# CVIBE: Consistent Video Inference for Human Body Pose and Shape Estimation

Zhiyi Lai
College of Information and Computer Sciences
UMass Amherst
zhiyilai@umass.edu

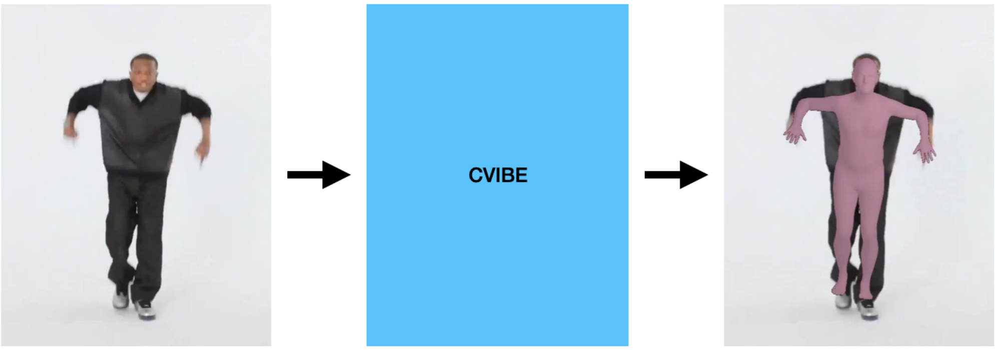

Figure 1: The input and output example of our CVIBE model

## Abstract

*Human motion information is fairly important in several areas including human motion analysis, game making and movie visual effects. Despite the progress on single-image 3D pose and shape estimation, only a few of existing video-based state-of-the-art methods are able to produce a good enough motion sequences. Existing state-of-the-art methods suffer from problems like flickering and inconsistence. To address these problems, we propose ”***C***onsistent* **V***ideo* **I***nference for* **B***ody Pose and Shape* **E***stimation”(CVIBE), which is a modified version of the VIBE methods. Our key novelty is a consistent loss to force the model produce a more consistent and smooth version of the motion sequence and a self attention module to replace the original gated recurrent unit(GRU) module. We perform several experimentation to analyze the performance of* **CVIBE** *on challenging 3D pose estimation datasets, achieving a slightly better performance compares to the state-of-the-art model baseline we trained.*

## 1. Introduction

Since 3D human pose and shape information has a lot of usage scenarios in the real world, there are tremendous methods has been proposed to address this problem. While there are several works [6, 12, 17, 19, 22] on the single image 3D human pose and shape estimation and they can produce good results, they will have a bad performance when predicting consecutive video frames, which contain the human motion information that have a wider usages in real world applications. With an accurate and stable human motion extraction methods, we can get tons of human motion data for motion analysis deep learning models, record animations for 3D game models and cinematic visual effects for the Hollywood movies. There are some existing video-based state-of-the-art methods that are able to produce a good enough motion sequences. One of the state-of-the-art method is the VIBE model provided by Kocabas [21], which utilize a discriminator to help the generator model generates natural sequences from videos with the help of a large ground-truth 3D motion dataset proposed by Mahmood [27]

However, the 3D model sequences generated by the VIBE model suffer from the flickering and inconsistence in shape problems. Which can be seen at this demo video [2].

To address the flickering problem, we take inspiration from Luo [26] who proposed a consistent loss focusing on the depth consistency of the same point in the 3D world between frames to solve the depth estimation problem. The question is what is a proper property that we can restrict as consistent. For that, we leverage the fact that the human in the input data frames has the same shape and the movement between two frames are small. Our approach add a consistent loss on top of the original loss from the VIBE [21] paper, which restrict the generated model sequence to have a same shape and close keypoint positions and angles.

Also, the Transformer model proposed by Vaswani [39] can better capture the global temporal information than the GRU [8] module does. Therefore, in CVIBE model, we replace the GRU module in the generator to a transformer encoder which contains several self attention layers.

The input of out model is a sequence of video clips that contain human motion and the output are sequences of shape and pose parameters for the SMPL [25] human model. This is shown in Figure 1. In training time, we will firstly preprocess the datasets to features with a CNN network. Then we train our model on these features. At inference time, we will firstly run a human detection on the input sequence, then generate features and predict the SMPL parameters.

