R}^{D_m\times nD_m}$, $W_f^1 \in \mathbb{R}^{D_m\times \frac{D_m}{k}}$ and $W_f^2 \in \mathbb{R}^{\frac{D_m}{k}\times nD_m}$. In contrast to the bidirectional block, the output of the unidirectional block is directly computed by combining $x_{id}$ and $x_f$ using multiplicative gating and a skip connection with the input to the block

$$x_{out} = x + (x_f \odot x_{id})W_{out} \quad \in \mathbb{R}^{B\times L\times D_m}$$

where $W_{out} \in \mathbb{R}^{nD_m\times D_m}$. We denote this causal block as $x_{out} = \text{UniGDSSM-Block}(x)$.

### 4.3. Building a spatiotemporal Layer

In this section, we construct a spatiotemporal layer utilizing the GDSSM Blocks discussed previously. Following [109], given input $x \in \mathbb{R}^{B\times F\times J\times D_m}$, we pass it through two different information processing streams. By adopting this approach, each stream captures distinct spatio-temporal aspects, thereby enhancing the overall expressivity of the model. The first stream combines information spatially and then temporally

$$\begin{aligned} x_s &= \text{BiGDSSM-Block}_s^1(x.\text{flatten}(0,1)) \\ x_s &= x_s.\text{reshape}(B,F,J,D_m).\text{T}(1,2) \\ x_{ts} &= \text{BiGDSSM-Block}_t^1(x_s.\text{flatten}(0,1)) \\ x_{ts} &= x_{ts}.\text{reshape}(B,J,F,D_m).\text{T}(1,2) \end{aligned}$$

where $\text{BiGDSSMBlock}_s^1(\cdot)$-$\text{BiGDSSM-Block}_t^1(\cdot)$ are the spatial-temporal GDSSM Blocks of stream 1, $x.\text{T}(a,b)$ denotes the transpose of $x$ and $x.\text{flatten}(a,b)$ the flattening operation of the $a$-th and $b$-th dimension of the tensor. The transpose, reshape and flattening operations are necessary to process the spatial and temporal dimension using the similar GDSSM blocks which expects a tensor of shape $B \times L \times D_m$.

The second stream aggregates information temporally and then spatially

$$\begin{aligned} x_t &= \text{BiGDSSM-Block}_t^2(x.\text{T}(1,2).\text{flatten}(0,1)) \\ x_t &= x_t.\text{reshape}(B,J,F,D_m) \\ x_{st} &= \text{BiGDSSM-Block}_s^2(x_t.\text{T}(1,2).\text{flatten}(0,1)) \\ x_{st} &= x_{st}.\text{reshape}(B,F,J,D_m) \end{aligned}$$

Finally, we combine the outputs of both the streams using learnable weights given by

$$\begin{aligned} \begin{bmatrix}\alpha_{st} & \alpha_{ts}\end{bmatrix} &= \text{softmax}(\begin{bmatrix}x_{st} & x_{ts}\end{bmatrix}\mathcal{W}) && \in \mathbb{R}^{B\times F\times J\times 2} \\ x_{out} &= \alpha_{st}\odot x_{st} + \alpha_{ts}\odot x_{ts} && \in \mathbb{R}^{B\times F\times J\times D_m} \end{aligned}$$

where $\mathcal{W} \in \mathbb{R}^{2D_m\times 2}$ is a learnable mapping to the weights which are normalized by using softmax$(\cdot)$.

**Causal model.** To design a causal variant of the spatiotemporal layer, we replace the temporal blocks in both the streams with an unidirectional GDSSM block as proposed in Sec. 4.2. In particular, we replace $\text{BiGDSSM-Block}_t^1(\cdot)$ $\text{BiGDSSM-Block}_t^2(\cdot)$ with $\text{UniGDSSM-Block}_t^1(\cdot)$ and $\text{UniGDSSM-Block}_t^2(\cdot)$.

### 4.4. Pretraining HumMUSS

To demonstrate HumMUSS ability to learn generic motion features, we pretrain our model on a pretext task, and then finetune it on downstream tasks that require human motion understanding. To ensure a fair comparison, we adopt the same pretraining strategy and datasets as MotionBERT [109], which has achieved state-of-the-art results in various motion understanding tasks. We also pretrain causal variants of HumMUSS and MotionBERT [109]. Causal variant of MotionBERT is implemented by employing a causal mask to establish a robust baseline in the causal setup. In all our experiments, we use 16M[2] parameters for both models unless specified otherwise (see supplementary material for implementation details).

First, we aim to learn a robust motion representation using a universal pretext task. We employ a "cloze" task, akin to recovering depth information from 2D visual observations, inspired by 3D human pose estimation. We use large-scale 3D motion capture data such as the AMASS

[2] [109] shows that a smaller 16M parameter version achieves comparable performance to the original 44M parameter model while being computationally cheap.

dataset [58] to create a 2D-to-3D lifting task, where we generate corrupted 2D skeleton sequences from 2D projections of 3D motion. These sequences mimic real-world issues like occlusions and errors. Then, HumMUSS outputs motion representation which can be used to reconstruct 3D motion using a final linear layer. With the reconstructed and ground-truth (GT) 3D motion represented as $\hat{\mathbf{x}}$ and $\mathbf{x}$ respectively, the total loss in 3D space is given by

$$\mathcal{L}_{3D} = \sum_{t=1}^{F}\sum_{j=1}^{J} \|\hat{\mathbf{x}}_{t,j} - \mathbf{x}_{t,j}\|^2 + \lambda \|\hat{\mathbf{v}}_{t,j} - \mathbf{v}_{t,j}\|^2 \quad (3)$$

where $\hat{\mathbf{v}}_{t,j} = \hat{\mathbf{x}}_{t,j} - \hat{\mathbf{x}}_{t-1,j}$ and $\mathbf{v}_{t,j} = \mathbf{x}_{t,j} - \mathbf{x}_{t-1,j}$ are 3D velocities, and $\lambda$ is the weight of the velocity loss. For this pretraining task, we rely on two datasets with 3D GT motion trajectories: MPI-INF-3DHP [61] and AMASS [58].

