Improving Articulated Pose Tracking and Contact Force Estimation for Qualitative Assessment of Human Actions

by

Nathan Louis

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical and Computer Engineering)
in The University of Michigan
2024

Doctoral Committee:

Professor Jason J. Corso, Co-Chair
Assistant Professor Andrew Owens, Co-Chair
Professor Donald S. Likosky
Dr. Daniel P. Nicolella, Southwest Research Institute

Nathan Louis

natlouis@umich.edu

ORCID iD: 0000-0003-4502-6012

To my faith, loved ones, and friends

ACKNOWLEDGEMENTS

First and foremost, I extend my gratitude to Prof. Jason J. Corso for granting me with the invaluable opportunity and independence to engage in computer vision research. And I am deeply grateful to my esteemed committee members, Prof. Andrew Owens, Prof. Donald S. Likosky, and Dr. Daniel P. Nicolella, for all of their unwavering support, patience, and invaluable feedback.

My journey to Michigan and time spent here would have been impossible without the collective support of individuals I have met throughout my entire academic journey. I would like to thank my collaborators at Southwest Research Institute and the Varsity Surgery team here at Michigan for their longstanding collaborative efforts. Thank you to all former and current members of the COG Lab, specifically Dr. Luowei Zhou, Dr. Madan Ravi Ganesh, and Dr. Stephan J. Lemmer for their consistent feedback and fruitful discussions.

I am deeply grateful to my family and friends, scattered across Georgia, New Jersey, Haiti, and many other places, for their support during these transformative years. I want to express a very special thank you to June, for always comforting and supporting me. I love you and I cherish our time together. Lastly, I want to thank my parents and my brothers for their steadfast support, lifting me through the highs and lows and championing my journey from its inception.

TABLE OF CONTENTS

Dedication . . . . . **ii**

Acknowledgments . . . . . **iii**

List of Tables . . . . . **vii**

List of Figures . . . . . **ix**

List of Appendices . . . . . **xii**

Abstract . . . . . **xiii**

Chapter

1 Introduction **1**

1.1 Motivation 1
1.2 Background 3
1.2.1 Pose Estimation and Tracking 3
1.2.2 Ground Reaction Force Prediction 4
1.2.3 Technical Skill Assessment 5
1.3 Contributions 6
1.3.1 Improving Articulated Hand Pose Tracking 6
1.3.2 Novel Approaches for Ground Reaction Force Prediction 6
1.3.3 Introducing Physical Simulation as a Metric for Physical Plausibility . . 7
1.3.4 Assessing Technical Skill from Open Surgery Procedures 7
1.4 Thesis Statement 8

2 Related Work **9**

2.1 Pose Estimation and Tracking 9
2D Human Pose 9
3D Human Pose 10
Physics-aware 3D Human Pose 10
Hand Pose 11
Instrument Pose 11
2.2 Physical Force and Load Analysis 12
Hardware Solutions 12
Video Solutions 13
2.3 Human Skill Assessment 13

Sports Assessment . . . 13
Generic Household Tasks . . . 14
Surgical Skill Assessment . . . 14
2.4 Semi-supervised Learning . . . 16
2.5 Operating on Temporal Sequences . . . 17
2.6 Physical Simulation and Optimization . . . 18

3 Articulated Hand Pose Tracking . . . 19

3.1 Introduction . . . 19
3.2 Dataset . . . 20
3.3 Method . . . 22
3.3.1 Hand Pose Estimation in Images . . . 23
3.3.2 Hand Pose Estimation in Videos . . . 23
3.3.3 Matching Strategies for Tracking . . . 24
3.4 Experiments and Evaluation . . . 25
3.4.1 Implementation Details . . . 25
3.4.2 Detection Performance . . . 25
3.4.3 Tracking Performance . . . 26
3.4.4 Ablation Analysis . . . 27
3.4.5 Evaluation on Human Pose . . . 28
3.5 Conclusion . . . 29

4 Ground Reaction Force Estimation via Multi-task Learning . . . 31

4.1 Introduction . . . 31
4.2 Data . . . 33
4.2.1 ForcePose Dataset . . . 33
4.2.2 LAAS Parkour Dataset . . . 34
4.3 Method . . . 34
4.3.1 Losses and Metrics . . . 35
4.3.2 Predicting Ground Reaction Forces . . . 35
Transformer Encoder . . . 36
Pre-training and Multi-task Learning . . . 37
4.4 Experiment Details . . . 38
4.5 Results . . . 39
4.5.1 *ForcePose* . . . 39
2D-to-3D subtask . . . 41
Zero-shot learning . . . 41
4.5.2 LAAS Parkour . . . 42
4.6 Extension to Experiments . . . 43
4.6.1 ForcePose version 1.1 . . . 43
4.6.2 Data augmentation for generalization . . . 43
4.7 Conclusion . . . 45

5 Physical Plausibility as a Metric for 3D Human Pose Estimation . . . 46

5.1 Introduction . . . 46

5.2 Method . . . 48
5.2.1 Kinematic Initialization . . . 48
5.2.2 Trajectory Optimization . . . 49
5.2.3 Simulation-based Metrics . . . 51
Metric 1: COM Trajectory Distance . . . 51
Metric 2: Dynamic Stability . . . 52
5.3 Experiments . . . 52
5.3.1 Evaluation Dataset . . . 52
5.3.2 Evaluation Models . . . 53
5.3.3 Evaluation Metrics . . . 53
5.4 Results . . . 55
5.5 Impact of Toe and Heel Joints . . . 59
5.6 Additional Qualitative Examples . . . 59
5.7 Discussion . . . 60

6 Contrastive Learning for Video-Based Skill Assessment in Open Cardiac Surgery . . . 64

6.1 Introduction . . . 64
6.2 Surgical Skill Assessment with Contrastive Learning . . . 65
6.2.1 Clip-level Contrastive Learning . . . 66
6.2.2 Feature Extraction . . . 67
6.3 Experimental Setup and Results . . . 68
6.3.1 Data . . . 68
6.3.2 Implementation Details . . . 69
6.3.3 Surgical Skill Classification Results . . . 69
6.3.3.1 Analysis of Motion Characteristics . . . 70
6.4 Discussion . . . 70

7 Conclusion and Future Directions . . . 73

7.1 Conclusion . . . 73
7.2 Limitations . . . 74
Failure at detection . . . 74
Extrapolation of contact forces . . . 74
Computational Efficiency and Accurate Simulation Modeling . . . 74
7.3 Future Work . . . 75
Constraining force embeddings with self-supervision . . . 75
Informing force embeddings with physical simulation . . . 75
Enhancing latent skill representations . . . 76

Appendices . . . 77

Bibliography . . . 89

LIST OF TABLES

2.1 Detailed description of OSATS categories developed by Martin *et al*. 15

3.1 We compare our proposed dataset to other existing hand pose datasets. Our data supports multiple object instances, along with tracking, in each clip. 21
3.2 Mean Average Precision (mAP). Performance is averaged across all folds 26
3.3 We optimize for the Multiple Object Tracking Accuracy (MOTA), each performance metric is averaged across all validation folds 26
3.4 MOTA performance between matching strategies, averaged across all folds. Each row is optimized for highest MOTA performance. Matching strategies share the same base model, so it is possible for them to share the same mAP score. 29
3.5 Ablation analysis using IoU matching strategy ($\delta = 1$). NC = No convolutional feature map, NA = No attention mechanism. 29
3.6 Effect of δ. Each model is trained with a separate δ value 30