In summary, we proposed two modifications to try to improve the performance of the original VIBE model, called CVIBE. The first modification is a consistent loss that captures inter-frame relationship. The second modification is changing the GRU [8] modules in the generator to self-attention modules like the Transformer model. We achieve a slightly better performance on the original VIBE model baseline. Code are available at `https://github.com/L4zyy/CVIBE`.

## 2. Related Work

**3D pose and shape from a single image** Since the ground truth 3D human mesh model are hard to get, parametric 3D human body models [4, 25, 33] are widely used as the output target for human pose estimation because they capture the statistics of human shape and provide a 3D mesh that can be used for many tasks. Early works [1, 5, 11, 36] lack of stability, need manual intervention or do not generalize well to in the wild data. Recently, deep neural networks are trained to directly regress the parameters of the SMPL human model from images [12, 19, 31, 34, 40, 37]. Due to the lack of in-the-wild 3D ground-truth labels, these methods use weak supervision signals obtained from a 2D keypoint reprojection loss [19, 40, 37], use body/part segmentation as an intermediate representation [31, 34], or employ a human in the loop [24]. Despite capturing the human body from single images, when applied to video, these methods tield jittery, unstable results.

**3D pose and shape from video** The caputre of human motion from video has a long history. Early approaches including fitting a simplified human body model to images features of a walking person [15], exploiting methods like PCA and GPLVMs to learn motion priors from mocap data [32, 38]. However, these methods were limited to simple motions. Many of the recent deep learning methods that estimate human pose from video [9, 16, 29, 33, 30] focus on joint locations only. Several methods [9, 16, 33] use a two-stage approach to "lift" off-the-shelf 2D keypoints into 3D joint locations. VIBE model proposed by Kocabas [21] utilized an existing large-scale motion capture dataset called AMASS [27] as well as the adversarial training at the sequence level to produces kinematically plausible motion sequences without in-the-wild ground-truth 3D labels.

**Sequential data processing** The original Recurrent Neural Networks [18] give us a way to deal with sequential data and capture information from them. However, Recurrent Neural Networks suffer from short-term memory problem. If a sequence is long enough, they will have a hard time carrying information from earlier time steps to later ones. Long Short-Term Memory proposed by Hochreiter [14] and Gated Recurrent Unit proposed by Chung [8] are seen as solutions to deal with this short-term memory problem. They have internal mechanisms called gates that can regulate the flow of information. These gates can learn which data in a sequence is important to keep or throw away. By doing that, it can pass relevant information down the long chain of sequences to make predictions. However, LSTM and GRU suffer from the long processing time problem. Like LSTM and GRU, Transformer proposed by Vaswani [39] is an architecture for transforming one sequence into another one without imply any recurrent networks. With transformer, we can process the data sequence in parallel, which enable us to get a better global information.

**Consistent Loss** The depth estimation from video is also a field that need to process sequential data. Besides the typical problems that any reconstruction system has to deal with, such as poorly textured areas, repetitive patterns, and occlusions, there are several additional challenges with video: higher noise level, shake and motion blur, rolling shutter deformations, small baseline between adjacent frames, and, often, the presence of dynamic objects, such as people. For these reasons, existing methods often suffer from a variety of problems, such as missing regions in the depth maps (Figure 1b) and inconsistent geometry and flickering depth. To solve these problems, Luo [26] proposed a new video-based reconstruction system that combines the strengths of traditional and learning-based techniques, which achieves higher accuracy and a higher degree of geometric consistency than previous monocular reconstruction methods and generates more visually stable results.

