Video2mesh

3D human pose and shape recovery by a temporal convolutional transformer network
Chao, Xianjin; Ge, Zhipeng; Leung, Howard

Published in:
IET Computer Vision

Published: 01/06/2023

Document Version:
Final Published version, also known as Publisher's PDF, Publisher's Final version or Version of Record

License:
CC BY

Publication record in CityU Scholars:
Go to record

Published version (DOI):
10.1049/cvi2.12172

Publication details:
Video2mesh: 3D human pose and shape recovery by a temporal convolutional transformer network

Xianjin Chao\textsuperscript{1} | Zhipeng Ge\textsuperscript{2} | Howard Leung\textsuperscript{1}

\textsuperscript{1}City University of Hong Kong, Hong Kong, China
\textsuperscript{2}Nanjing University, Nanjing, China

\textbf{Abstract}
From a 2D video of a person in action, human mesh recovery aims to infer the 3D human pose and shape frame by frame. Despite progress on video-based human pose and shape estimation, it is still challenging to guarantee high accuracy and smoothness simultaneously. To tackle this problem, we propose a Video2mesh, a temporal convolutional transformer (TConvTransformer) based temporal network which is able to recover accurate and smooth human mesh from 2D video. The temporal convolution block achieves the sequence-level smoothness by aggregating image features from adjacent frames. The subsequent multi-attention transformer improves the accuracy due to its multi-subspace for better middle-frame feature representation. Meanwhile, we add a TConvTransformer discriminator which is trained together with our 3D human mesh temporal encoder. This TConvTransformer discriminator further improves the accuracy and smoothness by restricting the pose and shape in a more reliable space based on the AMASS dataset. We conduct extensive experiments on three standard benchmark datasets and show that our proposed Video2mesh outperforms other state-of-the-art methods in both accuracy and smoothness.

\textbf{KEYWORDS}
image motion analysis, motion estimation, pose estimation, shape measurement, video signal processing

\textbf{1 | INTRODUCTION}

Human mesh recovery aims to lift the 3D human pose and shape parameters from the RGB images, which is an ongoing research problem in the field of computer vision and computer graphics. Due to the rich and flexible body information brought by the parametric 3D mesh model compared with keypoint representation, there existed promising applications in human part and foreground segmentation [1], human motion and behaviour analysis, action manipulation, augmented reality (AR) and virtual reality (VR) [2]. The traditional approach is based on optimisation [3–5], which fitted the given parametric model SMPL [6] to the 2D joint locations iteratively. Unfortunately, the optimisation approach is very slow and very dependent on initial values.

Following successful efforts in deep learning, there is a great deal of deep learning work on the 3D human mesh recovery from a single image, which are mainly divided into multi-stage methods and end-to-end single-stage methods. For multi-stage methods [7, 8], 2D joint locations [9] or semantic segmentations need to be estimated from a single RGB image through the first stage. However, this method is limited by the accuracy of the first stage and cannot achieve end-to-end. Single-stage method are therefore proposed. For example, HMR [1] directly estimated the single-frame SMPL shape and pose parameters using the features extracted from a single 2D RGB image by ResNet50 [10]. SPIN [11] combined the advantages of the above traditional methods and deep learning methods, which adopts the HMR-based deep neural network to predict the SMPL parameters as the initial value of iterative fitting and then align the model with the 2D key points in a traditional way.

Using only a single frame results in limited information so it is desirable to use more frames. However, even if the
estimation is performed frame by frame, the result may be a discontinuous sequence if the correlation between frames is not considered to ensure smoothness. With explosive growth of video data, the dynamic learning of human motion between adjacent frames has become important clues for realistic human pose and shape recovery and behaviour understanding [12–14]. Human pose and shape recovered from 2D space to 3D space in sequences dramatically increase the complexity of feature expression. Multiple approaches have focussed on improving the accuracy and stability of 3D human mesh recovery from video in recent years.