4.1 Average Sequence Losses and mean k-peaks on the *ForcePose* dataset, measured in Newtons 38
4.2 Average Sequence Losses on the *ForcePose* dataset. We compare results from transformers using 2D-to-3D HPE as a subtask 41
4.3 LAAS Parkour Dataset. Estimation errors of forces (in Newtons). Each subject has an assumed mass of 74.6 kg 41
4.4 Zero-shot learning, RMSE measured in Newtons 42
4.5 Updated results on *ForcePose* version 1.1. 43

5.1 3D human pose estimation methods used for comparison and published scores on the Human3.6M dataset. S = Single image, MV = Multiview. 50
5.2 We share results on our validation subset of the Human3.6M dataset. We compare standard 3D pose evaluation metrics as well as physical plausibility metrics. 55
5.3 We break down the per-class performance for the dynamic stability (Dyn_{100}) across methods. We underlined the classes with the most spatial displacement in the world plane. 56
5.4 We show results on our validation subset on Human3.6M dataset. Baseline-17 removes the toe and heel keypoints. The results here are comparable to Table 5.2. 59
5.5 Here we show the per-class performance for the Dyn_{100} metric on Baseline-17. Baseline-17 removes the toe and heel keypoints. The results here are comparable to Table 5.3. . 59
5.6 The detected joint format used for each method, GT 3D and PoseFormer use the same exact joints. If the pelvis joint is not detected, it is estimated from the left and right hip joints. 63

6.1 Skill classification results on VTS (left) and COSSA(right) datasets with comparison to baselines. Averaged across 5-folds. 70
6.2 Corresponding summary statistics computed from samples in Figure 6.2. Measures below are shown as Left Hand(s) / Right Hand(s). 71

A.1 Network architecture details on the branches in our model. Each layer shows the number of input features, output features, kernel size, and stride. 78
A.2 Object Detection Metrics of Hand Detections across detection thresholds 79
A.3 MOTA performance between matching strategies, averaged across all folds. Each row is optimized for highest MOTA performance. Matching strategies share the same base model, so it is possible for them to share the same mAP score. Parenthesis show added optical flow. 80

B.1 Detailed skill classification results on the VTS dataset. 83
B.2 Detailed skill classification on the **COSSA** dataset. 83
B.3 MAE for OSATS prediction on **COSSA**. Averaged cross 5-folds. (Lower is better). . . 84
B.4 Distribution of OSATS scores ($n = 14$). 84

C.1 Sweep input receptive field 86
C.2 Zero-shot learning, sweep input receptive field (single run) 87
C.3 Sweep threshold, T, in gated-MSE loss 88

LIST OF FIGURES

1.1 An example of monitoring the quality of a barbell squat, we can compare (a) hardware-based solutions and (b) video-based solutions. Video-based solutions can leverage deep learning methods to extract additional information. Image borrowed from 2

1.2 Examples of pose estimation for (a) Human pose , (b) hand pose, (c) bird pose from the CUB-200-2011 dataset , and (d) car pose 4

2.1 We show various forms of hand pose data in the literature. With our video-based *SurgicalHands* dataset in (a), synthetic and single images in (b) and (c) , and video data with one or two hands in controlled environment shown in (d) and (d) 12

2.2 We show the range of data supervision available in machine learning. On the left, we have fully-supervised data which provides dense annotations for all samples. On the right, we have unsupervised data which only includes the raw data itself with no labels. While in the middle, semi-supervision is a combination of the two where labeled examples can inform us about unlabeled examples. The bulleted terms depict applicable machine learning applications for each label of supervision. Figure is inspired by . . . 17

3.1 On the left, a method only performing frame-wise independent predictions may miss out on properly localizing joints, while on the right, temporally passing past predictions from previous frames improves the network's localization. 20

3.2 The baseline generates a heatmap, $\hat{\mathcal{H}}'_t$, for each detection using a pose estimation network. In our model, we provide additional information by incorporating a heatmap prior from $t-\delta$. Concatenating the image features at t with $\hat{\mathcal{H}}_{t-\delta}$, we pass this through our attention mechanism to produce a weighted heatmap prior, $\hat{\mathcal{H}}'_{t-\delta}$. Both $\hat{\mathcal{H}}'_t$ and $\hat{\mathcal{H}}'_{t-\delta}$ are concatenated and passed through the fusing module, using context from both heatmaps to produce the final articulated hand pose. The initial and final heatmaps represent real outputs from the network, while the heatmap prior (during training) shows ground truth at $t-\delta$) 21

3.3 Statistics on visibility of each joint. The least visible joints belong to the 4^{th} and 5^{th} digits, this is expected as they are underutilized in most surgical actions. 22

3.4 We show samples from our annotations. Each hand is labeled with a bounding box, handedness, tracking id, and visibility of joints. 23

3.5 We show a qualitative comparison between the baseline model and our method. We note a higher recall and consistency between frames, as shown for the hand to the left. Even when the pinky finger is not visible, the past predictions reinforces those joint locations. 27

3.6 We show qualitative samples of frames from the best performing (top row) and lower performing (bottom row) videos. (Best viewed in color). 28

3.7 Optimized for maximum Multiple Object Tracking Accuracy (MOTA) score, we show the top performing models on PoseTrack18. Consistent with our earlier findings, our model maintains a higher mAP for comparable MOTA scores. 30

4.1 Standard practice for modeling human motion with forces requires physical reflective markers to capture kinematics and force plates to measure GRFs. In contrast, we propose a video-only approach that yields comparable performance yet does not require any physical apparatus to predict forces. Here the subject is performing a jumping movement on two force plates, measured forces are shown on the right. (Reflective markers are highlighted for visibility) 32

4.2 On the left, we show the (eight) camera views used from each trial and on the right, the (six) movements captured in the *ForcePose* dataset. (Single Leg Squat = SLS, Counter Movement Jump = CMJ, Single Leg Jump = SLJ). 34

4.3 We show a brief overview of (a) the vanilla transformer encoder and the Spatial-Temporal architecture (b)-(c) we used in our experiments for this work. This is composed of (b) a spatial encoder and (c) a temporal encoder. The input is a sequence of 2D poses and the output is a prediction of the 3D pose and corresponding GRF at the center of that sequence. 36

4.4 We compare the net GRF outputs on the Single Leg Jump (R) (top row) and Squat (bottom row) trained using the (a) MSE Loss and (b) gated-MSE. The force plates are shown in green and predicted forces are shown in blue. We show much smaller differences across the mean k-peaks when compared to the ground truth force plates. Horizontal lines mark the detected peaks in each plot, for brevity we only show the top-3. 40

4.5 We demonstrate qualitative contact force results (b)-(c) on the *S9 - Walking 1* motion from the Human3.6m dataset (a). In (b), we see predicted begin to attenuate as subject moves away from the starting position or rotates around the y-axis. While in (c), including translation and rotation augmentations on the poses produces a plausible force magnitudes. 44

