Human Meshes in Olympic Sports

Nithin Chalapathi
UC Berkeley - CS 294-26 - Final Project Report
SID: 3032738412
nithinc@berkeley.edu

Abstract

The task of generation human meshes on static images and video data has made tremendous progress. However, many state of the art methods struggle on complicated videos with multiple occlusions, multiple people, and fast motion. In this work, we generate 200 short video clips from the 2018 and 2020 Olympics and evaluate VIBE and HMMR models on each clip. Our results are public in a Box Repository. Finally, we attempt to use the motion discriminator from VIBE to match pose tracks to generate a smooth track. Our method provides sub-par results and we discuss future directions.

1. Introduction

Meshes are a common tool used in computer graphics. MoCap (Motion Capture) allows us to generate meshes of humans provided that the MoCap sensors are used. However, MoCap is a costly to set up, record, and requires a controlled environment. Naturally, one question that arises is the automatic generation of a human mesh from videos; instead of using a costly MoCap environment, is there a way to infer human meshes directly from raw videos? Two of the most popular methods are HMMR [5] and VIBE [6]. However, these models struggle with high velocity motion and motion that hasn't been seen before. 3D Poses in the Wild Dataset [7] is a large dataset with various videos of natural human motion. In this project, we provide two contributions:

- We generate 200 clips of Olympians performing in the 2018 and 2020 winter and summer Olympics.
- We compare how HMMR and VIBE perform on our Olympians dataset, examining various failure modes.
- Finally, we use the motion discriminator from VIBE to attempt pose track matching.

2. Related Work

We first cover the Skinned Multi-Person Linear Model, Pose Detection Models, and the current state of the art methods in human mesh reconstruction (HMMR and VIBE).

2.1. SMPL

The Skinned Multi-Person Linear Model (SMPL) is a learned mesh model consisting of $N = 6890$ vertices in 3 dimensions. The model $\mathcal{M}(\beta, \theta)$ is a differentiable function of $\beta \in \mathbb{R}^{10}$ and $\theta \in \mathbb{R}^{72}$. β represent the shape parameter and θ represents the pose parameter. $\mathcal{M}$ is a fixed trained model that we make no changes to. We instead treat it as a black box to generate meshes for visualizations; both VIBE and HMMR predict θ and β.

2.2. Pose Detection

There are two commonly used pose detection libraries, namly AlphaPose[1] [2] and OpenPose[2] [1]

AlphaPose. Built on top of Regional Multi-person Pose Estimation (RMPE), AlphaPose is a real time multi-person pose estimation library. RMPE uses a top down approach to generate each pose. RMPE runs a person object detector (e.g. Mask R-CNN [3]). The cropped persons are then fed into spatial transformer network (STN) [4], a Single Person Pose Estimator, and finally a spatial de-transformer network. Finally they perform a parametric pose non-maximal suppression.

OpenPose. They introduce Part Affinity Fields (PAF) which are 2D vectors that encode the location and orientation of limbs. OpenPose is a bottom up approach where they generate confidence maps for each limb. They then use PAFs to associate limbs together. Finally, both the confidence maps and PAFs are sent to a greedy bipartite matching algorithm. The bipartite matching associates body parts; using this information, OpenPose constructs the fully body pose.