Second, to utilize heterogeneous human motion data in various formats, we extract 2D skeletons from different motion data sources using in-the-wild RGB video datasets such as PoseTrack [2] and InstaVariety [36]. Given the dataset RGB videos, we either use the provided manually labelled 2D GT skeletons, or follow [109] in computing them using the publicly available 2D pose estimation network from [64]. Since GT 3D motion is not available for this data, we use a weighted 2D re-projection loss. The final pre-training loss in the 2D image space is computed as

$$\mathcal{L}_{2D} = \sum_{t=1}^{F}\sum_{j=1}^{J} \delta_{i,j} \|\hat{x}_{t,j} - x_{t,j}\|^2 \quad (4)$$

with $\hat{x}$ the 2D orthographic projection of the predicted 3D motion $\hat{\mathbf{x}}$ and $\delta$ the 2D joint detection confidence.

To degrade the 2D skeletons, we apply random zero masking to $15\%$ of the joints and sample noises from a combination of Gaussian and uniform distributions [8]. We use curriculum learning to pretrain HumMUSS for 90 epochs where $\mathcal{L}_{3D}$ is minimized for the first 30 epochs and $\mathcal{L}_{2D}$ is minimized in the remaining 60 epochs.

# 5. Experiments

In this section, we evaluate pre-trained HumMUSS using different downstream tasks. Following [109], we choose 3D pose estimation, human mesh recovery and action recognition as our downstream tasks. After adding necessary heads on top of the HumMUSS backbone to produce outputs specific to each task, we finetune the entire model and showcase its competitive performance against current state-of-the-art models. Finally, we demonstrate the benefits of using HumMUSS over Transformer-based methods in Sec 5.4 and provide additional experiments in the supplementary material.

| Method | $F$ | Params | MPJPE↓ | PCK↑ | AUC↑ |
|---|---|---|---|---|---|
| Causal | | | | | |
| Mehta et al. [61] | 1 | - | 117.6 | 75.7 | 39.3 |
| †MotionBERT (s) [109] | 243 | 16M | 25.4 | 97.9 | 85.2 |
| †MotionBERT (f) [109] | 243 | 16M | 21.1 | **99.0** | **86.8** |
| HumMUSS (s) | 243 | 16M | 24.6 | 98.2 | 85.6 |
| HumMUSS (f) | 243 | 16M | **21.0** | 98.7 | 86.1 |
| Bidirectional | | | | | |
| UGCN [90] | 96 | - | 68.1 | 86.9 | 62.1 |
| PoseFormer [106] | 81 | - | 77.1 | 88.6 | 56.4 |
| MHFormer [47] | 9 | 30.9M | 58.0 | 93.8 | 63.3 |
| MixSTE [103] | 27 | 33.6M | 54.9 | 94.4 | 66.5 |
| Einfalt *et al.* [21] | 81 | 10.4M | 46.9 | 95.4 | 97.6 |
| P-STMO [73] | 81 | 6.2M | 32.2 | 97.9 | 75.8 |
| HDFormer [9] | 96 | 3.7M | 37.2 | 98.7 | 72.9 |
| HSTFormer [70] | 81 | 22.7M | 41.4 | 97.3 | 71.5 |
| STCFormer[84] | 81 | 4.7M | 23.1 | 98.7 | 83.9 |
| PoseFormerV2[105] | 81 | 14.3M | 27.8 | 97.9 | 78.8 |
| GLA-GCN [99] | 81 | 1.3M | 27.7 | 79.1 | 98.5 |
| MotionAGFormer [60] | 81 | 19M | 16.2 | 98.2 | 85.3 |
| † MotionBERT (s) [109] | 243 | 16M | 18.2 | 99.1 | 88.0 |
| † MotionBERT (f) [109] | 243 | 16M | **16.0** | **99.3** | **89.9** |
| HumMUSS (s) | 243 | 16M | 18.7 | 99.0 | 87.1 |
| HumMUSS (f) | 243 | 16M | 16.3 | 99.2 | 89.2 |

Table 1. **Human 3D Pose Estimation** Comparison on the MPI-INF-3DHP dataset. MPJPE (in $mm$) from detected 2D poses are reported. $F$ and Params denote the context/clip length and the number of parameters used by the method respectively. † indicates results obtained from finetuning official implementation of MotionBERT [109]. (s) indicates models trained from sctach, (f) models finetuned after pre-training.

## 5.1. 3D Pose Estimation

Since HumMUSS learns to predict 3D poses during pretraining, we directly finetune it using the final linear layer from the pretraining phase on the MPI-INF-3DHP dataset [61]. Following previous work, we use the ground truth 2D skeletons from the videos in the dataset. In Table 1, we report the mean per joint position error (MPJPE) in millimeters (mm), Percentage of Correct Keypoint (PCK) within 150 mm range, and Area Under the Curve (AUC) as evaluation metric on the MPI-INF-3DHP dataset [61]. We categorize the methods based on their causal or non-causal nature. Causal methods are more applicable to real-time scenarios where future frame information is unavailable. For a strong baseline we also finetune and train both causal and non-causal variant of MotionBERT on MPI-INF-3DHP. We observe that HumMUSS consistently outperforms existing methods in the causal setup and competes favorably with state-of-the-art results in the bidirectional setup.