5.1 The top row shows a reprojected 3D prediction on the Human3.6M dataset (*S9 - Directions 1*). While the error is relatively low on current metrics (MPJPE = 49.0mm, FS = 0%, GP = 0.59mm), with our physical plausibility simulator (bottom row), we see that a slight unnatural lean eventually causes a loss of balance. 48

5.2 To analyze the physical plausibility of 3D human pose estimates, we first extract a 3D skeletal pose from the output of a 3D HPE method. We then initialize kinematics through the estimation of reference joint angles, floor height, and normalizing joint segment lengths. We apply the joint angles directly to a simulated body within a simulated environment and optimize the joint angles to imitate the reference motion under simulator constraints. We measure the approximate plausibility of this optimized output by analyzing COM trajectory distance, dynamic stability, and contact forces. . . 49

5.3 2D re-projected (top row) and corresponding simulated body (bottom row) from NeuralPhysCap . While this example displays a low MPJPE-2D=50.0mm, we measure dynamic stability to be adequate, $\text{Dyn}_{100} = 71.9$. 54

5.4 For the *S11 - WalkTogether* sequence, we show 3D prediction and simulated output results between (a) Baseline (b) PoseFormer (c) NeuralPhysCap . Inaccurate camera and ground plane assumptions (shown with the red arrows) in (c) causes the motion to fail early on as the simulated body tries to step through the ground plane. The right column shows the 2D re-projected predictions on the final frame. 57
5.5 External contact force results on *S11 - WalkTogether 1*. The row, (a), represents the estimated contact using GT 3D keypoints. The remaining rows are results from (b) Baseline (c) PoseFormer (d) NeuralPhysCap 58
5.6 We show results on PoseFormerfor the *S11 - Purchases 1* sequence. 61
5.7 We show results on PoseFormer for the *S9 - WalkDog 1* sequence. 61
5.8 We show results on our baseline for the *S11 - Walking 1* sequence. 62
5.9 We show results on our baseline for the *S9 - Waiting 1* sequence. 62

6.1 For unsupervised contrastive pretraining, we temporally sample clip-level features, I_i, from each video, which could be space-time patches, learned features or those tuned to the domain. We further embed them into a latent representation, z_i, which is obtained through a constrastive loss-based pretraining. Green lines indicate positive pairs (collected from the same video v_i) and red lines indicate negative pairs (collected from different videos). After pretraining, we append the final layer of the contrastively-trained feature encoder and fine-tune for the actual skill classification task. 66
6.2 We show correctly classified results on VTS (a)-(d) and COSSA (e)-(h), using $T = 10$ object tracks. Each examples shows a detected frame and corresponding hand tracks. We note that novice motions traverse a wider area while expert motions are more localized and precise. Purple tracks denote detected right hands and green tracks are detected left hands. 71

A.1 A detailed overview of our model during evaluation on videos. Like training, newly introduced objects are accompanied with a blank, zero heatmap prior. We use two post-processing steps to filter out improbable priors when assigning matches. Each image crop is padded and centered on the detected hand. 78

B.1 Video lengths between novices and experts are easily separable on VTS (a) than they are on **COSSA**. This bias can prevent learning-based methods from discovering meaningful characteristics correlated with technical skill. 81
B.2 We compare full video statistics (a) and clip-level statistics (b) for the left and right hands between novices and experts. We note a higher total distance bias in full videos which we significantly reduce with clip-level sampling, while maintaining notable differences in average and maximum tracking velocity. 82
B.3 Visualization of spatio-temporal attention from the transformer. 85
B.4 Visualization of spatio-temporal attention from the transformer. 85

LIST OF APPENDICES

A Articulated Hand Pose Tracking **77**

B Contrastive Learning for Video-Based Skill Assessment in Open Cardiac Surgery . . **81**

C Ground Reaction Force Estimation via Multi-task Learning **86**

ABSTRACT

Using video to automate human performance metrics or skill analysis is an important but under-explored task. Currently, measuring the quality of an action can be highly subjective, where even assessments from experts are affected by bias and inter-rater reliability. In contrast, Computer vision and AI have the potential to provide real-time non-intrusive solutions with increased objectivity, scalability, and repeatability across various domains. From video alone, we can automatically provide supplemental objective scoring of Olympic sports, evaluate the technical skill of surgeons for training purposes, or monitor the physical rehabilitation progress of a patient. Today we solve these problems with supervised learning, obtaining features that represent high correlation with our desired point of analysis. Supervised learning is powerful, data-driven, and sometimes the best available option. However alone, it may be sub-optimal in the presence of scarce data and insufficient when needed to generalize to varying conditions or to truly understand the target task.

In this dissertation, the bases of our human analysis understanding are skeletal poses, namely hand poses and full body poses. For articulated hand poses, we improve tracking using our *CondPose* network to integrate prior detection confidences and encourage tracking consistency. While for human poses, we propose two physical simulation-based metrics for evaluating physical plausibility and perform external force estimation through predicted ground reaction forces (GRFs). However, in the human analysis domain, collecting and annotating data at the scale of other deep learning tasks is a recurring challenge. This limits our generalizability to different environments, procedures, and motions. We address this by exploring semi-supervised learning methods, such as contrastive pre-training and multi-task learning.

We apply articulated hand pose tracking in the surgical environment for assessing surgical skill. By applying a time-shifted sampling augmentation, we introduce clip-contrastive pre-training on embedded hand features as an unsupervised learning step. We show that this contrastive pre-training improves performance when fine-tuned on surgical skill classification and assessment task. Unlike most prior work, we evaluate on open surgery videos rather than solely simulated environments. Specifically, we use videos of non-laparoscopic, collected through collaboration with the Cardiac Surgery department at Michigan Medicine.

We use full body poses and contact force estimation to bridge the gap between visual observations and the physical world. This physically-grounded component is vital for understanding actions

involving sports or physical rehab where humans interact with their environment. We leverage multi-task learning to perform 2D-to-3D human pose estimation and integrate other abundant sources of motion capture data, without requiring additional force plate supervision. Our experiments shows that this improves GRF estimation on unseen motions.

To address data limitations, we also collect two novel datasets *SurgicalHands* and *ForcePose*. We use *SurgicalHands* in the surgical domain as a multi-instance articulated hand pose tracking dataset. It encompasses a high degree of complexity in appearance and movement, not present in prior datasets. *ForcePose* is a multi-view GRF dataset of tracked human poses and time-synchronized force plates, to our knowledge the largest and most varied of its kind. This dataset serves as a benchmark for mapping human body motion and physical forces, enabling physical grounding of specific actions.