Temporal HMR [15] learnt the human motion kinematics by feeding each frame's image feature from Resnet50 to feed-forward convolutional networks [16, 17]. Unfortunately, due to the lack of supervision of 3D video data at that time, the result was regarded as over smooth with weak accuracy. The subsequent work VIBE [18] adopted a new 3D motion capture dataset called AMASS [19] as a good solution for the supervision with 3D data. At the same time, VIBE fed self-improving features from each frame extracted from the SPIN method to Recurrent Neural Networks (RNNs) [20, 21] for temporal information learning. RNNs present an intuitive method to deal with the sequential video data, which can be applied on various input such as 3D joints, 2D joints, or image features extracted by Resnet50. However, RNNs suffered from problems such as error accumulation and gradient explosion. Alternatively, feed-forward convolutional networks do not process input frame by frame and rather feed all the frames at once. As a result, the smoothness of RNN based method is inferior to the feed-forward convolutional networks.

At present, the main problem is that previous methods fail to guarantee the high accuracy and smoothness together. Recovering the 3D human mesh from 2D videos still needs to be improved. To tackle this, we take inspiration from SPIN [11], which embraced the strengths of previous methods and made use of them in tight collaboration. We address this by proposing Video2mesh, a temporal convolutional transformer (TConvTransformer) based method to recover 3D human mesh from 2D video. Unlike the state-of-the-art RNN-based method [18], our Video2mesh method adopts the temporal convolutional block over image feature extractor for sequence local smoothness. Specifically, the Temporal Convolution Networks (TCNs) outperforms RNNs in some video-based applications such as human action recognition and human motion prediction [22–24] due to the parallelisable training process with more stable gradients, which is a promising alternative to guarantee the integrity and smoothness of video data by efficient temporal feature extraction. However, the aggregated temporal feature will inevitably lose the middle-frame accuracy since there is no focus on the output middle-frame feature. We thus apply the multi-head attention based transformer to make use of its multi-subspace to help the network capture the information that is more important to the middle frame. Finally, referring to the VIBE [18], an adversarial 3D supervision from the AMASS dataset is applied on the output middle-frame for accurate human pose and shape recovery.

Our Video2mesh has the following technical contributions:

- We leverage the temporal convolution block which obtains the sequence-level smoothness by aggregating image features from adjacent frames.
- We employ a multi-head attention based transformer to improve the frame-level accuracy by exploring middle-frame information in multi-subspace. Our approach is the trade-off between smoothness and accuracy.
- We adopt the temporal convolution transformer in both generator and adversarial discriminator. Extensive quantitative and qualitative experiments show that our method has significant improvements over state-of-the-art methods on three benchmarks.

2 | RELATED WORK

2.1 | 3D human pose and shape estimation from single image

Recent deep learning based work adopt two-stage methods that estimate intermediate results such as keypoint locations and silhouettes [7], body part segmentation [8] to recover the 3D mesh. Undisturbed by background and other information, the two-stage methods help to deal with the domain shift between the indoor MoCap datasets and the in-the-wild datasets. HMR [1] proposed an end-to-end method to infer 3D mesh parameters directly from the single image via iterative error feedback and weak supervision from adversarial prior [25, 26]. However, due to the limited pair of 3D human ground truth in the existing datasets, the direct regression network lacks sufficient single-frame supervision. SPIN [27] applied the iterative optimisation-based method that fits the parametric body model to 2D joints to generate strong supervision for the 3D mesh regression network which in turn provides reliable initialisation for fast and accurate pixel fit.

2.2 | 3D human pose and shape estimation from monocular video

Considering the rich dynamic information and consistency constraints in video, recent works focus on recovering the 3D human mesh from monocular video. Temporal HMR [28] utilise a temporal encoder to aggregate the frame-level feature directly extracted by Resnet50 for sequence-level representation generation and incorporate a hallucinator which predicts 3D pose changes in adjacent frames to maintain temporal consistency. Similar to multi-task learning, the prediction loss helps the temporal encoder to focus on the dynamic changes between nearby frames for better smoothness, however the accuracy of the 3D mesh regression will be limited by the prediction uncertainty.