## 5.2. Mesh Recovery

We perform experiments on 3DPW [88] datasets. Following prior work [103, 106, 109], we augment the training set with the COCO [52] dataset. In Table 2, we report the performance of our finetuned model using MPJPE (mm), PA-MPJPE (mm) and MPVE (mm) metrics. Additionally, re-

| Method | Input | $F$ | MPVE↓ | MPJPE↓ | PA-MPJPE↓ |
|---|---|---|---|---|---|
| Causal | | | | | |
| HMR [35] | image | 1 | - | 130.0 | 81.3 |
| † SPIN [40] | image | 1 | 129.1 | 100.9 | 59.1 |
| Pose2Mesh [13] | 2D pose | 1 | 109.3 | 91.4 | 60.1 |
| I2L-MeshNet [63] | image | 1 | 110.1 | 93.2 | 58.6 |
| † HybrIK [44] | image | 1 | 82.4 | 71.3 | 41.9 |
| METRO [50] | image | 1 | 88.2 | 77.1 | 47.9 |
| Mesh Graphormer[47] | image | 1 | 87.7 | 74.7 | 45.6 |
| PARE [39] | image | 1 | 88.6 | 74.5 | 46.5 |
| ROMP [82] | image | 1 | 108.3 | 91.3 | 54.9 |
| PyMAF [102] | image | 1 | 110.1 | 92.8 | 58.9 |
| ProHMR [43] | image | 1 | - | - | 59.8 |
| OCHMR [37] | image | 1 | 107.1 | 89.7 | 58.3 |
| 3DCrowdNet [15] | image | 1 | 98.3 | 81.7 | 51.5 |
| CLIFF [49] | image | 1 | 81.2 | 69.0 | 43.0 |
| FastMETRO [12] | image | 1 | 84.1 | 73.5 | 44.6 |
| VisDB [97] | image | 1 | 85.5 | 73.5 | 44.9 |
| MotionBERT (f) [109] | 2D motion | 16 | 93.5 | 82.3 | 50.9 |
| MotionBERT (f) + [44] | video | 16 | 80.9 | 70.1 | **41.3** |
| HumMUSS (f) | 2D motion | 16 | 93.4 | 82.0 | 50.2 |
| HumMUSS (f) + [44] | video | 16 | **80.5** | **69.8** | **41.3** |
| Bidirectional | | | | | |
| TemporalContext[3] | video | 32 | - | - | 72.2 |
| HMMR [36] | video | 20 | 139.3 | 116.5 | 72.6 |
| DSD-SATN[81] | video | 9 | - | - | 69.5 |
| VIBE[38] | video | 16 | 99.1 | 82.9 | 51.9 |
| TCMR [14] | video | 16 | 102.9 | 86.5 | 52.7 |
| † MAED [89] | video | 16 | 93.3 | 79.0 | 45.7 |
| MPS-Net [93] | video | 16 | 99.7 | 84.3 | 52.1 |
| † PoseBERT [4] (+[40]) | video | 16 | - | - | 57.3 |
| † SmoothNet [101] (+[40]) | video | 32 | - | 86.7 | 52.7 |
| † MotionBERT (f) [109] | 2D motion | 16 | 88.1 | 76.9 | 47.2 |
| † MotionBERT (f) + [44] | video | 16 | **79.4** | **68.8** | **40.6** |
| HumMUSS (f) | 2D motion | 16 | 88.9 | 77.4 | 47.5 |
| HumMUSS (f) + [44] | video | 16 | 80.0 | 69.1 | 40.7 |

Table 2. **Human mesh recovery** Quantitative comparison on 3DPW dataset. Input and $F$ correspond to the input type and context length used by the method. † denotes that the results are taken from [109]. (s) indicates models trained from sctach, (f) models finetuned after pre-training.

sults of finetuning both the causal HumMUSS and MotionBERT for the Mesh Recovery task are also provided. Our model is competitive with existing approaches. However, as highlighted by [109], recovering full-body mesh solely from sparse 2D keypoints is inherently challenging due to the lack of human shape information. Therefore, we also provide results with a hybrid approach introduced by [109] that use an MLP to combine pretrained motion representations, and an initial prediction provided by RGB-based method HybrIK [44] to refine joint rotations. We notice that the hybrid approach significantly improves performance in both the causal and bidirectional setup making HumMUSS competitive with existing approaches.