CHAPTER 1

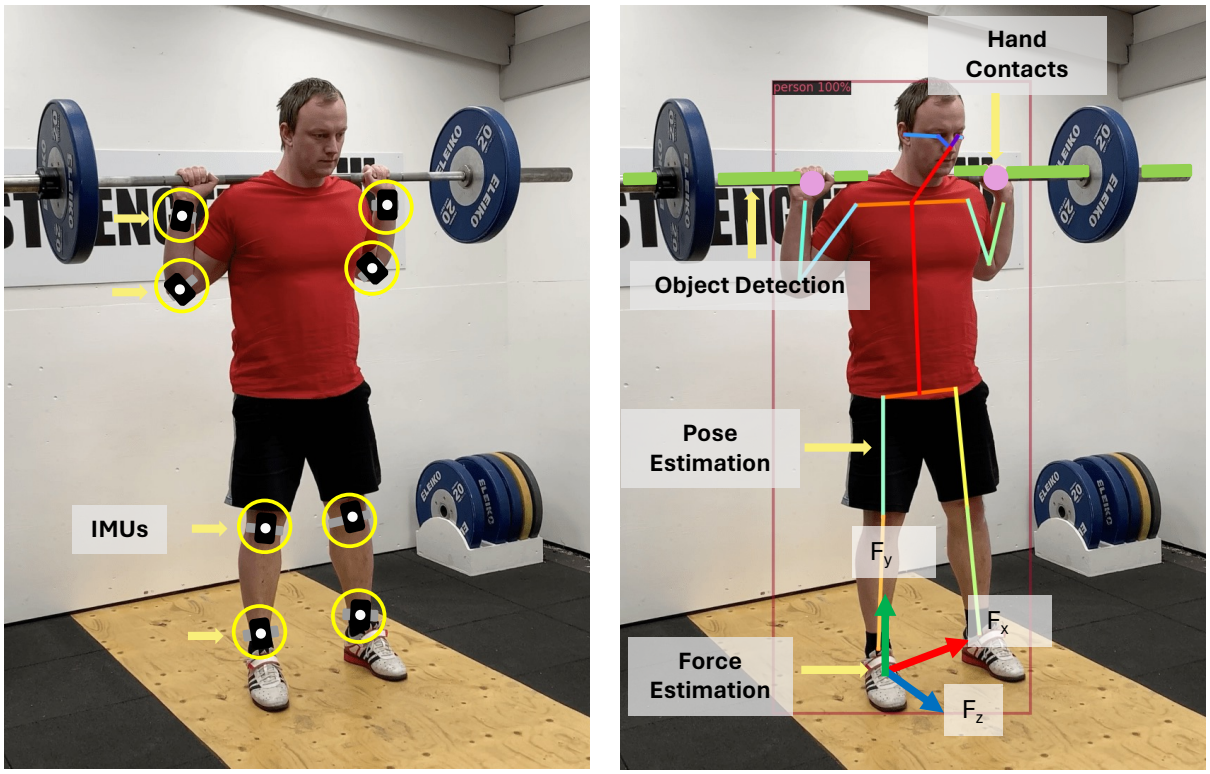
Introduction

1.1 Motivation

Since 2018, the NFL has hosted an annual Big Data Bowl competition [1] to identify metrics that assesses offensive and defensive decision making, while also mitigating risks to protect players. Concurrently, newly-founded start-ups are monitoring the safety of seniors through video for events like fall prevention [2] or tracking human activity to identify areas for improvement in industrial manufacturing processes [3]. These human-centric analysis pursuits are emerging due to the increasing availability of high-quality video collection and the prominence of deep learning algorithms. As computer vision scientists, we seek ways to apply video understanding to this vast amount of data, predominantly through deep learning methods that extract meaning and comprehension.

We identify automated skill assessment as a key element for human-centric analysis, which has vast implications in how we interpret and utilize video data. Today, these technical actions are manually assessed, but they can be increasingly time-demanding, require expert evaluators, and still subject to bias [4], [5]. Existing computer vision and AI research have produced work in this direction with applications in scoring of Olympic events [6]–[8], surgical skill assessment [9]–[13], and proficiency of generic household activities [14], [15]. But there still exists gaps in literature such as lack of physical grounding for full body actions or non-simulated environments for surgical skills, which does not mimic the real-world.

Human skill assessment, when related to human motion, can be used to identify factors in performance or injury prevention of certain physical activities (*e.g.* running and jumping) . From simply watching a video itself, we can roughly infer the action, the amount of force used, the quality of said action, and what corrections could be made. For an exercise such as barbell squats, it may take months of training and personal coaching to adequately learn the correct setup, grip, positioning, and squatting motion. As show in Figure 1.1 (a), modern engineering solutions will suggest hardware such as attached inertial measurement units (IMUs) [16]–[18] to monitor the positions and accelerations of human joints (or exercise equipment) and approximate some forces.



(a) Hardware-based Solutions (b) Video-based Solutions

Figure 1.1: An example of monitoring the quality of a barbell squat, we can compare (a) hardware-based solutions and (b) video-based solutions. Video-based solutions can leverage deep learning methods to extract additional information. Image borrowed from [20].

But this approach is cumbersome and time-consuming, as it requires expertise for proper placement [19] and devices that are not commonly accessible. However, recording video from a camera is much simpler, non-intrusive, and can introduce deep learning for post-processing. In Figure 1.1 (b), using video data, deep learning methods may also demonstrate the flexibility to process other aspects such as external forces, produced by the human, and human-object interaction, examining contact with the barbell. In this dissertation, we limit our study to external contact forces of the human body, rather than object interactions.

In the medical domain, videos recorded in the operating room have potential as lifelong surgical training devices or for projecting surgical outcomes through early detection of complications. Surgical skill assessment can serve as a proxy for understanding these components through extracted visual formation (*e.g.* hand pose, event detection, instrument tracking). However, most existing work focuses on simulated environments [13], [21]–[23], which lack the fidelity of real-world operations or challenges faced by active surgeons. Others use easier-to-acquire videos of robotic-assisted surgeries [9], [10], [12], [24], but these works represent less than 3% percent of major operating

room procedures in the U.S. [25]. On the other hand, in this dissertation we collect and evaluate on operations of live open-surgery videos, particularly in cardiac surgery. These data are essential in the highly complex surgical operating room, where hardware sensors cannot interfere with the procedure.

Given the nuanced nature of skill assessment, current approaches require manual ratings from human annotators with domain expert knowledge. Hence, we cannot rely on extensive annotated datasets found in other deep learning tasks due to severe limitations in this niche space. An approach to overcome these data and generalization limitations is semi-supervised learning [26], [27], leveraging existing data with minimal labeling to implicitly generalize to different domains. Another under-explored aspect is the physical grounding of visual observations, whereas many works only consider pixel-level information. They form physical understanding through the abundance of data instead of laws of physics. In contrast, approximating the physical dynamics of actions allows us to extrapolate to novel movements and events, without overfitting to specific visual features. Addressing these gaps moves us towards the extraction of quantitative metrics, where we seek to answer the key question: How can we automatically assess the technical skill of actions directly from video?

1.2 Background

In this dissertation we focus on the building blocks for analyzing human-centric actions and activities. Our goal is to transform raw video features into physical scene components to facilitate skill assessment. We generalize our problem into three distinct phases: first extracting image-level information, then inferring real-world physical properties, and finally evaluating and interpreting significance of the derived features. Specifically, in terms of applications, we study pose estimation and tracking, ground reaction force prediction, and technical skill estimation.