## 3. Approach

Our CVIBE model is based on the VIBE model proposed by Kocabas [21], the overall structure of our model is sum-

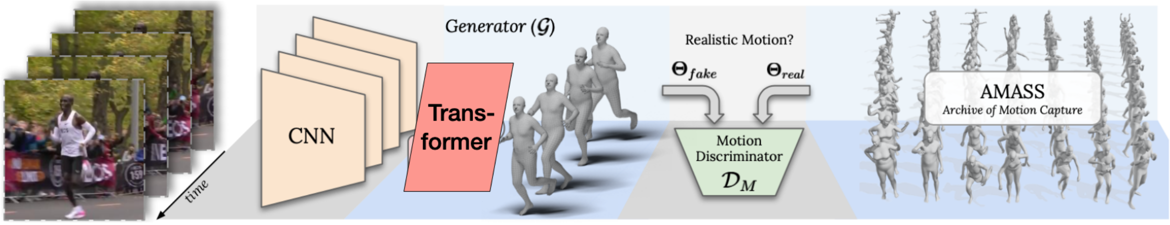

Figure 2: **CVIBE structure** The CVIBE model takes video sequence features processed by a CNN network as inputs and estimates the SMPL body model parameters for each frame in the video sequence as outputs. The generator is trained together with a motion discriminator, which has access to a large corpus of human motions in SMPL format. (This figure is modified based on the VIBE paper [21])

marized in Figure 2. Given a video sequence $V = \{I_t\}_{t=1}^T$ with length $T$ as input, we output the parameters for the SMPL body model. The SMPL parameter consists of pose parameter $\theta \in \mathbb{R}^{72}$ and shape parameter $\beta \in \mathbb{R}^{10}$. The pose parameters include the global body rotation and the relative rotation of 23 joints in axis-angle format. The shape parameters are the first 10 coefficients of a PCA shape space. Given these parameters, the SMPL model is a differentiable function, $\mathcal{M}(\theta, \beta) \in \mathbb{R}^{6890\times3}$, that outputs a posed 3D mesh.

To get the output, we firstly put our input video sequence through a pretrained CNN network to get the $D \times T$ feature sequence where $D$ is the length of the one dimentional output of the pretrained CNN network. After getting the feature sequence, we train a transformer encoder with several self attention layers and multiple heads to process the temporal feature sequence and output latent vectors. Then, we train a fully connected layer to regress these latent vectors to the SMPL pose and shape parameters. Following the VIBE paper [21], we apply average pooling on all $\hat{\beta}^i$ to get a single shape $(\hat{\beta})$ across the whole input sequence. The modules we described above form our generator $\mathcal{G}$. Following the VIBE paper [21], we feed a discriminator $\mathcal{D}_M$ with the output $\hat{\Theta} = [\hat{\theta}, \hat{\beta}]$ of our generator as well as real 3D human model sequences parameters $\Theta_{real}$ from dataset AMASS [27]

In the following part of this section, we will describe the approaches we use to improve the performance of our CVIBE model. Since our CVIBE model is based on the VIBE model [21], we will firstly describe a part of the loss function and the discriminator part of our model that has the same structure with the VIBE model in the first two subsection. Then, we will describe the modified encoder and the consistent loss in the third and fourth subsections.

### 3.1. Main Loss

We utilize all the loss terms in the original VIBE paper [21] as the main part of our loss. This part of the loss composed of 2D(x), 3D(X), pose($\theta$) and shape($\beta$) losses when they are available (Since we follow the VIBE model and they use both 2D and 3D datasets, therefore, sometimes only part of these losses are available). These are combined with an adversarial $\mathcal{D}_M$ loss. Specifically the main loss of the $\mathcal{G}$ is:

$$L_{\mathcal{G}} = L_{3D} + L_{2D} + L_{SMPL} + L_{adv} \tag{1}$$

Where each term is calculated as:

$$L_{3D} = \sum_{t=1}^{T} ||X_t - \hat{X}_t||_2,$$

$$L_{2D} = \sum_{t=1}^{T} ||x_t - \hat{x}_t||_2,$$

$$L_{SMPL} = ||\beta - \hat{\beta}||_2 + \sum_{t=1}^{T} ||\theta_t - \hat{\theta}_t||_2$$

The adversarial loss $L_{adv}$ is defined in the next subsection.