DSD-SATN [29] associate the decoupled single frame feature in a sequence via an adversarial self-attention temporal network [30], where the self-attention mechanism applied on
the initial input sequences has a bad effect on the smoothness due to the order insensitivity. VIBE [18] apply the GRU (Gate Recurrent Unit), a variant of RNNs, for temporal motion aggregation and adopt unpaired SMPL parameters from the AMASS dataset to obtain the motion prior in an adversarial way, which encounter incoherence and error accumulation among adjacent frames especially in the case of complex background and partial body occlusion.

3 | METHODOLOGY

3.1 | Framework

Our approach is based on a temporal encoding for dynamic scenes that combines the advantages of temporal convolutional block and multi-head attention based transformer to achieve high performance both in temporal consistency and per-frame 3D human mesh accuracy. Our task aims to regress the triangulated 3D human mesh with 6890 vertices from the input 2D video. The overall framework of our model is shown in Figure 1.

In the context of human mesh recovery from video, given N frames of 2D RGB images, the self-improving feature extractor aims to obtain the image feature by iterative fitting [31]. Our method is the first one to adopt Temporal Convolutional Network (TCN) with multi-head attention based transformer to aggregate the temporal information among the sequences for better human pose and shape recovery.

The input 2D image sequences \( X = \{x_1, x_2, \ldots, x_t\} \) of length \( T \) for a single person are first fed into the self-improving feature extractor to obtain the feature \( f_t \in \mathbb{R}^{2048} \) for each frame \( x_t \). In order to aggregate the information among sequences, we adopt a temporal convolutional layers with multi-head attention transformer (TConvTransformer) as the temporal encoder. Next, we adopt a parametric statistical 3D human body model, SMPL, to disentangle the 3D human body into the pose and shape parameters. In order to unify the pose formats of different datasets, we adopt the 14 common nodes of the LSP [32]. Specifically, we regress the parameters \( \Theta = [\theta, \beta] \), in which the parameter \( \theta \in \mathbb{R}^{32} \) describes the global rotation of the human body along with the relative rotation of 23 joint points in axis-angle representation and shape parameter \( \beta \in \mathbb{R}^{10} \) belongs to the first 10 linear coefficients of a PCA shape space of SMPL body model as the middle frame results form the aggregated feature. We regard the TConvTransformer as the generator and employ another TConvTransformer as discriminator \( D \) to generate natural and reliable human poses adversarially, where the real motion sequences \( \Theta_{\text{real}} \) are from the AMASS dataset.

3.2 | 2D video temporal convolution block

Compared to RNNs, our temporal convolutional blocks aggregate the past and future temporal clues to estimate the intermediate feature and the sequence length has no impact on the gradient’s path to avoid exploding and vanishing gradients.

**Figure 1** The framework of our Video2Mesh method. From left to right, we show that a video sequence is sent to CNN for each frame feature extractor. Then, our method adopt TCN with Multi-head attention mechanism as the generator to obtain the aggregated temporal middle-frame feature which is used to regress the parameters of SMPL human model. A temporal discriminator is trained together with our 3D human mesh generator to constraint the pose and shape to be reliable based on the AMASS dataset. In the TconvTransformer module, which consists of TCN block and multi-head attention followed by a Feed-forward network (FFN).
in RNNs. Moreover, the temporal convolutional networks support parallelisation over time and batch processing to maintain the integrity and smoothness of temporal information instead of processing input frame by frame like chain-structured RNNs [20].

In order to make our model generate smooth human mesh output, we adopt a temporal convolutional network (TCN) as shown in Figure 2 with kernel size \( W \) and dilation factor \( D = W^d \) where \( B \) represents ResNet-style blocks. Given a sequence of 2D RGB images self-improving feature [31] as input, the TCN obtains the temporal feature fusion which is used to regress the pose and shape parameter for the middle frame. The input layer takes the concatenated feature for each frame. The one-dimensional convolution with the kernel size \( W \) and the dilation factor \( D \) are performed first in each block, and then the convolution with the kernel size 1 is applied. Each convolution (except for the last layer) is followed by batch normalisation [33], rectifying linear elements [34], and dropout [35]. The filter hyperparameters \( W \) and \( D \) are set in order to form a tree with which the receptive field for any output frame is able to cover all input frames. The receptive field of each block has exponential growth as \( W \) increases, while the number of parameters increases only linearly. The temporal feature aggregated from the adjacent frames provides more temporal dependencies and motion cues in the 3D human mesh generator.