1.2.1 Pose Estimation and Tracking

Pose estimation is a method for processing an image of a target class and extracting a standardized representation. This first requires a well-defined two-dimensional pose, $\mathbf{P} = \{P_1, P_2, ..., P_J\}$, normally represented using a tree or graph structure [31], where J denotes the number of nodes on the graph. Note that the pose may be extended into three-dimensions, but we describe the 2D case for brevity. For humans this is depicted using joint centers (*e.g.* shoulder, elbow, knee), but may also include other points such as the eyes, ears, and nose, as shown in Figure 1.2 (a). This pose definition also applies to hands, or non-human classes such as cars or birds, also depicted in Figure 1.2 (b)-(d).

Given an input image, $\mathcal{I} \in \mathbb{R}^{H \times W \times 3}$, the output is the two-dimensional target pose $\hat{\mathbf{P}}$, where

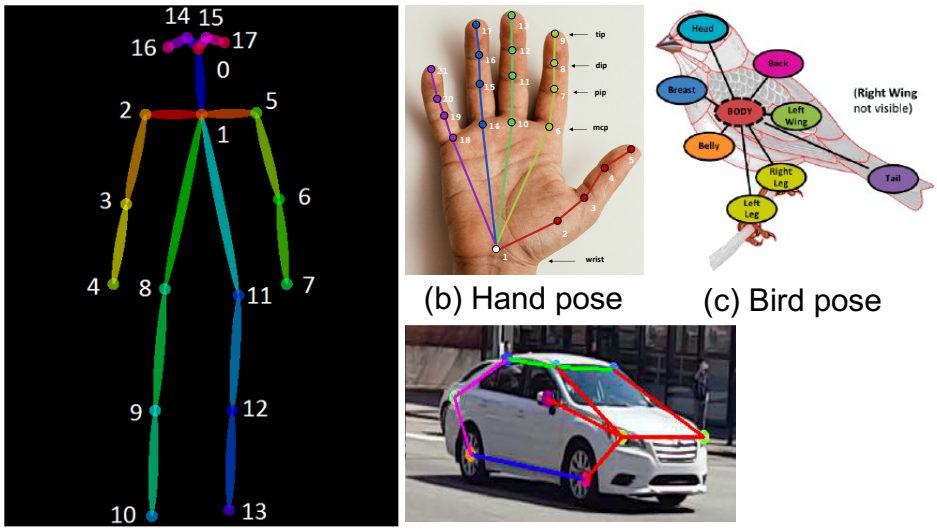


Figure 1.2: Examples of pose estimation for (a) Human pose [28], (b) hand pose, (c) bird pose from the CUB-200-2011 dataset [29], and (d) car pose [30].

each j-th point, $\hat{P}_j \in \mathbb{R}^2$, is bounded by the input image dimensions height, H, and width, W. Instead of regressing two-dimensional coordinates directly, modern works output heatmap probabilities, $\mathbf{H}_j$, for each joint to learn implicit spatial dependencies in an end-to-end fashion [32], [33]. The final position is then selected from the maximum peak of the heatmap, $\hat{P}_j = \max(\mathbf{H}_j)$.

Pose tracking is the natural extension from images to video, but also introduces a temporal consistency constraint. This states that inferred poses from consecutive video frames, $\{\hat{\mathbf{P}}^1, \hat{\mathbf{P}}^2, ..., \hat{\mathbf{P}}^T\}$ for T frames, must refer to the same body and joints. The complexity of this task is further increased when also considering multiple instances, progressing to a multi-object pose tracking problem. In this problem space, many challenges arise from variations in appearance, lighting, self-occlusions, instance swapping, and availability of training data.

Inferring the articulated pose of a subject is an essential piece for extracting image-level information from video, one of the first steps for tackling the human action analysis problem. It allows us to highlight the most salient information while avoiding overfitting to noisy environment (visual) cues. In turn, this provides robustness to variations in appearance for downstream tasks. In our work we operate on human and hand poses. We propose a new video dataset with hand annotations and a neural network model to improve hand pose estimation of videos.

1.2.2 Ground Reaction Force Prediction

We define the Ground Reaction Force (GRF) as the three-dimensional contact force, $\mathbf{F}$, between the human body and the ground plane. The ground truth is sourced from force plates placed under

the feet, measuring in Newtons the horizontal shear forces (F_x, F_z) and a vertical force (F_y) for each foot. This measurement represents external forces that may help us understand how a subject is physically interacting with the environment, which gives us further insight into understanding how effective a particular movement may be. Industry practice typically measures force plate data in a lab environment and correlates this with motions estimated from a motion capture (mocap) system, but we instead learn to predict the GRF from video via inferred poses, $\{\hat{\mathbf{P}}^1, \hat{\mathbf{P}}^2, ..., \hat{\mathbf{P}}^T\}$ (discussed in the preceding section). We limit our estimated collisions and inferred forces only to interactions between the subject and the ground plane, forces with objects are out of scope of our work.

Similar to modern works [34]–[36], we use neural networks to predict these contact forces given a video sequence in a supervised manner. However, publicly available data in this space is very sparse, so we provide a novel video dataset containing tracked human poses with synchronized force plate data. To soften the labeled data requirement, we can show that multi-task learning on an auxiliary task can improve the prediction of unseen motions. Many datasets such as Human3.6m [37] contain massive amounts of motion capture videos with no forces, but can used in conjunction with force-labeled data. Adjacently, we examine the importance of physical plausibility from the inferred poses using a rigid body simulator [38]. Using these simulation-derived metrics can further reinforce the relationship between poses and realistic forces by systematically identifying implausible actions.

1.2.3 Technical Skill Assessment

Assessing human performance is the process of measuring the quality of a performed action. This can be highly subjective, so in our work we focus on technical skills identified from a standardized rubric, such as OSATS [39] for surgery or ISU Judging System [5] in Olympic-level skating. Normally these technical actions are manually assessed, but this can be increasingly time-demanding, require expert evaluators, and still subject to bias [4], [5]. In contrast, we work towards automated alternatives which are scalable, provide immediate feedback, and show promise of objectivity by learning underlying features correlated with ratings. We view technical skill assessment as a regression problem, supervised by ratings collected from experts. While the applicable domains for this problem space maybe broad, in our work we focus only on technical skill assessment in the surgical domain. Some works use general video features [13] and surgical instrument tracking [9], [10] to approximate surgical technical skill and levels of experience. However, we focus on articulated hand poses which are applicable to diverse procedures and independent of specific surgical instruments. Others [21], [22], [40], [41] have introduced work in similar directions but are confined to simulated environments with a single actor and no distractors. Azari *et al*. [40] generate kinematic features from very short (5-30 seconds) operating room videos but with manually initialized and corrected tracking regions. Similar to us, Goodman *et al*. [42]

extracts features from the open-surgery environment but performs dimensionality reduction analysis on top-down kinematic hand attributes.

1.3 Contributions

Our contributions in this dissertation center around learning features that quantify performance of human actions in video, collecting essential data for those tasks, and overcoming dataset limitations through semi-supervised learning techniques.