We will use the same method to map 3D keypoints to 2D and calculate 2D loss, which first use body vertices and a pretrained regressor $W$ to compute the 3D joint locations $\hat{X}(\Theta) = W\mathcal{M}(\theta, \beta)$, then use a weak-perspective camera model with scale and translation parameters $[s, t], t \in \mathbb{R}^2$ to calculate 2D projection of the 3D joints as $\hat{x} \in \mathbb{R}^{j\times2} = s\pi(R\hat{X}(\Theta)) + t$ where $R \in \mathbb{R}^3$ is the global rotation matrix and $\pi$ represents orthographic projection.

### 3.2. Motion Discriminator

Following the VIBE paper [21], we trained a motion discriminator $\mathcal{D}_M$ to tell us whether the generated sequence of poses corresponds to a realistic sequence or not. The output of the generator $\hat{\Theta}$ is given as input to a multi-layer GRU model $f_M$ depicted in Figure 3, which estimates a latent code $h_i$ at each time step $i$ where $h_i = f_m(\hat{\Theta}_i)$. We use self attentiion layers to aggregate hidden states $[h_i, \dots, h_T]$ elaborated below. Finally, a linear layer predicts a value $\in [0, 1]$ representing the probability that $\hat{\Theta}$ belongs to the manifold of plausible human motions. The adversarial loss

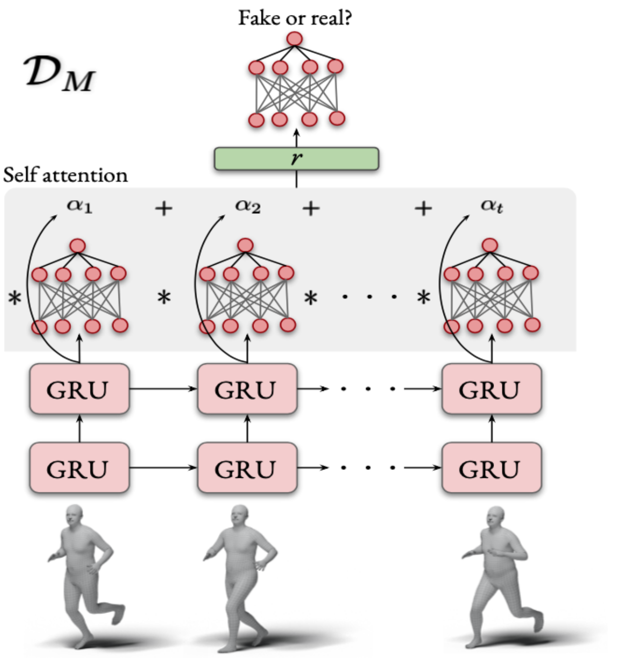

Figure 3: **Motion discriminator structure** $\mathcal{D}_M$ consists of GRU layers followed by a self attention layer. $\mathcal{D}_M$ outputs a real/fake probability for each input sequence.(This figure is borrowed from the VIBE paper [21])

term that is backpropagated to $\mathcal{G}$ is:

$$L_{adv} = \mathbb{E}_{\Theta \sim p_G}[(\mathcal{D}_M(\hat{\Theta}) - 1)^2] \tag{2}$$

and the objective for $\mathcal{D}_M$ is:

$$L_{\mathcal{D}_M} = \mathbb{E}_{\Theta \sim p_R}[(\mathcal{D}_M(\Theta) - 1)^2] + \mathbb{E}_{\Theta \sim p_G}[\mathcal{D}_M(\hat{\Theta})^2] \tag{3}$$

where $p_R$ is a real motion sequence from the AMASS [27] dataset while $p_G$ is a generated motion sequence. Since $\mathcal{D}_M$ is trained on ground-truth poses, it also learns plausible body pose configurations.

### 3.3. Temporal Encoder

To capture the whole information of the input sequences in parallel, we choose to switch the GRU module in the original VIBE model [21] to self attention layers like the encoder in the Transformer model [39]. The intuition behind using self attention layers to replace the recurrent modules like GRU is that, in the self attention layers, we consider all the frames in the sequence at once, which let us get a more consistent result among all the frames. This consistency improvement is shown in our experiment section.