### 5.3. Skeleton-based Action Recognition

For this task, following [109], we perform global average pooling across motion representations of different persons and timesteps. The outcome is subsequently input into an MLP with one hidden layer. The network is trained using cross-entropy classification loss. In Table 3, we present a comparison between HumMUSS and recent SOTA methods for action recognition on the NTU-RGBD dataset [53]. Following [109], we finetune pretrained HumMUSS on the training set of NTU-RGBD. We observe that HumMUSS performs favorably compared to recent methods and is only outperformed by SkeleTR [19], the most recent approach to action recognition. It is worth noting that while SkeleTR [19] employs an extremely task specific architecture, HumMUSS is designed to serve as a generic motion understanding backbone that can be applied to several other motion-related tasks. Enhancing performance through modifications in the fine-tuning architecture is left as future work.

| Method | X-Sub↑ | X-View↑ |
|---|---|---|
| ST-GCN [94] | 81.5 | 88.3 |
| 2s-AGCN [76] | 88.5 | 95.1 |
| MS-G3D [55] | 91.5 | 96.2 |
| Shift-GCN [11] | 90.7 | 96.5 |
| CrosSCLR [45] | 86.2 | 92.5 |
| MCC (finetune) [78] | 89.7 | 96.3 |
| SCC (finetune) [96] | 88.0 | 94.9 |
| UNIK (finetune) [95] | 86.8 | 94.4 |
| CTR-GCN [10] | 92.4 | 96.8 |
| PoseConv3D [18] | 93.1 | 95.7 |
| † MotionBERT (finetune) [109] | 93.0 | 97.2 |
| UPS [22] | 92.6 | 97.0 |
| SkeleTR [C] [19] | **94.8** | **97.7** |
| HumMUSS (finetune) | 92.0 | 97.4 |

Table 3. **Action Recognition** Quantitative comparison on the NTU-RGB+D dataset. Left and right column report the top-1 accuracy for the cross-subject and cross-view split respectively. † indicates results taken from [109] using the 44M parameter version of MotionBERT.

### 5.4. HumMUSS vs Transformer-based methods

HumMUSS offers several advantages over existing SOTA models that depend on transformer-based architectures. In this section, we delve into the benefits of HumMUSS over one such method, MotionBERT [109].[3]

**Robust to new frame rate.** In contrast to transformer-based architectures, HumMUSS, being a continuous-time model, demonstrates the ability to generalize to unseen frame rates with minimal drops in accuracy. This adaptability can be achieved by appropriately adjusting the discretization parameter $\Delta$ within the model using the new frame rate. Figures 5 and 6 present both qualitative and quantitative comparison between HumMUSS and MotionBERT [109] trained on the MPI-INF-3DHP dataset [61] and evaluated on sub-sampled motion videos from the test set. The findings reveal that transformer-based methods like MotionBERT's performance declines significantly as the sub-sampling rate increases compared to HumMUSS. The capability to perform reliably under various sampling rates is valuable in real-world scenarios, where input frame rates might fluctuate due to factors such as thermal throttling of the capturing devices. We provide a more detailed discussion in the supplementary material.

[3] We use same number of parameters (16M) for both the models.

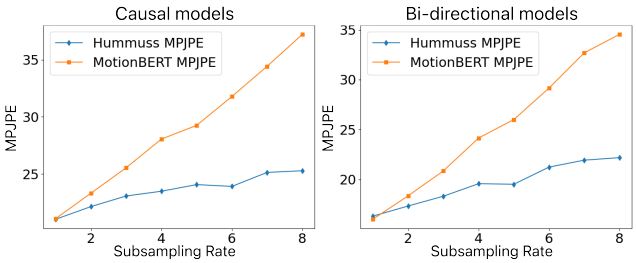


Figure 5. Comparison between HumMUSS and MotionBERT [109] 3D pose estimation performance (MPJPE in $mm$) on MPI-INF-3DHP at different sub-sampling rates. *Left*: causal models; *Right*: bi-directional models.

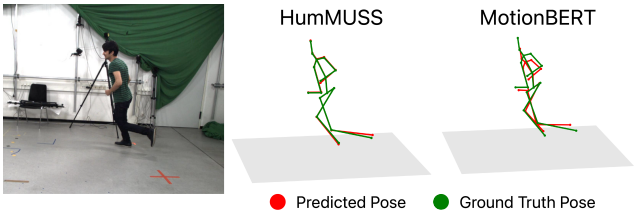


Figure 6. Example of reconstructed 3D poses between HumMUSS and MotionBERT [109] when input signal is sub-sampled at rate 8 (using one frame every 8). Ground-truth represented in green, predictions in red. Best viewed in color.

**Training Speed and convergence** The training speed of HumMUSS scales favorably due to its $O(F\log(F))$ complexity, in contrast to the $O(F^2)$ complexity of attention-based architectures, where $F$ represents the number of frames. As a result, HumMUSS is significantly faster than MotionBERT for longer sequence lengths. Training with longer context length can be crucial for certain human motion understanding tasks including action recognition, gesture analysis and Gait Analysis. Furthermore, HumMUSS demonstrates significantly faster convergence compared to MotionBERT during the training phase both for the bidirectional and causal model, as illustrated in Fig. 7. Exploring the underlying reasons for these training dynamics presents an interesting avenue for future research.

**Efficient Sequential Inference.** One of the remarkable advantages of HumMUSS lies in its efficiency for real-time sequential inference, making it a drop in replacement for transformer-based models in various high-accuracy real-time applications. HumMUSS operates as a stateful recurrent model during sequential inference, requiring only the current frame and the state that summarizes the past frames. This substantially boosts the inference speed and efficiency of HumMUSS relative to MotionBERT [109][4]. As depicted in Fig. 8, we observe HumMUSS is 3.8× memory efficient and an 11.1× faster during sequential inference on 243 frames. Moreover, the constant memory usage and inference speed of HumMUSS result in increasingly significant improvements in GPU memory and latency for longer frame sequences.

[4]We used the official implementation of MotionBERT. Adding causal transformer inference speedup tricks like KV-caching will improve MotionBERT's speed and memory.