1.3.1 Improving Articulated Hand Pose Tracking

Among existing hand pose datasets [43]–[46], none support the essential component of multi-instance tracking. In response, we publish *SurgicalHands*, a novel video dataset with multiple instances of articulated hand poses in the surgical environment. While these videos are sourced from operating rooms, they contain a high degree of complexity in movements and variety in appearance (*e.g.* camera placement, multiple surgical team members, colored gloved hands). Most importantly, we introduce a hand pose tracking benchmark that can be used to evaluate models. Also within the tracking domain, we address temporal consistency limitations from existing an frame-wise pose estimation model [47]. This contributes to flickering artifacts and noisy estimates between frames. To address this we propose CondPose, a conditional hand pose estimation model that incorporates past observations as priors. When compared with a frame-wise independent strategy, we see that our model has a much greater impact on the localization accuracy hence also improving tracking accuracy and consistency. With a better precision and better tracking, we can guarantee a better representation of the hands in the scene. This is important for a reliable method that provides a salient signal to be used for other applications such as approximating skill or understanding certain actions.

1.3.2 Novel Approaches for Ground Reaction Force Prediction

There is an absence of publicly available data that provides ground reaction forces paired with videos of human actions. We are only knowledgeable of a single dataset, LAAS Parkour [48], which only supports a single viewpoint of four actions and 28 videos in total. In contrast we publish *ForcePose*, a video dataset of multi-view tracked human motions with paired force plate data and $1,300$ videos. First, we show comparable performance for predicting GRFs between pose detections and motion capture markers, demonstrating that state-of-the-art physical markers can be substituted. We also present a new approach for estimating GRFs through multi-task learning on 2D-to-3D human pose estimation as a subtask, and that we can drastically minimize peak impact

errors with a Gated-MSE loss, at a low cost in Root Mean Squared Error (RMSE). The impact of which allows us to train concurrently with motions from other multiview datasets, such as the walking motion from Human3.6M [37]. This contribution takes a great step towards analyzing the quality of human motions and actions through estimating ground reaction forces. However, there are still errors when performing on unseen motions and multi-task learning does not sufficiently leverage existing multi-view video data for force prediction. We begin to address this limitation with data-augmentation, which improves the plausibility of those force predictions, but comes at an increased cost in error for known motions.

1.3.3 Introducing Physical Simulation as a Metric for Physical Plausibility

An often overlooked aspect of pose estimation is physical plausibility, which can identify violations of physical laws or impossible postures. The standard Mean Per Joint Position Error (MPJPE) metric does not consider global translations in the world space and often displays little correlation between errors and visual quality [49]. While alternative metrics [50]–[52] have been studied, they do not model the progression of instability or how earlier faults impact the stability of latter poses. In 3D human pose estimation, we propose two simulation-based metrics, COM trajectory distance and dynamic stability, for evaluating physical plausibility within rigid body physics simulation [38]. We hypothesize a positive correlation between the physical plausibility of a sequence of 3D poses and stability during physical simulation. The more stable the simulation, the greater the likelihood of the pose being physically plausible. These two simulation-based metrics are designed to understand how well a pose can be simulated and at which point it fails catastrophically. We evaluate on Human3.6m and show agreement with previous metrics and also invariance to spatial alignment of 3D poses.

1.3.4 Assessing Technical Skill from Open Surgery Procedures

Many related works focus purely on videos of benchtop simulations [13], [21]–[23], but these may not translate to proficiency of actions in surgical procedures. Instead, we perform surgical skill analysis on videos of real cardiac surgery. We collect videos from surgical phases that are most indicative of technical ability with collaboration from Michigan Medicine and affiliate institutions. In a pre-training process, we introduce clip-level contrastive learning to learn good generic features before fine-tuning on specific tasks. Rather than coarse frame-level features, we use instance-level hand pose features. Formally, we separate skill analysis into two tasks: classification and assessment. In skill classification, we predict the expertise (novice or expert) of surgeons from video. In skill assessment, we measure our predicted Objective Structured Assessment of Technical Skill (OSATS) [39] scores to those of medical experts. We share results on a simulation dataset, VTS [21], and

our newly collected data, showing superior performance when implementing our clip-contrastive learning pre-training on a simple linear neural network.

1.4 Thesis Statement

Through limited labeled video data, we can infer quantitative evaluation characteristics of human actions.

CHAPTER 2

Related Work

In this chapter, we summarize works related to various aspects of this dissertation.

2.1 Pose Estimation and Tracking

Pose estimation and tracking covers a myriad of classes that can be extracted from images and video. We review select works that are related to our group of problems starting with human pose, hand pose, and surgical instruments.

2D Human Pose The most common of pose estimation problem is 2D human pose estimation. Current research can be grouped into top-down [47], [53]–[56] and bottom-up [57]–[60] approaches. Top-down methods first detect all persons from an image using an object detector [61], followed by estimation of a human pose for each detection using a pose estimation network. In contrast, bottom-up methods detect all human joints in an image in a single-shot, followed by graph minimization and bipartite matching [57] to assign joints likely belonging to each person. Top-down approaches are typically shown to have superior performance [56], where publicly available frameworks such as Detectron [62] or OpenPose [63] are used off-the-shelf for object detections as an intermediate step or for direct human pose estimation. In our work, we take estimation of 2D human poses for granted and focus on downstream tasks using fixed pose detections from Detectron.

To perform human pose tracking, many of these works rely on post-processing optimization and greedy matching [60] to produce the likeliest associations between frames. Xiao *et al*.[47] proposes greedy matching of bounding boxes between frames using IoU (intersection-over-union) overlap and propagated using optical flow, [53] use deformable convolutions to warp predictions between frames, and [55] introduce a Graph Convolutional Network (GCN) [64] to match learned embeddings between human poses. These approaches spatially shift pose predictions that while effective, cannot overcome certain factors (e.g. missed joints) present in the pose estimation step. In Chapter 3, we address this problem for hand poses at the detection step by integrating past pose observations into each new predicted output.

3D Human Pose 3D human pose estimation is a similar problem that seeks to resolve a human pose within a three-dimensional space, from either monocular [65]–[67] or multiple viewpoints [68]–[72]. Monocular camera-based solutions typically use a two-stage process regressing a 3D pose from estimated 2D joint centers [67], [73]–[75] (2D-to-3D inference) or end-to-end pipelines [65], [76]–[79] often by approximating parameters of statistical 3D body models [80], [81] without any intermediate supervision. Monocular camera-based solutions are cost-effective, however, they present unique challenges due to their inherent depth ambiguity. Hence, these methods require training on large motion capture datasets [37], [82] to deliver sufficient results. Multi-stage methods on the other hand, commonly employ a 2D detector to approximate a 2D pose of the person and then "lift" this into a 3D pose. Multi-view methods aggregate multiple camera angles when estimating a 3D pose, hence they require known camera projection matrices. Some works regress a 3D pose from multi-view 2D detections in the feature space[69], [71], [83], for volumetric regression, or in the image space [68], [70] through differentiable triangulation. More recent work [72] has introduced transformers for direct regression of the 3D pose. In Chapter 4, we assume 3D poses are given, generated from triangulated 2D poses from an off-the-shelf pose estimator. We argue that 2D-to-3D inference can serve as appropriate pre-training and simultaneous supervision for our GRF prediction task.