[1] https://github.com/MVIG-SJTU/AlphaPose

[2] https://github.com/CMU-Perceptual-Computing-Lab/openpose

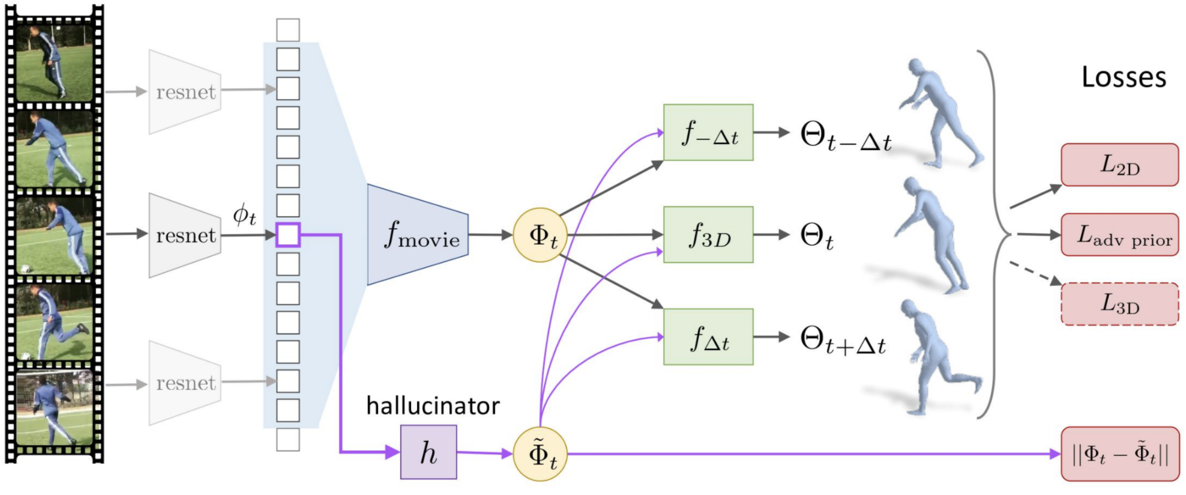

Figure 1. HMMR Pipeline

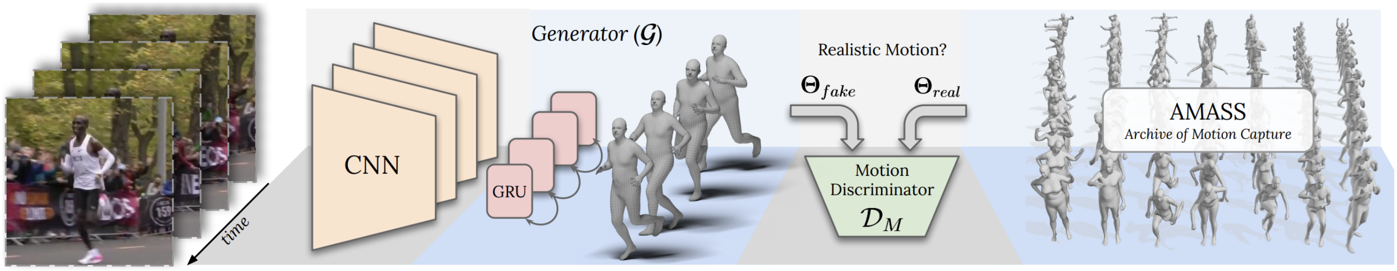

Figure 2. VIBE Pipeline

2.3. Mesh Generation Models

There are two human mesh generation models we look at: HMMR [5] and VIBE [6]. Both methods operate on videos.

HMMR. In HMMR, they attempt to explicitly learn the dynamics of human motion. They do this by generating a motion encoding, Φ_t by running a variant of ResNet-50. The input to the encoder is a sliding window of frames. From Φ_t, the SMPL parameters of the current, past, and next frame is predicted. Please see figure 1 for an illustration from [5]

VIBE. VIBE is a generator-discriminator adversarial architecture. The generator is a gated recurrent neural network. The discriminator takes in an input of SMPL parameters and outputs whether the sequence is a valid form of motion. In order to train the discriminator, they use AMASS, a dataset of MoCap instances. While HMMR uses a a sliding window when predicting motion, VIBE uses a self-attention mechanism. Specifically, the discriminator uses attention to on the input sequence. Once again, a visual representation of their pipeline is in figure 2

3. Olympics Dataset Generation

In order to evaluate HMMR and VIBE, we also generate a database with 200 clips from the from 2018 and 2020 Olympics. We specifically focus on rock climbing, snowboarding, skateboarding, and figure skating. For each clip, we run both OpenPose and AlphaPose to ensure that at least one method is able to detect a valid pose track. We run both since HMMR uses AlphaPose and VIBE uses OpenPose; by making sure at least one method can recover a track, we try to avoid biasing the results.

To build the clip extractor, we use PyTube[3] to download various YouTube clips. Each clip is then played in a TKinter[4] window using OpenCV2. Figure 3 has our interface. The person clipping the video selects a start and end time stamp. Multiple clips can be extracted from the same video.

For each of the sports, we extract 50 clips; each clip is roughly 7-10 seconds. We provide a public Box link to view the videos[5].