Given a sequence of frames $I_1, \ldots, T_T$, outputs the corresponding pose and shape parameters in each frame. The input frames is fed to a pretrained CNN and get a feature sequence $[f_1, \ldots, f_T]$ where $f_i \in \mathbb{R}^{2048}$. These are sent to a transformer encoder, which contains several self attention layers $s_i$ where $i \in S$ stand for the number of the self attention layers, to get the intermediate results $[\hat{h}_1, \ldots, \hat{h}_T]$. Finally, we sent these intermediate results to a regressor $r$ to get the final output of our generator $\hat{\Theta}$. The output of the whole temporal encoder is:

$$\hat{h}_i = s_1(s_2(\ldots s_S(f_i)))$$
$$\hat{\Theta} = r(\hat{h})$$

Following the VIBE paper [21], we use a 6D continuous rotation representation [42] instead of axis angles.

### 3.4. Consistent Loss

Despite the original VIBE model [21] utilized interframe ralationship on the discriminator, it failed to use this relationship on the loss of the generator. To keep the consistent in the pose, we limited the generated sequences to have a consistent pose in a short range, which means that the difference between the start frame and the end frame of a gap should be small. For a time period $T$, we choose a gap $g$ and accumulate all the difference value in the time $T$ with the gap $g$. Therefore, we proposed a consistent loss $L_{const}$:

$$L_{const} = \sum_{i=1}^{T-g} \frac{1}{T-g} ||\theta_i - \theta_{i+g}|| \tag{4}$$

Therefore, the total loss of the $\mathcal{G}$ is:

$$L_{\mathcal{G}} = L_{3D} + L_{2D} + L_{SMPL} + L_{adv} + L_{const} \tag{5}$$

## 4. Experiment

In this section, we first describe the training procedure. Next, we describe the datasets for training and evaluation. Then, we compares different methods with the VIBE baseline trained on the selected datasets. Finally, we will analyze some interesting results.

### 4.1. Training Procedure

We use a ResNet-50 network [13] as the feature extractor pretrained on single frame pose and shape estimation task [19, 23] that outputs $f_i \in \mathbb{R}^{2048}$. Following VIBE paper [21], we precompute each frame's $f_i$ and do not update the ResNet-50. We use $T = 16$ as the sequence length with a minibatch size of 32, which makes it possible to train our model on a single Nvidia RTX1080 Max-Q GPU. For the temporal encoder, we use several self attention layers followed by a final regressor layer that outputs $\hat{\Theta} \in \mathbb{R}^{85}$, containing pose, shape and camera parameters. For the self attention layers, we use 6 self attention layers modified from Carion and Massa's work [7] with 8 heads. We have also tried 3 layers with 4 heads, but more layers and heads yields

| Performance Comparison | | | | |
|---|---|---|---|---|
| Methods | PA-MPJPE↓ | MPJPE↓ | PVE↓ | Accel↓ |
| Baseline | 55.48 | 77.9 | 94.86 | 25.98 |
| CLO (300, 8) | 55.09 | 77.68 | 94.35 | 27.26 |
| CLO (100, 8) | 55.43 | 77.78 | 94.98 | 26.1 |
| CLO (300, 4) | 55.47 | 77.69 | 94.55 | 25.76 |
| CL (300, 4) | 74.57 | 103.1 | 123.4 | 17.46 |
| CL (100, 4) | 64.06 | 87.8 | 106.4 | 18.67 |
| CL (300, 8) | 81.38 | 115.1 | 138.84 | 17.44 |
| CL (100, 8) | 70.68 | 97.5 | 118.3 | 20.01 |
| SA+CLO (3, 4, 300, 8) | 54.18 | 76.55 | 93.35 | 18.53 |
| SA+CLO (3, 4, 100, 8) | 54.98 | 77.6 | 94.64 | 18.71 |
| SA+CL (6, 8, 300, 8) | 63.92 | 88.71 | 106.4 | 19.95 |
| SA+CL (6, 8, 100, 8) | 61.47 | 85.09 | 102.4 | 20.82 |
| SA (3, 4) | 54.05 | 76.1 | 92.77 | 16.98 |
| SA (6, 8) | **53.7** | **75.96** | **92.2** | **13.88** |