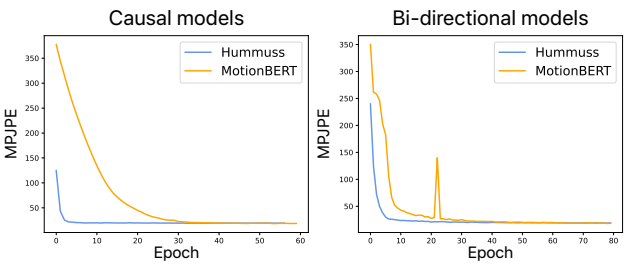


Figure 7. Training dynamics of HumMUSS and MotionBERT [109] for 3D pose estimation on MPI-INF-3DHP. We plot the validation performance (MPJPE in $mm$) over epochs. *Left*: causal models; *Right*: bi-directional models.

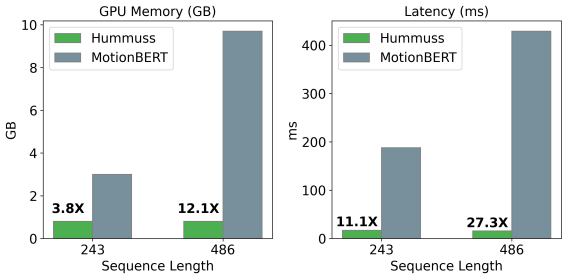


Figure 8. GPU memory usage and Latency comparison between MotionBERT and HumMUSS for sequential inference using a causal model. Both the models are tested on a 80GB A100 GPU with a batch-size of 32. GPU memory utilized is reported in GB and average inference latency is reported in ms for every frame.

## 6. Conclusion

In this work, we present HumMUSS, a novel attention-free architecture designed specifically for human motion understanding. HumMUSS leverages diagonal state space models, effectively overcoming some of the major limitations inherent in current state-of-the-art transformer-based models, notably their slow sequential inference and limited robustness when faced with various frame rates. Our extensive experiments highlight HumMUSS' versatility in a range of motion understanding tasks, proving its effectiveness in 3D pose estimation, mesh estimation, and action recognition. We are confident that HumMUSS can serve as a powerful motion understanding backbone, especially in applications that require real-time sequential inference. This approach marks a significant step forward, potentially leapfrogging existing methods and narrowing the divide between highly accurate transformer-based techniques and their practical application in the real world.

## References


[1] Dasom Ahn, Sangwon Kim, Hyunsu Hong, and Byoung Chul Ko. Star-transformer: a spatio-temporal cross attention transformer for human action recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3330–3339, 2023. 2

[2] Mykhaylo Andriluka, Umar Iqbal, Eldar Insafutdinov, Leonid Pishchulin, Anton Milan, Juergen Gall, and Bernt Schiele. Posetrack: A benchmark for human pose estimation and tracking. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5167–5176, 2018. 6

[3] Anurag Arnab, Carl Doersch, and Andrew Zisserman. Exploiting temporal context for 3d human pose estimation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3395–3404, 2019. 7

[4] Fabien Baradel, Romain Brégier, Thibault Groueix, Philippe Weinzaepfel, Yannis Kalantidis, and Grégory Rogez. Posebert: A generic transformer module for temporal 3d human modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 7

[5] Valentin Bazarevsky, Ivan Grishchenko, Karthik Raveendran, Tyler Zhu, Fan Zhang, and Matthias Grundmann. Blazepose: On-device real-time body pose tracking. *arXiv preprint arXiv:2006.10204*, 2020. 1

[6] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*, pages 561–578. Springer, 2016. 2

[7] E Oran Brigham. *The fast Fourier transform and its applications*. Prentice-Hall, Inc., 1988. 3

[8] Ju Yong Chang, Gyeongsik Moon, and Kyoung Mu Lee. Poselifter: Absolute 3d human pose lifting network from a single noisy 2d human pose. *arXiv preprint arXiv:1910.12029*, 2019. 6

[9] Hanyuan Chen, Jun-Yan He, Wangmeng Xiang, Zhi-Qi Cheng, Wei Liu, Hanbing Liu, Bin Luo, Yifeng Geng, and Xuansong Xie. Hdformer: High-order directed transformer for 3d human pose estimation. *arXiv preprint arXiv:2302.01825*, 2023. 6

[10] Yuxin Chen, Ziqi Zhang, Chunfeng Yuan, Bing Li, Ying Deng, and Weiming Hu. Channel-wise topology refinement graph convolution for skeleton-based action recognition. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 13359–13368, 2021. 2, 7

[11] Ke Cheng, Yifan Zhang, Xiangyu He, Weihan Chen, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with shift graph convolutional network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 183–192, 2020. 7

[12] Junhyeong Cho, Kim Youwang, and Tae-Hyun Oh. Cross-attention of disentangled modalities for 3d human mesh recovery with transformers. In *European Conference on Computer Vision*, pages 342–359. Springer, 2022. 7

[13] Hongsuk Choi, Gyeongsik Moon, and Kyoung Mu 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. 1, 2, 7

[14] Hongsuk Choi, Gyeongsik Moon, Ju Yong Chang, and Kyoung Mu Lee. Beyond static features for temporally consistent 3d human pose and shape from a video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1964–1973, 2021. 7

[15] Hongsuk Choi, Gyeongsik Moon, JoonKyu Park, and Kyoung Mu Lee. Learning to estimate robust 3d human mesh from in-the-wild crowded scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1475–1484, 2022. 7

[16] Sungho Chun, Sungbum Park, and Ju Yong Chang. Learnable human mesh triangulation for 3d human pose and shape estimation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2850–2859, 2023. 1, 2