[3] https://github.com/pytube/pytube

[4] https://docs.python.org/3/library/tkinter.html

[5] https://app.box.com/s/qprh57jlygipanue4rcjgfdbssm6j398

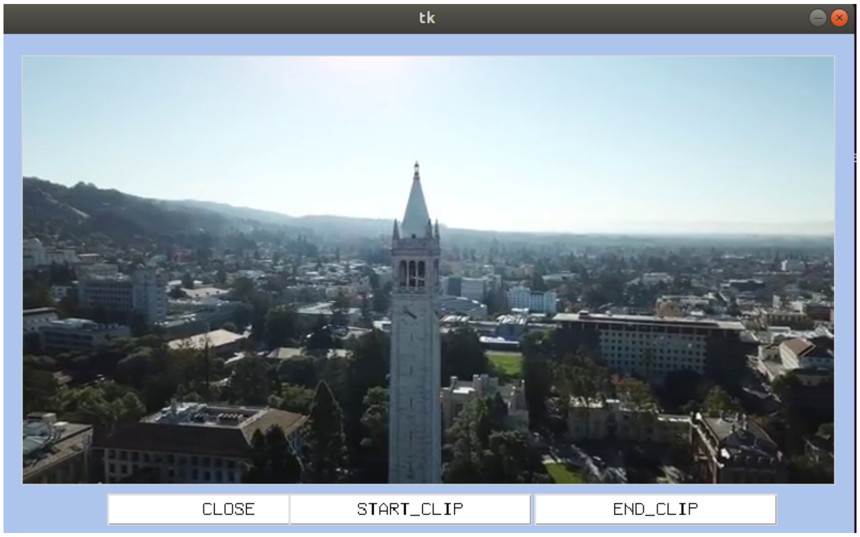

Figure 3. Our clip creator. The program automatically searches YouTube and exports clips to the database.

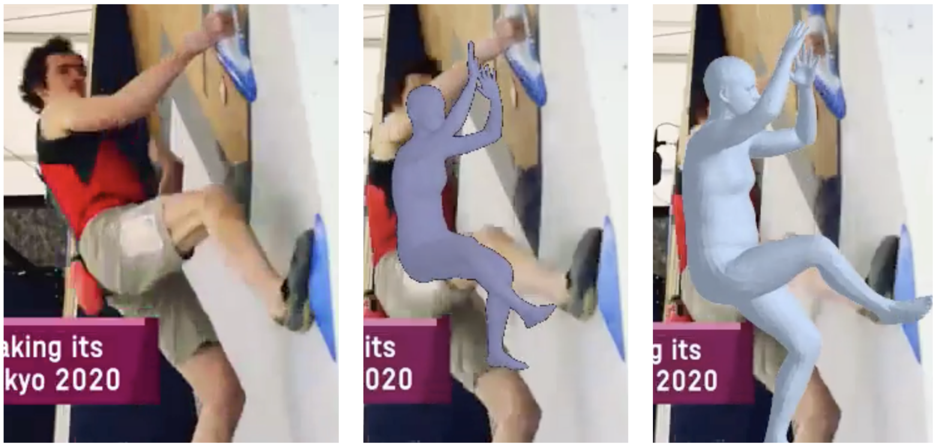

Figure 4. Climbing results. Left: Raw Frame Middle: VIBE Result Right: HMMR Result

4. Results

For the report, we only select one frame for each of the four sports. Please see the project website for animated gifs and links to all the results.

4.1. Discriminator Based Track Matching

One common problem we noticed was OpenPose has trouble keeping continuity between tracks that should be the same. While VIBE visually performs better on a per frame basis, the discontinuity in tracks negatively affects the final output. In order to attempt to tackle this, we introduce discriminator based track matching.

For tracks that are within 30 (1 second) frames, concatenate the predicted motion sequences from both. After, pass the result into the motion discriminator from VIBE. If the probability the combined sequence is “real” is higher than each of the individual tracks. Finally, in order to ensure that tracks are spatially similar, for the last bounding box of track 1 and the first bounding box of track 2, we compute an intersection over union score. If all of the conditions are met, the tracks are combined and VIBE is rerun on the track. For frames in between the tracks, the bounding box is linearly interpolated. Please see our project website for the comparison GIF.

5. Discussion on Results

There are many places where HMMR and VIBE fail on Olympic videos. We’ll discuss each failure mode.

Unnatural Motion. In a lot of Olympic sports, humans move in unnatural ways. VIBE and HMMR are unable to

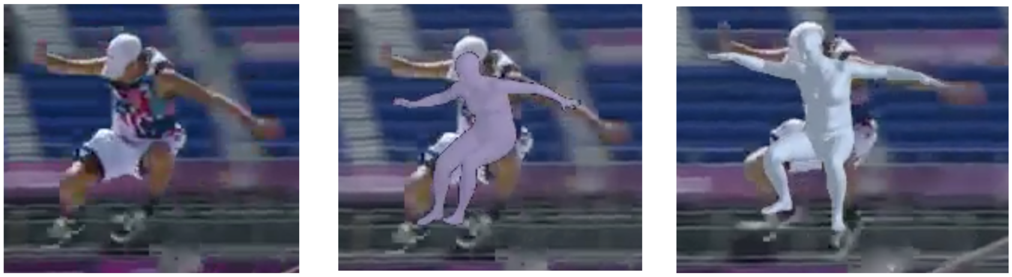

Figure 5. Skateboarding results. Left: Raw Frame Middle: VIBE Result Right: HMMR Result