Table 1: The CLO, CL, SA stand for old version consistent loss, consistent loss and self attention layer. The parameters for consistent loss is the weight of this loss and the frame gap we mentioned in Section 3.4. The parameters for self attention layers is the number of layers and the number of heads. All the evaluation metrics is the smaller the better.

better results. The outputs of the genrator are given as input to the $\mathcal{D}_M$ as fake samples along with the ground truth motion sequences as real samples. We borrow the motion discriminator architecture from the VIBE paper [21]. For self attention, we use 2MLP layers with 1024 meurons each and $tanh$ activation to learn the attention weights. The final linear layer predicts a single fake/real probability for each sample. We also use the Adam optimizer with a learning rate of $5 \times 10^{-5}$ and $1 \times 10^{-4}$ for the $\mathcal{G}$ and $\mathcal{D}_M$, respectively.

## 4.2. Datasets

The original VIBE paper [21] train their model on several 2D and 3D datasets including AMASS [27], InstaVariety [20], MPI-3D-HP [28], 3DPW [41], PennAction [10] and PoseTrack [3].

We utilize one 2D dataset and one 3D dataset for training and one 3D dataset as validation dataset. Specifically, we use InstaVariety [20] as our 2D dataset and 3DPW [41] as our 3D training and validation dataset. We also use AMASS [27] for the discriminator training. We will simply describe these datasets.

(a) **AMASS** [27] is a large and varied database of human motion that proposed by Mahmood [27], which unifies 15 different optical marker-based mocap datasets by representing them within a common framework and parameterization.

(b) **InstaVariety** [20] is a dataset proposed by Kanazawa [20], which contain diverse rangeof human dynamics. The data are collected from Instagram using 84 hashtags such as #instruction, #swimming, and #dancing. A large proportion of the videos collected contain only one or two people moving with much of their bodies visible, which is good for 3D human model extraction.

(c) **3DPW** [41] is the first dataset in the wild with accurate 3D poses for evaluation. While other datasets outdoors exist, they are all restricted to a small recording volume. 3DPW is the first one that includes video footage taken from a moving phone camera.

## 4.3. Comparison and Findings

In this section, we first compare the final performance of the methods we tried with the baseline. Next, we show the loss of some methods and give a guess of what happened to cause this. The overall comparison of the performance is shown in Table 1.

**Numerical Evaluation Metrics** We will utilize the evaluation metrics used by the orig-inal VIBE paper [21]. Which includes MPJPE (Mean Per Joint Position Error), PA-MPJPE (Procrustes Analysis Mean Per Joint Position Error), PVE (Per Vertex Error),PCK [35] (Percentage of Correct Keypoints) and the acceler-ation error which measures the acceleration difference of keypoints.

**Baseline** Since the original VIBE paper [21] trained their model on multiple datasets, due to the time and device limitation, we train the VIBE model on two datasets we selected in Section 4.2 for 80 epochs as our baseline.

**Old version consistent loss** The old version we use will calculate the accumulation first then get the absolute value of the loss, Although it has a very slightly performance improvement, we claim that that may due to the fact that the overall loss will be very small since the positive and nega-

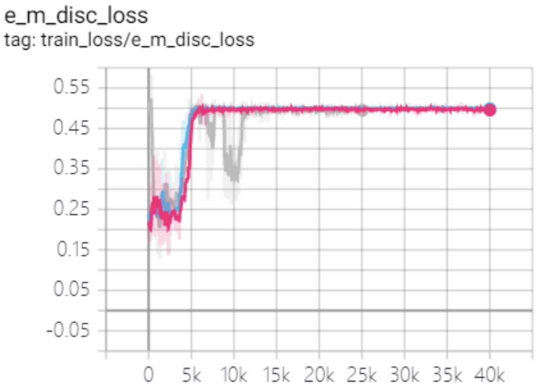

Figure 4: This figure shows that the adversarial loss $L_{adv}$ is stay in high value after first 25% epochs. The red line is the baseline, the blue line is the line with the old version of the consistent loss, and the grey line is the model with self attention layers.

tive value will cancel each other and this small performance improvement is just caused by the randomness.