3.3 Multi-head attention transformer

Considering the inherent hierarchical structure and the kernel size of TCN, the correlation among adjacent frames is efficient for modelling and the information redundancy is reduced by the fusion ability of the temporal convolutional network. However the aggregated temporal feature will inevitably lose the middle-frame accuracy since there is no focus on the output middle-frame feature.

In order to further improve the per-frame accuracy and keep the smoothness simultaneously, we adopt the transformer with the Multi-head Attention (MHA) following the TCN for better middle-frame feature representation. The MHA transformer makes use of its multi-subspace to help the network express the information that is more important to the middle frame, so as to express the short-term current frame while keeping the long-term temporal information.

Multi-head attention (MHA) [36–39] is the core module of the transformer, which is a variation of typical scaled dot-product attention (SDPA) [36] as shown in Figure 3. In SDPA, the keys \( K \), values \( V \) and queries \( Q \) come from the aggregated feature \( f_t \). Then \( f_t \) is first linearly projected to \( \{ f_i^t \in \mathbb{R}^{N \times 64}, i = 1, \ldots , 8 \} \) via 8 fully connected (FC) layers. Then SDPA is performed on each \( \{ f_i^t \} \) in parallel. The outputs are concatenated and linearly projected to \( Y_t \in \mathbb{R}^{N \times 512} \) using a FC layer. The SDPA is derived as:

\[
SDPA(Q, K, V) = \text{softmax} \left( \frac{QK^T}{\sqrt{d_k}} \right) V, \tag{1}
\]

\[
SDPA(f_t) = \text{softmax} \left( \frac{f^T f_t}{\sqrt{d_k}} \right) f_t, \tag{2}
\]

where \( d_k = 512 \) is the dimension of inputs and serves as the scaling factor. In Equation 2, the output is the weighted sum of \( f_t \). The weight matrix represents the relation between each pair of frames.

In this work, we employ 8 heads for MHA which can be calculated as:

\[
\text{Multibehead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_8)W_o, \tag{3}
\]
which ensures the smoothness and accuracy of the 3D mesh expression of the output middle sequences from the AMASS dataset.

3.4 3D human mesh temporal discriminator network

Different from the existing method VIBE [18], which adopted the sequence-to-sequence RNN with scaled dot-product attention motion discriminator, we apply the TConvTransformer as discriminator D to achieve the full-to-single (sequence to frame) supervision, that will treat the single-frame results from the temporal generator as false and the aggregated frame from the AMASS dataset as true. Given a sequence of 3D human meshes from AMASS dataset, our TConvTransformer network makes use of the temporal convolutional layer to learn the temporal information in the sequence and the multi-head attention is used to improve the expression of the output middle-frame in multi-subspace, which ensures the smoothness and accuracy of the 3D mesh sequences from the AMASS dataset.

3.5 Loss function

The loss of our model is composed of 2D (x), 3D (X), pose (\(\theta\)) and shape (\(\beta\)) losses when they are available. This is combined with an adversarial loss. Specifically, the total loss of the is:

\[
L_G = L_{3D} + L_{2D} + L_{\text{SMPL}} + L_{\text{adv}}
\]

where each term is calculated as:

\[
L_{3D} = \sum_{i=1}^{T} \|X_i - \hat{X}_i\|_2,
\]

\[
L_{2D} = \sum_{i=1}^{T} \|x_i - \hat{x}_i\|_2,
\]

\[
L_{\text{SMPL}} = \|\beta - \hat{\beta}\|_2 + \sum_{i=1}^{T} \|\theta_i - \hat{\theta}_i\|_2,
\]