The most commonly used metrics for quantitative evaluation of 3D human pose estimation models are Mean-per Joint Position Error (MPJPE), Procrustes-aligned MPJPE (PA-MPJPE), and Probability of Correct Keypoint (PCK). MPJPE measures the average displacement between predicted 3D joints and corresponding ground truth joints, while PA-MPJPE first aligns both poses using translation, scaling, and rotation. It should be noted that while these methods emphasize close proximity to ground truth joints, they fall short in their ability to quantitatively evaluate plausible postures or motion. A recurrent issue is the production of unrealistic artifacts such as foot skating, ground penetration, and unrealistic postures [50]. This stems from the purely kinematics-driven approach of these methods, which are incapable of considering the physical plausibility of produced poses. Recent works [52], [84]–[86] have sought to resolve this by incorporating human dynamics to ensure feasible 3D poses. They estimate required joint torques using trajectory optimization [50], [52], [87]–[89], reinforcement learning [51], [90], [91], or differentiable simulators [86], [92]. But the learned dynamics have not been shown to be grounded to real data and accurate motion reconstruction is often only achieved after applying non-existent residual forces [91].

Physics-aware 3D Human Pose The previously mentioned 3D HPE methods are strictly kinematics-based, meaning they only model motion and disregard underlying forces driving joint movements. Failure to represent dynamics results in many implausible pose estimations such as floating, foot skating, and unrealistic postures [50]. To address these implausibilities, some works have proposed

floor contact metrics to measure ground penetration [51], [52] and foot sliding (or skating) [50], [51]. But these metrics are most reliable on a calibrated floor plane and use handcrafted heuristics to compute errors. For a measure of the stability of static poses, previous works [52], [93] incorporate terms to measure balanced and unbalanced static postures. They assume the inverted pendulum model from biomechanics literature [94]–[96] which defines a stable pose as one with a center of gravity that falls within its base-of-support (BoS), *i.e*. the convex hull of all ground contacts. However, this is only valid for stationary poses and is insufficient for dynamic scenarios because it does not account for the velocity of the center of gravity [94]. Included with these losses are often temporal smoothness errors [51], [52], [65], [89] to discourage jittery motion and 2D re-projection losses [52], [84], [89] to encourage re-alignment on the 2D image plane. Despite these advances, these metrics do not analyze the dynamic impacts of physical implausibility and oftentimes still generate body motions lacking desired levels of physical realism. For a more comprehensive evaluation, we look to physical simulation, in Chapter 5, as a test bed for plausible motion assessment through gravitational and frictional forces, collisions, and temporal consistency.

Hand Pose Works in 2D hand pose estimation [43], [45], [97] are analogous to human pose estimation, in terms that they can be solved either bottom-up or top-down. At the time of our work, there were many image datasets [43]–[46] for hand pose estimation but virtually none for video. In Figure 2.1, we share different forms of existing hand pose datasets in the literature. These image datasets are sourced from a combination of manual, synthetic, and network-predicted annotations, but none satisfy the conditions of multiple object instances and tracking from video. Zhang *et al*. [46] operate on video data to perform hand pose tracking, where they use a disparity map from stereo camera inputs to estimate a 3D hand pose. However their data consists of only a single hand and at most one detection per frame. Larger, newer, datasets such as InterHand 2.6M [98] boasts 2.6 million frames of one pair hands interacting in a fixed environment but is generally not diverse. To supplement this need for hand pose tracking data, we propose our *SurgicalHands* dataset. While this data covers our target environment with operating rooms, it supports hand pose detections, multiple instances, interaction with instruments, and tracking. We provide additional detail in Chapter 3.

Instrument Pose Deep learning methods in the medical video domain primarily compose of Robotic Assisted Surgery (RAS) videos. Works in this space [99]–[101] may measure kinematic data directly, but require specialized instruments and tools. But full kinematic information (position, velocity, acceleration) is only available for robotic-controlled tools, even less so for hand-held instruments. Adding any external apparatus to capture kinematic data can be prohibited and may negatively impact the flow of an operation. Computer vision-based approaches follow the top-down paradigm by first using a region proposal network to perform detection [10], [11], [102], and then

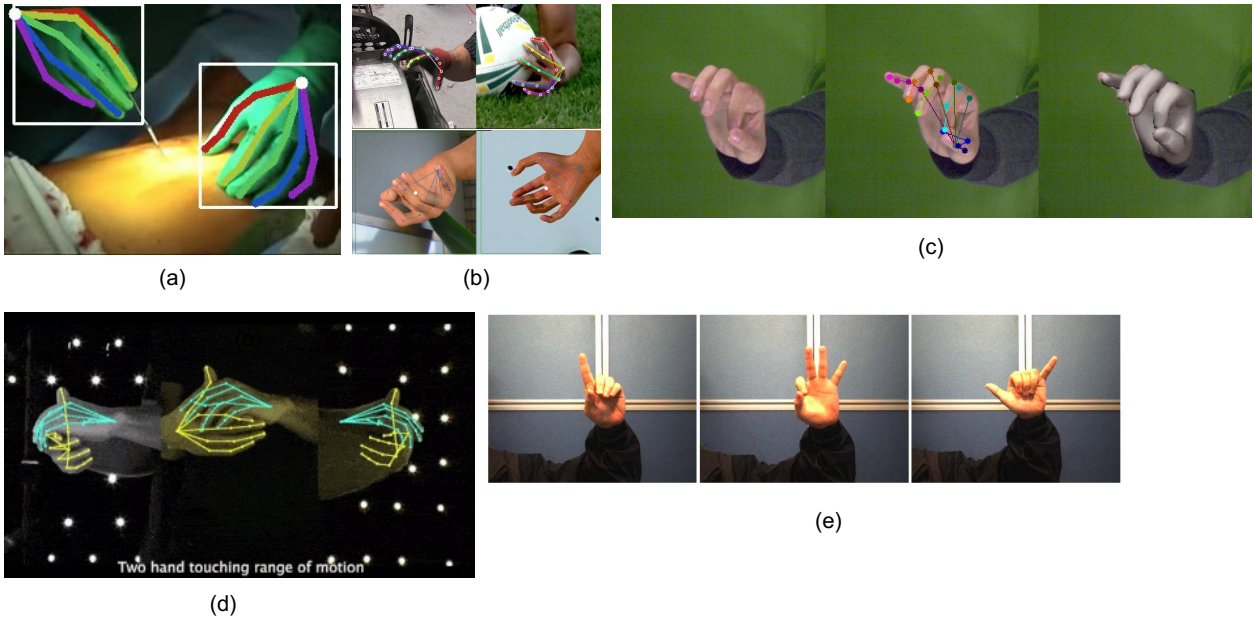


Figure 2.1: We show various forms of hand pose data in the literature. With our video-based *SurgicalHands* dataset in (a), synthetic and single images in (b) [43] and (c) [45], and video data with one or two hands in controlled environment shown in (d) [46] and (d) [98].

segmentation [103], [104] or articulated pose estimation [105], [106] within each bounding box. To incorporate tracking, existing works may use a similarity function based on weighted mutual information [107] or Bayesian filtering as part of a minimization problem [108]. Nwoye et al. [109] are the first to measure the Multiple Object Tracking Accuracy (MOTA) [110] for surgical instruments in this setting, using a weakly-supervised approach with coarse binary labels indicating the presence or absence of seven surgical instruments. However, their evaluation contains at most one unique type of tool at each frame; hence, can be narrowed down to an object detection problem. Unlike their work, we track multiple instances of the same object (hands) in each frame.