[17] Yann N Dauphin, Angela Fan, Michael Auli, and David Grangier. Language modeling with gated convolutional networks. In *International conference on machine learning*, pages 933–941. PMLR, 2017. 3

[18] Haodong Duan, Yue Zhao, Kai Chen, Dahua Lin, and Bo Dai. Revisiting skeleton-based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2969–2978, 2022. 2, 7

[19] Haodong Duan, Mingze Xu, Bing Shuai, Davide Modolo, Zhuowen Tu, Joseph Tighe, and Alessandro Bergamo. Skeletr: Towards skeleton-based action recognition in the wild. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13634–13644, 2023. 1, 2, 7

[20] Sai Kumar Dwivedi, Nikos Athanasiou, Muhammed Kocabas, and Michael J Black. Learning to regress bodies from images using differentiable semantic rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11250–11259, 2021. 2

[21] Moritz Einfalt, Katja Ludwig, and Rainer Lienhart. Uplift and upsample: Efficient 3d human pose estimation with uplifting transformers. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2903–2913, 2023. 6

[22] Lin Geng Foo, Tianjiao Li, Hossein Rahmani, Qiuhong Ke, and Jun Liu. Unified pose sequence modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13019–13030, 2023. 2, 7

[23] Albert Gu, Tri Dao, Stefano Ermon, Atri Rudra, and Christopher Ré. Hippo: Recurrent memory with optimal polynomial projections. *Advances in neural information processing systems*, 33:1474–1487, 2020. 3

[24] Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. *arXiv preprint arXiv:2111.00396*, 2021. 3

[25] Albert Gu, Karan Goel, Ankit Gupta, and Christopher Ré. On the parameterization and initialization of diagonal state space models. *Advances in Neural Information Processing Systems*, 35:35971–35983, 2022. 1, 3

[26] Albert Gu, Karan Goel, and Christopher Re. Efficiently modeling long sequences with structured state spaces. In *International Conference on Learning Representations*, 2022. 1, 3

[27] Peng Guan, Alexander Weiss, Alexandru O Balan, and Michael J Black. Estimating human shape and pose from a single image. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1381–1388. IEEE, 2009. 2

[28] Ankit Gupta, Albert Gu, and Jonathan Berant. Diagonal state spaces are as effective as structured state spaces. *Advances in Neural Information Processing Systems*, 35: 22982–22994, 2022. 1, 3

[29] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 4

[30] Weizhe Hua, Zihang Dai, Hanxiao Liu, and Quoc Le. Transformer quality in linear time. In *International Conference on Machine Learning*, pages 9099–9117. PMLR, 2022. 3

[31] Arieh Iserles. *A first course in the numerical analysis of differential equations*. Number 44. Cambridge university press, 2009. 3

[32] Karim Iskakov, Egor Burkov, Victor Lempitsky, and Yury Malkov. Learnable triangulation of human pose. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7718–7727, 2019. 1, 2

[33] Wen Jiang, Nikos Kolotouros, Georgios Pavlakos, Xiaowei Zhou, and Kostas Daniilidis. Coherent reconstruction of multiple humans from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5579–5588, 2020. 2

[34] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[35] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra 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. 7

[36] Angjoo Kanazawa, Jason Y Zhang, Panna Felsen, and Jitendra Malik. Learning 3d human dynamics from video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5614–5623, 2019. 6, 7

[37] Rawal Khirodkar, Shashank Tripathi, and Kris Kitani. Occluded human mesh recovery. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1715–1725, 2022. 7

[38] Muhammed Kocabas, Nikos Athanasiou, and Michael 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. 7

[39] Muhammed Kocabas, Chun-Hao P Huang, Otmar Hilliges, and Michael J Black. Pare: Part attention regressor for 3d human body estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11127–11137, 2021. 2, 3, 7

[40] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2252–2261, 2019. 7

[41] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *ICCV*, 2019. 2

[42] Nikos Kolotouros, Georgios Pavlakos, and Kostas Daniilidis. Convolutional mesh regression for single-image human shape reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4501–4510, 2019. 2

[43] Nikos Kolotouros, Georgios Pavlakos, Dinesh Jayaraman, and Kostas Daniilidis. Probabilistic modeling for human mesh recovery. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11605–11614, 2021. 7

[44] Jiefeng Li, Chao Xu, Zhicun Chen, Siyuan Bian, Lixin Yang, and Cewu Lu. Hybrik: A hybrid analytical-neural inverse kinematics solution for 3d human pose and shape estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3383–3393, 2021. 7

[45] Linguo Li, Minsi Wang, Bingbing Ni, Hang Wang, Jiancheng Yang, and Wenjun Zhang. 3d human action representation learning via cross-view consistency pursuit. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4741–4750, 2021. 7

[46] Tianjiao Li, Lin Geng Foo, Ping Hu, Xindi Shang, Hossein Rahmani, Zehuan Yuan, and Jun Liu. Token boosting for robust self-supervised visual transformer pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24027–24038, 2023. 1

[47] Wenhao Li, Hong Liu, Hao Tang, Pichao Wang, and Luc Van Gool. Mhformer: Multi-hypothesis transformer for 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13147–13156, 2022. 2, 3, 6, 7

[48] Yanjie Li, Shoukui Zhang, Zhicheng Wang, Sen Yang, Wankou Yang, Shu-Tao Xia, and Erjin Zhou. Tokenpose: Learning keypoint tokens for human pose estimation. In *Proceedings of the IEEE/CVF International conference on computer vision*, pages 11313–11322, 2021. 3