To compute the 2D keypoint loss, we need the SMPL 3D joint locations \(\hat{X}(\theta) = \hat{W}M(\theta, \beta)\). The 14 joints were derived from 6890 vertices of the human body mesh using the linear regressor \(\hat{W}\). The linear combination of the regressor ensures the differentiability of the joint position with respect to the shape and the position parameters. We use a weak-perspective camera model with scale and translation parameters \([s, t], t \in \mathbb{R}^2\). With this we compute the 2D projection of the 3D joints \(\hat{x} \in \mathbb{R}^{2x2} = s\Pi(R\hat{X}(\theta)) + t\), where \(R \in \mathbb{R}^{3}\) is the global rotation matrix and \(\Pi\) represents orthographic projection. Furthermore, \(L_{\text{adv}}\) is the adversarial loss defined as:

\[
L_{\text{adv}} = \mathbb{E}_{\theta \sim \rho_C} \left[ (\Theta - 1)^2 \right]
\]

and the objective for discriminator is:

\[
L = \mathbb{E}_{\theta \sim \rho_C} \left[ (\Theta - 1)^2 \right] + \mathbb{E}_{\rho \sim \rho_G} \left[ (\Theta - 2)^2 \right]
\]

where \(\rho_C\) is a real motion sequence from the AMASS dataset, and \(\rho_G\) is a generated motion sequence.

4 EXPERIMENTS

4.1 Datasets and evaluation metrics

4.1.1 3DPW dataset

The 3D Pose in the Wild dataset (3DPW) [47] is the first outdoor dataset captured by IMU sensors and moving phone cameras, which includes accurate 3D body scans, 3D people models and video footage. There are 60 video sequences divided into 24 video sequences for training, 12 video sequences for validation, 24 video sequences for testing.

Following previous method [18], the 3DPW dataset is used for both training and evaluation.

4.1.2 MPI-INF-3DHP dataset

MPI-INF-3DHP Dataset [48] is a multi-view, mostly indoor dataset captured using markerless motion capture system. We use the training set proposed in ref. [48], which consists of 8 subjects and 16 videos per subject, and we evaluate on the official test set.

4.1.3 Human3.6 M dataset

Human 3.6 Million (H3.6 M) dataset [49] is an indoor dataset including multiple action scenarios performed by 11 professional actors. There are five subjects (S1, S5, S6, S7, S8) for training and two subjects (S9, S11) for testing. The SMPL parameters provided by MoSH [50] are used during training.

4.1.4 Evaluation metrics

We evaluate the accuracy of our method according to mean per joint position error (MPJPE), Procrustes aligned mean per
**TABLE 1** Evaluation of state-of-the-art models on 3DPW, MPI-INF-3DHP, and H3.6M datasets. Ours (w3dpw) is our proposed model trained on video datasets similar to [31, 40] and VIBE (wo3dpw) [18], while Ours (w3dpw) is trained with extra data from the 3DPW training set similar to VIBE (w3dpw) [18]. Our method outperforms all state-of-the-art temporal models including VIBE [18] on the challenging in-the-wild datasets (3DPW and MPI-INF-3DHP) and obtains comparable result on H36 M. ‘-’ shows the results that are not available.