**Consistent Loss** We can see that the consistent loss will drop the performance for a large amount since it limit the position of the keypoints. However, we find that using consistent loss will make the acceleration loss smaller. This means that the consistent loss is making the output sequence more stable.

**Self Attention Layers** With self attention layers, we observe a improvements in all metrics. And the acceleration error drop a significant amount. We assume that the most performance improvement is due to the fact that the self attention layers better capture the global motion information, therefore, the acceleration is more matching the ground truth motion sequence. Due to the device limitation (lacks of GPU memory), we can not generate videos with our model using the demo code provided by Kocabas [21]. However, with the acceleration performance improvement, we assume that the global information captured by the self attention layers are actually helping the model to generate less flickering output sequence than the one generated by the VIBE model [21].

**Combination** We test the performance with both old version consistent loss and the self attention layers. However, the overall performance are dropping slightly. We assume that this is because that the self attention layers, while recieve the global information, is already learned how to generate a stable output.

**discrimination Loss Analysis** Although the self attention layers improve the performance, we found the discrimination loss has the same patterns with the original VIBE model [21] even after we switch to self attention layers. This is shown in Figure 4. We assume that this may indicate that there are some modifications that we can do on the discriminator part to further improve the performance of this model, which is one of the future works that we want to try.

## 5. Conclusion

Most 3D human pose methods works well on single images, but failed on the video sequence data. While current 3D human pose and shape from video methods works well, it suffer from flickering problem. Here we explore several novel methods to extend 3D human pose and shape from video methods: (1) We test our consistent loss with the original VIBE model; (2) We replace the GRU modules with several self attention layers. We carefully evaluate our contributions and shows that, for the consistent loss, while the performance on PA-MPJPE and PVE are dropping a lot, we still get a more consistent outputs since the acceleration error is dropping; for the self attention layers, we shows that it improve the original VIBE model and the main contribution is coming from the acceleration error dropping.

**Future Work** As we discussed in Section 4.3, the adversarial loss is high for both the VIBE model and our CVIBE model. Therefore, we want to modify the discriminator in the future to further improve the performance of our model.

## References

[1] A. Agarwal and B. Triggs. Recovering 3d human pose from monocular images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1):44–58, 2006.

[2] S. Alex. Marinera vibe (video inference for human body pose and shape estimation). `https://www.youtube.com/watch?v=rfLxcTujc6M`.

[3] M. Andriluka, U. Iqbal, E. Ensafutdinov, L. Pishchulin, A. Milan, J. Gall, and S. B. PoseTrack: A benchmark for human pose estimation and tracking. In *CVPR*, 2018.

[4] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. Scape: shape completion and animation of people. *ACM Trans. Graph*, 24:408–416, 2005.

[5] A. Balan and M. J. Black. The naked truth: Estimating body shape under clothing,. In *European Conf. on Computer Vision, ECCV*, volume 5304 of *LNCS*, pages 15–29, Marseilles, France, Oct. 2008. Springer-Verlag.

[6] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image, 2016.

[7] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-to-end object detection with transformers, 2020.

[8] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling, 2014.

[9] R. Dabral, A. Mundhada, U. Kusupati, S. Afaque, A. Sharma, and A. Jain. Learning 3d human pose from structure and motion, 2018.

[10] M. Z. eiyu Zhang and K. Derpanis. From actemes to action: A strongly-supervised representation for detailed action understanding. In *International Conference on Computer Vision (ICCV)*, 2013.

[11] Grauman, Shakhnarovich, and Darrell. Inferring 3d structure with a statistical image-based shape model. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pages 641–647 vol.1, 2003.

[12] R. A. Guler and I. Kokkinos. Holopose: Holistic 3d human reconstruction in-the-wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[13] K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks, 2016.

[14] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, Nov. 1997.

[15] D. Hogg. Model-based vision: a program to see a walking person. *Image and Vision Computing*, 1(1):5 – 20, 1983.

[16] M. R. I. Hossain and J. J. Little. Exploiting temporal information for 3d human pose estimation. *Lecture Notes in Computer Science*, page 69–86, 2018.