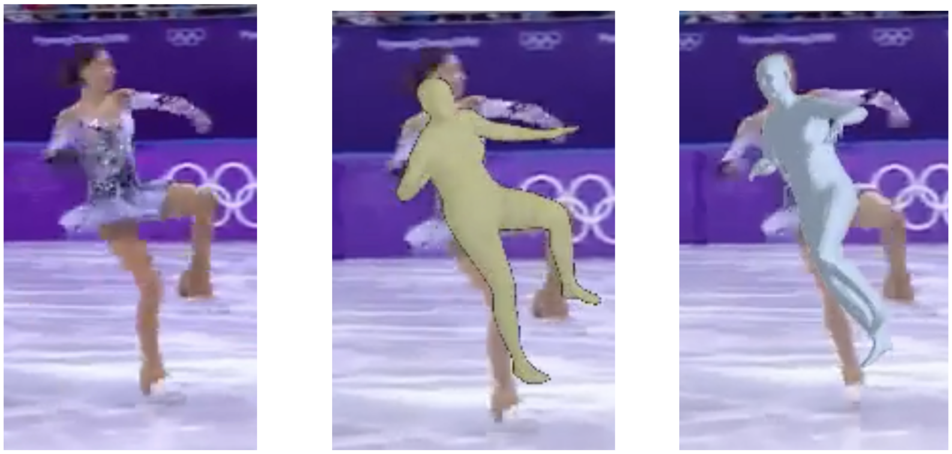

Figure 6. Figure Skating results. Left: Raw Frame Middle: VIBE Result Right: HMMR Result

recover the proper mesh dynamics since the data is too far out of the domain. For example, someone riding a skateboard is not a common pose in everyday life.

Climbing detection. One area where VIBE really struggled was the climbing videos, specifically the speed climbing race. It looks as though OpenPose is unable to recover any poses.

Scale. In some of the videos, only half of the human would inside of the frame. However, both VIBE and HMMR try to force a full human mesh into the half human.

6. Conclusion and Future Work

In this class project, we generate a dataset of Olympic clips. We demonstrate a few failure modes and release all of our results. We also try to solve the continuity issue with OpenPose with mixed results.

In the future, it would be very interesting to see if the clip extract could be automated. It is already very quick to pull the clips in terms of human effort. However, it would be best if the program could auto-detect clips.

A natural extension of this project is to evaluate context based human mesh generation systems. It seems some new human mesh generation systems use the context of the entire frame in order to more accurately predict the motion.

References

[1] Z. Cao, G. H. Martinez, T. Simon, S. Wei, and Y. A. Sheikh. OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[2] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu. RMPE: Regional Multi-person Pose Estimation, 2018. _eprint: 1612.00137.

[3] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN, 2018. _eprint: 1703.06870.

[4] M. Jaderberg, K. Simonyan, A. Zisserman, and k. kavukcuoglu. Spatial Transformer Networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[5] A. Kanazawa, J. Y. Zhang, P. Felsen, and J. Malik. Learning 3D Human Dynamics from Video. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

[6] M. Kocabas, N. Athanasiou, and M. J. Black. VIBE: Video Inference for Human Body Pose and Shape Estimation, 2020. _eprint: 1912.05656.

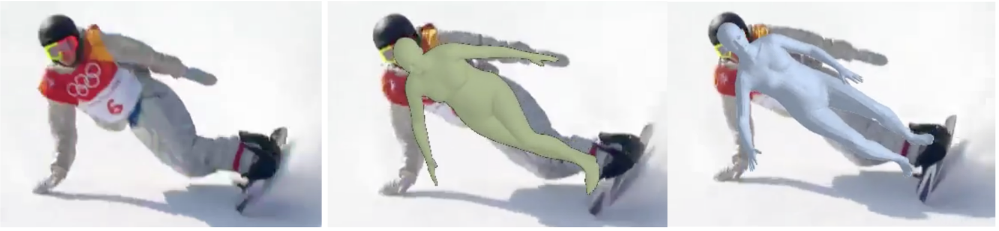

Figure 7. Snowboarding results. Left: Raw Frame Middle: VIBE Result Right: HMMR Result

[7] 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)*, Sept. 2018.