<table>
<thead>
<tr>
<th>Models</th>
<th>3DPW PA-MPJPE</th>
<th>3DPW MPJPE</th>
<th>3DPW PVE</th>
<th>3DPW Accel</th>
<th>MPI-INF-3DHP PA-MPJPE</th>
<th>MPI-INF-3DHP MPJPE</th>
<th>MPI-INF-3DHP PCK</th>
<th>H36 M PA-MPJPE</th>
<th>H36 M MPJPE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kanazawa et al. [41]</td>
<td>76.7</td>
<td>130.0</td>
<td>-</td>
<td>37.4</td>
<td>89.8</td>
<td>124.2</td>
<td>72.9</td>
<td>56.8</td>
<td>88</td>
</tr>
<tr>
<td>Omran et al. [42]</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Pavlakos et al. [43]</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Kolotouros et al. [44]</td>
<td>70.2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Arnab et al. [45]</td>
<td>72.2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>54.3</td>
<td>77.8</td>
</tr>
<tr>
<td>Kolotouros et al. [31]</td>
<td>59.2</td>
<td>96.9</td>
<td>116.4</td>
<td>29.8</td>
<td>67.5</td>
<td>105.2</td>
<td>76.4</td>
<td>41.1</td>
<td>-</td>
</tr>
<tr>
<td>Kanazawa et al. [15]</td>
<td>72.6</td>
<td>116.5</td>
<td>139.3</td>
<td>15.2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>56.9</td>
<td>-</td>
</tr>
<tr>
<td>Doersch et al. [46]</td>
<td>74.7</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Sun et al. [11]</td>
<td>69.5</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>42.4</td>
<td>59.1</td>
</tr>
<tr>
<td>I2L et al. [40]</td>
<td>60.0</td>
<td>100.0</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>41.1</td>
<td>55.7</td>
</tr>
<tr>
<td>VIBE (wo3dpw) et al. [18]</td>
<td>56.4</td>
<td>96.7</td>
<td>115.0</td>
<td>28.1</td>
<td>64.8</td>
<td>99.1</td>
<td>87.4</td>
<td>41.8</td>
<td>66.3</td>
</tr>
<tr>
<td>VIBE (w3dpw) et al. [18]</td>
<td>54.1</td>
<td>87.3</td>
<td>103.2</td>
<td>27.9</td>
<td>63.4</td>
<td>97.7</td>
<td>89.0</td>
<td>41.5</td>
<td>65.9</td>
</tr>
<tr>
<td>Ours (wo3dpw)</td>
<td>53.2</td>
<td>86.4</td>
<td>102.1</td>
<td>8.5</td>
<td>61.9</td>
<td>96.8</td>
<td>89.6</td>
<td>41.6</td>
<td>66.1</td>
</tr>
<tr>
<td>Ours (w3dpw)</td>
<td><strong>51.4</strong></td>
<td><strong>84.4</strong></td>
<td><strong>100.5</strong></td>
<td><strong>6.7</strong></td>
<td><strong>61.2</strong></td>
<td><strong>95.3</strong></td>
<td><strong>90.8</strong></td>
<td><strong>40.9</strong></td>
<td><strong>65.4</strong></td>
</tr>
</tbody>
</table>

**Note:** The meaning of the values provided in bold are the best quantified results compared with other methods.

**FIGURE 4** Visual accuracy comparisons between VIBE and our Video2mesh method. From left to right, we show the qualitative comparison. As indicated by the circles, the recent state-of-the-art approach VIBE fails to produce accurate 3D human mesh compared with our Video2mesh method in challenging 2D image pose in MPI dataset.
joint position error (PA-MPJPE), Percentage of Correct Keypoints (PCK) and Per Vertex Error (PVE) on 3DPW [47], MPI-INF-3DHP Dataset and Human3.6 M Dataset [49]. Acceleration error is used to evaluate the smoothness of our method by calculating the average difference between ground truth 3D acceleration and predicted 3D acceleration of each joint in mm/s².

4.1.5 Implementation details

Our proposed algorithm is implemented on Pytorch [51] and trained on a NVIDIA Tesla V100 GPU. We adopted the Adam [52] optimiser to train our model for about 100 epochs. We experimented with different input sequence length $T = [3, 9, 15]$, we use $T = 9$, as it yields the best results. The learning rate was set to 0.0001 and the batch size was 128. Our models use pretrained model from SPIN [31] as the self-improving feature extractor.

4.2 Comparison to state-of-the-art results

We compare our model Video2mesh with previous state-of-the-art methods on 3DPW, MPI-INF-3DHP and Human3.6 M in Table 1.

We utilise the same setting as VIBE [18]. Our method is trained on video datasets similar to VIBE (w3dpw) and on extra data from the 3DPW training set similar to VIBE (w3dpw).