[49] Zhihao Li, Jianzhuang Liu, Zhensong Zhang, Songcen Xu, and Youliang Yan. Cliff: Carrying location information in full frames into human pose and shape estimation. In *European Conference on Computer Vision*, pages 590–606. Springer, 2022. 7

[50] Kevin Lin, Lijuan Wang, and Zicheng Liu. End-to-end human pose and mesh reconstruction with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1954–1963, 2021. 2, 3, 7

[51] Kevin Lin, Lijuan Wang, and Zicheng Liu. Mesh graphormer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12939–12948, 2021. 2, 3

[52] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 6

[53] Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C Kot. Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. *IEEE transactions on pattern analysis and machine intelligence*, 42(10):2684–2701, 2019. 7

[54] Xinyu Liu, Houwen Peng, Ningxin Zheng, Yuqing Yang, Han Hu, and Yixuan Yuan. Efficientvit: Memory efficient vision transformer with cascaded group attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14420–14430, 2023. 1

[55] Ziyu Liu, Hongwen Zhang, Zhenghao Chen, Zhiyong Wang, and Wanli Ouyang. Disentangling and unifying graph convolutions for skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 143–152, 2020. 7

[56] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 851–866. 2023. 2

[57] Diogo C Luvizon, David Picard, and Hedi Tabia. 2d/3d pose estimation and action recognition using multitask deep learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5137–5146, 2018. 2

[58] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *International Conference on Computer Vision*, pages 5442–5451, 2019. 6

[59] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *Proceedings of the IEEE international conference on computer vision*, pages 2640–2649, 2017. 2

[60] Soroush Mehraban, Vida Adeli, and Babak Taati. Motionagformer: Enhancing 3d human pose estimation with a transformer-gcnformer network. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6920–6930, 2024. 6

[61] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *2017 international conference on 3D vision (3DV)*, pages 506–516. IEEE, 2017. 6, 7

[62] Harsh Mehta, Ankit Gupta, Ashok Cutkosky, and Behnam Neyshabur. Long range language modeling via gated state spaces. *arXiv preprint arXiv:2206.13947*, 2022. 1, 3, 4

[63] Gyeongsik Moon and Kyoung Mu Lee. I2l-meshnet: Image-to-lixel prediction network for accurate 3d human pose and mesh estimation from a single rgb image. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 752–768. Springer, 2020. 2, 7

[64] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14*, pages 483–499. Springer, 2016. 2, 6

[65] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3d human pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7025–7034, 2017. 2

[66] Georgios Pavlakos, Xiaowei Zhou, and Kostas Daniilidis. Ordinal depth supervision for 3d human pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7307–7316, 2018. 2

[67] Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3d human pose and shape from a single color image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 459–468, 2018. 2

[68] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael 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. 2

[69] Reiner Pope, Sholto Douglas, Aakanksha Chowdhery, Jacob Devlin, James Bradbury, Jonathan Heek, Kefan Xiao, Shivani Agrawal, and Jeff Dean. Efficiently scaling transformer inference. *Proceedings of Machine Learning and Systems*, 5, 2023. 1

[70] Xiaoye Qian, Youbao Tang, Ning Zhang, Mei Han, Jing Xiao, Ming-Chun Huang, and Ruei-Sung Lin. Hstformer: Hierarchical spatial-temporal transformers for 3d human pose estimation. *arXiv preprint arXiv:2301.07322*, 2023. 6

[71] N Dinesh Reddy, Laurent Guigues, Leonid Pishchulin, Jayan Eledath, and Srinivasa G Narasimhan. Tessetrack: End-to-end learnable multi-person articulated 3d pose tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15190–15200, 2021. 2

[72] Edoardo Remelli, Shangchen Han, Sina Honari, Pascal Fua, and Robert Wang. Lightweight multi-view 3d pose estimation through camera-disentangled representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6040–6049, 2020. 2

[73] Wenkang Shan, Zhenhua Liu, Xinfeng Zhang, Shanshe Wang, Siwei Ma, and Wen Gao. P-stmo: Pre-trained spa-

tial temporal many-to-one model for 3d human pose estimation. In *European Conference on Computer Vision*, pages 461–478. Springer, 2022. 6

[74] Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020. 3

[75] Dahu Shi, Xing Wei, Liangqi Li, Ye Ren, and Wenming Tan. End-to-end multi-person pose estimation with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11069–11078, 2022. 3

[76] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Two-stream adaptive graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12026–12035, 2019. 7

[77] Sijie Song, Cuiling Lan, Junliang Xing, Wenjun Zeng, and Jiaying Liu. An end-to-end spatio-temporal attention model for human action recognition from skeleton data. In *Proceedings of the AAAI conference on artificial intelligence*, 2017. 2

[78] Yukun Su, Guosheng Lin, and Qingyao Wu. Self-supervised 3d skeleton action representation learning with motion consistency and continuity. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 13328–13338, 2021. 7

[79] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5693–5703, 2019. 2

[80] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *Proceedings of the European conference on computer vision (ECCV)*, pages 529–545, 2018. 2

[81] Yu Sun, Yun Ye, Wu Liu, Wenpeng Gao, Yili Fu, and Tao Mei. Human mesh recovery from monocular images via a skeleton-disentangled representation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5349–5358, 2019. 7

[82] Yu Sun, Qian Bao, Wu Liu, Yili Fu, Michael J Black, and Tao Mei. Monocular, one-stage, regression of multiple 3d people. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11179–11188, 2021. 2, 7