[17] Y. Huang, F. Bogo, C. Lassner, A. Kanazawa, P. V. Gehler, I. Akhter, and M. J. Black. Towards accurate markerless human shape and pose estimation over time, 2018.

[18] M. I. Jordan. Serial order: A parallel, distributed processing approach. In J. L. Elman and D. E. Rumelhart, editors, *Advances in Connectionist Theory: Speech*. Erlbaum, Hillsdale, NJ, 1989.

[19] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik. End-to-end recovery of human shape and pose, 2018.

[20] A. Kanazawa, J. Y. Zhang, P. Felsen, and J. Malik. Learning 3d human dynamics from video. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

[21] M. Kocabas, N. Athanasiou, and M. J. Black. Vibe: Video inference for human body pose and shape estimation, 2020.

[22] M. Kocabas, S. Karagoz, and E. Akbas. Self-supervised learning of 3d human pose using multi-view geometry, 2019.

[23] N. Kolotouros, G. Pavlakos, M. J. Black, and K. Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop, 2019.

[24] C. Lassner, J. Romero, M. Kiefel, F. Bogo, M. J. Black, and P. V. Gehler. Unite the people: Closing the loop between 3d and 2d human representations, 2017.

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

[26] X. Luo, J.-B. Huang, R. Szeliski, K. Matzen, and J. Kopf. Consistent video depth estimation, 2020.

[27] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black. AMASS: Archive of motion capture as surface shapes. In *International Conference on Computer Vision*, pages 5442–5451, Oct. 2019.

[28] D. Mehta, H. Rhodin, D. Casas, P. Fua, O. Sotnychenko, W. Xu, and C. Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *3D Vision (3DV), 2017 Fifth International Conference on*. IEEE, 2017.

[29] D. Mehta, O. Sotnychenko, F. Mueller, W. Xu, S. Sridhar, G. Pons-Moll, and C. Theobalt. Single-shot multi-person 3d pose estimation from monocular rgb, 2018.

[30] D. Mehta, S. Sridhar, O. Sotnychenko, H. Rhodin, M. Shafiei, H.-P. Seidel, W. Xu, D. Casas, and C. Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. volume 36, 2017.

[31] M. Omran, C. Lassner, G. Pons-Moll, P. V. Gehler, and B. Schiele. Neural body fitting: Unifying deep learning and model-based human pose and shape estimation, 2018.

[32] D. Ormoneit, H. Sidenbladh, M. Black, and T. Hastie. Learning and tracking cyclic human motion. In T. Leen, T. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems*, volume 13, pages 894–900. MIT Press, 2001.

[33] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. A. Osman, D. Tzionas, and M. J. Black. Expressive body capture: 3d hands, face, and body from a single image, 2019.

[34] G. Pavlakos, L. Zhu, X. Zhou, and K. Daniilidis. Learning to estimate 3d human pose and shape from a single color image, 2018.

[35] B. Sapp and B. Taskar. Modec: Multimodal decomposable models for human pose estimation. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3674–3681, 2013.

[36] L. Sigal, A. Balan, and M. J. Black. Combined discriminative and generative articulated pose and non-rigid shape estimation. In *Advances in Neural Information Processing Systems 20, NIPS-2007*, page 1337–1344. MIT Press, 2008.

[37] H.-Y. F. Tung, H.-W. Tung, E. Yumer, and K. Fragkiadaki. Self-supervised learning of motion capture, 2017.

[38] R. Urtasun, D. J. Fleet, and P. Fua. 3d people tracking with gaussian process dynamical models. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 1, pages 238–245, 2006.

[39] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need, 2017.

[40] I. B. Vince Tan and R. Cipolla. Indirect deep structured learning for 3d human body shape and pose prediction. In G. B. Tae-Kyun Kim, Stefanos Zafeiriou and K. Mikolajczyk, editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 15.1–15.11. BMVA Press, September 2017.

[41] T. von Marcard, R. Henschel, M. Black, B. Rosenhahn, and G. Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *European Conference on Computer Vision (ECCV)*, sep 2018.

[42] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li. On the continuity of rotation representations in neural networks, 2020.