The results show that our method has better performance in not only accuracy (i.e. PA-MPJPE, MPJPE, PVE) but also smoothness which is illustrated from the acceleration error in the challenging in-the-wild 3DPW by a significant amount. These results confirm our hypothesis that the use of feed-forward temporal convolution network with multi-head attention based transformer in our video temporal encoder and mesh discriminator is important for dealing with the temporal information in video for competitive pose and shape estimation.

4.3 Visualisation

We visualised the results of our method in comparison with the state-of-the-art VIBE [18] in the MPI dataset [48]. As shown in the Figure 4, given continuous video input, the 3D projection of human body model obtained by our method is more consistent with the input 2D image than VIBE. At the same time, the advantages of our method are even more obvious in the hands and feet as shown in the circles.

More visualisations on Internet images are shown in Figure 5. We observed that on both input view and another side view, VIBE produced obvious errors, especially when hands and arms were crossed. The visualisation shows that our method is able to estimate more accurate 3D human mesh than
Figure 6 Visual smoothness comparisons between VIBE and our Video2mesh method on Internet video sequences. Given the same video sequence, we observe that our Video2mesh method produces more smooth intermediate results than VIBE.

5 | ABLATION STUDIES

5.1 | Comparison of model complexity

We also compare the computation complexity between VIBE and our network as shown in Table 2. The FLOPs of our network is 14.224 G and the FLOPs of VIBE is 12.794 G as shown in Table 2. The frame rate of the video is 30 FPS and the resolution is 1080P. The number of parameters of our network is 92.1 M and VIBE is 86.4 M. Although our method will increase FLOPs and the number of parameters, the inference time of our network is 1.01s which is slightly higher than VIBE. Here, the batch size is 128 and the GPU is a NVIDIA Tesla V100 GPU. Since the improvement of accuracy and smoothness are more important for this human mesh recovery task, appropriate time consumption is acceptable.

Table 2 Comparison on model complexity. We calculate the FLOPs, the number of parameters and inference time between VIBE and our method. Despite the higher complexity of our model, the difference in inference time is acceptable considering the accuracy and smoothness.

<table>
<thead>
<tr>
<th>Model</th>
<th>FLOPs</th>
<th>Parameters</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>VIBE [18]</td>
<td>12.8 G</td>
<td>86.4 M</td>
<td>0.96s</td>
</tr>
<tr>
<td>Ours</td>
<td>14.2 G</td>
<td>92.1 M</td>
<td>1.01s</td>
</tr>
</tbody>
</table>

Note: The meaning of the values provided in bold are the best quantified results compared with other methods.

Table 3 Ablation study on different temporal encoders. RNN is the temporal model adopted by VIBE [18]. TCN is the baseline model for our method. Then, we add self-attention (SA) in front of the TCN as SATCN and after the TCN as TCNSA. The results show that the improvements are not significant. We also add multi-head attention (MHA) before the TCN as MATCN and after TCN as TConvTransformer. The results show that our TConvTransformer model has the best performance in both accuracy and smoothness.

<table>
<thead>
<tr>
<th>Methods</th>
<th>3DPW</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PA-MPJPE ↓</td>
</tr>
<tr>
<td>RNN</td>
<td>59.6</td>
</tr>
<tr>
<td>TCN</td>
<td>56.8</td>
</tr>
<tr>
<td>SATCN</td>
<td>52.7</td>
</tr>
<tr>
<td>TCNSA</td>
<td>56.5</td>
</tr>
<tr>
<td>MATCN</td>
<td>52.4</td>
</tr>
<tr>
<td>TConvTransformer</td>
<td>52.2</td>
</tr>
</tbody>
</table>

Note: The meaning of the values provided in bold are the best quantified results compared with other methods.

Table 4 Ablation study on different discriminators in our temporal method. We report the adversarial training comparisons on the 3DPW dataset with only generator (Only G), single-frame discriminator (w/Single frame D) and our temporal based TConvTransformer discriminator (w/Our D). The results show that our TConvTransformer based discriminator outperforms other designs.