[83] Yutao Sun, Li Dong, Shaohan Huang, Shuming Ma, Yuqing Xia, Jilong Xue, Jianyong Wang, and Furu Wei. Retentive network: A successor to transformer for large language models. *arXiv preprint arXiv:2307.08621*, 2023. 1

[84] Zhenhua Tang, Zhaofan Qiu, Yanbin Hao, Richang Hong, and Ting Yao. 3d human pose estimation with spatio-temporal criss-cross attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4790–4799, 2023. 1, 2, 6

[85] Denis Tome, Chris Russell, and Lourdes Agapito. Lifting from the deep: Convolutional 3d pose estimation from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2500–2509, 2017. 2

[86] Denis Tome, Matteo Toso, Lourdes Agapito, and Chris Russell. Rethinking pose in 3d: Multi-stage refinement and recovery for markerless motion capture. In *2018 international conference on 3D vision (3DV)*, pages 474–483. IEEE, 2018. 2

[87] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2

[88] Timo Von Marcard, Roberto Henschel, Michael J Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *Proceedings of the European conference on computer vision (ECCV)*, pages 601–617, 2018. 6

[89] Ziniu Wan, Zhengjia Li, Maoqing Tian, Jianbo Liu, Shuai Yi, and Hongsheng Li. Encoder-decoder with multi-level attention for 3d human shape and pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13033–13042, 2021. 3, 7

[90] Jingbo Wang, Sijie Yan, Yuanjun Xiong, and Dahua Lin. Motion guided 3d pose estimation from videos. In *European Conference on Computer Vision*, pages 764–780. Springer, 2020. 6

[91] Junxiong Wang, Jing Nathan Yan, Albert Gu, and Alexander M Rush. Pretraining without attention. *arXiv preprint arXiv:2212.10544*, 2022. 1, 3, 4

[92] Z Wang, Y Peng, Z Zhang G Yu, J Sun, et al. Cascaded pyramid network for multi-person pose estimation. 2018. 2

[93] Wen-Li Wei, Jen-Chun Lin, Tyng-Luh Liu, and Hong-Yuan Mark 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. 7

[94] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI conference on artificial intelligence*, 2018. 7

[95] Di Yang, Yaohui Wang, Antitza Dantcheva, Lorenzo Garattoni, Gianpiero Francesca, and Francois Bremond. Unik: A unified framework for real-world skeleton-based action recognition. *BMVC*, 2021. 7

[96] Siyuan Yang, Jun Liu, Shijian Lu, Meng Hwa Er, and Alex C Kot. Skeleton cloud colorization for unsupervised 3d action representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13423–13433, 2021. 7

[97] Chun-Han Yao, Jimei Yang, Duygu Ceylan, Yi Zhou, Yang Zhou, and Ming-Hsuan Yang. Learning visibility for robust dense human body estimation. In *European Conference on Computer Vision*, pages 412–428. Springer, 2022. 7

[98] Ziyu Yao, Xuxin Cheng, and Yuexian Zou. Poserac: Pose saliency transformer for repetitive action counting. *arXiv preprint arXiv:2303.08450*, 2023. 1

[99] Bruce XB Yu, Zhi Zhang, Yongxu Liu, Sheng-hua Zhong, Yan Liu, and Chang Wen Chen. Gla-gcn: Global-local adaptive graph convolutional network for 3d human pose

estimation from monocular video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8818–8829, 2023. 6

[100] Ye Yuan, Umar Iqbal, Pavlo Molchanov, Kris Kitani, and Jan Kautz. Glamr: Global occlusion-aware human mesh recovery with dynamic cameras. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11038–11049, 2022. 3

[101] Ailing Zeng, Lei Yang, Xuan Ju, Jiefeng Li, Jianyi Wang, and Qiang Xu. Smoothnet: A plug-and-play network for refining human poses in videos. In *European Conference on Computer Vision*, pages 625–642. Springer, 2022. 7

[102] Hongwen Zhang, Yating Tian, Xinchi Zhou, Wanli Ouyang, Yebin Liu, Limin Wang, and Zhenan Sun. Pymaf: 3d human pose and shape regression with pyramidal mesh alignment feedback loop. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11446–11456, 2021. 1, 2, 7

[103] Jinlu Zhang, Zhigang Tu, Jianyu Yang, Yujin Chen, and Junsong 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. 2, 3, 6

[104] Zhe Zhang, Chunyu Wang, Weichao Qiu, Wenhu Qin, and Wenjun Zeng. Adafuse: Adaptive multiview fusion for accurate human pose estimation in the wild. *International Journal of Computer Vision*, 129:703–718, 2021. 2

[105] Qitao Zhao, Ce Zheng, Mengyuan Liu, Pichao Wang, and Chen Chen. Poseformerv2: Exploring frequency domain for efficient and robust 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8877–8886, 2023. 2, 3, 6

[106] Ce Zheng, Sijie Zhu, Matias Mendieta, Taojiannan Yang, Chen Chen, and Zhengming 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. 2, 6

[107] Huanyu Zhou, Qingjie Liu, and Yunhong Wang. Learning discriminative representations for skeleton based action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10608–10617, 2023. 1, 2

[108] Kun Zhou, Xiaoguang Han, Nianjuan Jiang, Kui Jia, and Jiangbo Lu. Hemlets pose: Learning part-centric heatmap triplets for accurate 3d human pose estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2344–2353, 2019. 2

[109] Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. Motionbert: A unified perspective on learning human motion representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15085–15099, 2023. 1, 2, 3, 5, 6, 7, 8