<table>
<thead>
<tr>
<th>Methods</th>
<th>3DPW</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PA-MPJPE ↓</td>
</tr>
<tr>
<td>Only G</td>
<td>52.2</td>
</tr>
<tr>
<td>w/Single frame D</td>
<td>51.9</td>
</tr>
<tr>
<td>w/Our D</td>
<td>51.4</td>
</tr>
</tbody>
</table>

Note: The meaning of the values provided in bold are the best quantified results compared with other methods.

5.2 | Different temporal encoders

We conduct the ablation studies on different temporal modules which are evaluated on 3DPW with MPJPE as shown in Table 3. The results show that TCN alone is superior to RNN in both smoothness and accuracy. While the accuracy is improved by using self-attention in front of the TCN as SATCN due to the better long-term correlation learning ability, the smoothness is affected by frame-by-frame position embedding, especially in some scenes with rapid changes in...
actions, which leads to jittery from the attention mechanism and misleads the current intermediate frame recovery. Additionally, we put the self-attention after the TCN as TCNSA. At this point, the model only keep the smoothness and accuracy compared with TCN. The results also show that simply adding parameters fail to improve the effect of the model. We also carry out experiments to quantify the influence of our multi-head attention (MHA) on different location. Firstly, we utilise the MHA before the TCN as MATCN. The results show that such model only improves the accuracy not the smoothness. Then we build the MHA behind the TCN as TConvTransformer. The results show that our TConvTransformer module which moves the multi-head attention after the temporal convolutional block, has the best performance compared with other methods in accuracy and smoothness.

5.3 | Different mesh discriminators

In order to verify the advantages of our discriminator, we compare the performance of different discriminators under the same generator in Table 4. This includes a generator (G) without any discriminator, with a single-frame discriminator, and with our TConvTransformer based discriminator. In the AMASS sequence data set, the single-frame discriminator can improve the accuracy slightly, but does not contribute to the smoothness. The results show that the smoothness is not satisfied due to the lack of sufficient learning of the sequence training data. Moreover, the sequence discriminator using our TConvTransformer can learn temporal sequence information from the AMASS dataset to improve smoothness without affecting the accuracy.

6 | CONCLUSION

In this paper, we propose Video2mesh, a novel Temporal Convolutional Transformer (TConvTransformer)-based method to recover 3D human mesh from video. Specifically, the temporal convolution block in our TConvTransformer aggregates the adjacent-frame feature leading to global smoothness. It reduces the redundancy of the sequence while preserves the sequence-level information. The subsequent transformer adopts the multi-head attention among the multi-subspace of the aggregated feature to improve the local middle-frame accuracy and maintain the smoothness. Moreover, a global-to-local supervision is applied by using our TConvTransformer adversarially on the AMASS dataset, which further improves the accuracy and smoothness. Experimental results on three challenging benchmark datasets show that our proposed Video2mesh outperforms the state-of-the-art approaches in accuracy and smoothness. The limitation of our method is that it is not able to achieve real-time estimation at present. Nevertheless, we will explore ways to speed up the computation in order to better apply it for video-based 3D human pose and shape recovery and dynamic 3D reconstruction.

AUTHOR CONTRIBUTIONS

Xianjin Chao: Conceptualisation, Methodology, Visualisation, Writing – original draft. Zhipeng Ge: Methodology, Software. Howard Leung: Supervision, Writing – review & editing.

ACKNOWLEDGEMENT

The work described in this paper was supported by a grant from City University of Hong Kong (Project Nos. 9220077 and 9678139).

CONFLICT OF INTEREST

We declare that we have no financial and personal relationships with other people or organisations that can inappropriately influence our work, there is no professional or other personal interest of any nature or kind in any product, service and/or company that could be construed as influencing the position presented in, or the review of, the manuscript entitled.

DATA AVAILABILITY STATEMENT

Data available on request due to privacy/ethical restrictions.

ORCID

Xianjin Chao https://orcid.org/0000-0002-4020-8223

REFERENCES


How to cite this article: Chao, X., Ge, Z., Leung, H.: Video2mesh: 3D human pose and shape recovery by a temporal convolutional transformer network. IET Comput. Vis. 17(4), 379–388 (2023). https://doi.org/10.1049/cvi.20212172