2.2 Physical Force and Load Analysis

We use ground reaction forces (GRFs) as a measure for physical forces and loads exerted during human motion.

Hardware Solutions To analyze these biomechanics, we require computed kinematics and dynamics, conventionally sourced from reflective surface markers and time-synchronized force plates in a constrained lab environment. Many existing works follow hardware solutions and use physical markers [111]–[113] or sensors such as IMUs and accelerometers [114], [115] to infer kinematics. Kim *et al.*[116] use a 3D-CNN to learn forces produced by a load cell on various

household objects. However, kinematic measures from physical devices are generally applicable only in rigid environments and may vary based on human-error of sensor placements [117]. Even the attachment of sensors are susceptible to variance, noise and movement artifacts. Hence, we opt for a marker-less (human pose estimation) approach to predict marker positions and shift to neural networks for predicting exerted forces.

Video Solutions Other works [34]–[36] have sought to instead operate on video-based inputs. Jeong *et al*. [36] estimated the center of mass trajectory from video and map it to GRFs using a spring mechanics-based walking model and Goldacre *et al*. [35] learn a least squares estimator matrix for sidestepping and running motions. Li *et al*.[48] solve a large-scale trajectory optimization problem to jointly learn contact forces and 3D motions from re-project detected 2D joints. Our work is most similar to Morris *et al*. [34], who use a recurrent network on detected human poses to predict GRFs in sidestepping maneuvers. The authors note a higher error in the medio-lateral (side-to-side) movements that we hypothesize is a result of not integrating poses from multiple views nor implicitly utilizing a 3D pose. In contrast to prior approaches, in Chapter 4 we are the first to use a transformer to address this problem, leveraging features learned from 2D-to-3D HPE domain to address limitations in single viewpoints. Given the lack of publicly available data with ground truth contact forces, we introduce a manually collected dataset, *ForcePose*, in our experiments. This dataset contains more dynamic movements and number of data samples in comparison to unpublished datasets described in other works [34], [36], [118]. We provide more detail in Chapter 4.

2.3 Human Skill Assessment

In this section, we review works that highlight strategies and prior research in measuring the quality of performed actions. This is a broad problem space covering a variety of topics. Specifically, we cover sports assessment, generic household tasks, and surgical skill assessment.

Sports Assessment The most common class of human skill assessment lies in the sports applications domain [6]–[8]. One of the earliest works from Pirsiavash *et al*. [6] collect the MIT Olympic Sports dataset by leveraging Olympic sports videos with professional scoring from judges. They extract the human pose of the performer and train a Linear Support Vector Regression (L-SVR) model to predict the scores from the given features. The authors also consider feedback by computing the gradients of each joint location and adjusting the position of the joint that most improves the score. However, this feedback may not be completely reliable as the method does not account for physical limitations. Parmar *et al*. [7] demonstrates that a video classification network combined with an

SVR model performs best for extremely low amounts of data. Rather than estimated poses or generic video features, Wang *et al*. [119] employ a visual object tracker and propose a self-attention model that attends only on the features related to the tracked boxes in a spatio-temporal tube. They claim that pose estimations are far too noisy and erroneous to be useful, but tracked image regions can still capture dynamic characteristics of human motion. Pan *et al*. [8] employ a similar approach by building a graph-based model using local spatio-temporal patches around the athlete joint pose centers to capture relational and global motion information. While most skill assessment works regress a single value, Tang *et al*. [120] show better correlation when approximating a Gaussian score distribution to account for uncertainty-awareness between actions. For tasks with multiple assessors, they learn a separate distribution for each one and display further improvements in the Spearman rank correlation score. In other attempts for alternative regression techniques, Yu *et al*. [121] propose a contrastive regression framework to learn relative scores between videos and Xu *et al*. [122] propose a grade decoupling approach, by training a Transformer to output a softmax distribution on a Likert-scale as evidence for final scores. Modern deep learning solutions operate primarily in the image-level domain, but learning a mapping to the physical domain could prove invaluable. Fieraru *et al*. [123] move in this direction by comparing the active kinematic joint angles between experts and trainees to recommend pose adjustments during fitness exercises. But we believe considering dynamics and forces can further ground the understanding of actions, the first step is estimating GRFs like in Chapter 4.

Generic Household Tasks There are a class of generic technical skill tasks, such as drawing, rolling pizza dough, or using chopsticks [14], [15], that lack structured rubrics for grading. Sener *et al*. [124] propose an ego-centric video dataset for assembly and disassembly of toys that with skill levels annotations, from 1 (low skill) to 5 (high skill), in addition to mistakes and action labels. However, a single value to characterize the demonstration of skill level in generic tasks would prove difficult to standardize across human annotators. Doughty *et al*. [14] overcomes the subjectivity of absolute ranking by instead considering pairwise rankings. They ask a human to consider a pair of videos and rank one as demonstrating more skill than the other. Through learning discriminative features using a temporal segmentation network, they demonstrate some ability to predict relative skills on four different tasks. The authors advance this work in [15] by considering that not all parts of a video demonstrate equal skill. They introduce an attention module that weighs different portions of the video when computing predictions.

Surgical Skill Assessment Surgical skill assessment aims to identify qualities that correlate to high proficiency of technical skill. Most research in this field is dominated by usage of robotic or hand-held surgical instruments [9]–[12]. While other existing work focuses on simulated

Category	Low Score Description	High Score Description
Respect for tissue	Frequently used unnecessary force on tissue.	Consistently handled tissues appropriately with minimal damage.
Time and motion	Many unnecessary moves.	Economy of movement and maximum efficiency.
Instrument handling	Repeatedly makes tentative or awkward moves with instruments.	Fluid moves with instruments and no awkwardness.
Knowledge of instruments	Frequently asked for wrong instrument or used an inappropriate instrument.	Obviously familiar with the instruments required and their names.
Use of assistants	Consistently placed assistants poorly or failed to use assistants.	Strategically used assistant to the best advantage at all times.
Flow of operation	Frequently stopped operating or needed to discuss next move.	Obviously planned course of operation with effortless flow between moves.
Knowledge of procedure	Needed specific instruction at most operative steps.	Demonstrated familiarity with all aspects of the operation.

Table 2.1: Detailed description of OSATS categories developed by Martin *et al*. [39].

environments [13], [21]–[23], which are not representative of challenges faced by active surgeons. Gao *et al*. [9] proposes the JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS) dataset which collects videos of surgeons of varying skill performing three simulated surgical tasks with robotic instruments. Jin *et al*. [10] shows positive correlations between tracking characteristics of surgical tools and manually rated assessments of surgical performance in laparoscopic surgery. Similarly, Liu *et al*. [12] jointly attends to the tool position, visual features, and event detection to solve this problem as part of a unified framework.

The Objective Structured Assessment of Technical Skill (OSATS) proposed by Martin *et al*. [39] seeks to produce a structured rubric designed to reduce bias in evaluations. We share an overview of the OSATS categories and descriptions in Table 2.1. This is normally labeled on a Likert scale with the lowest score, given a one, and the highest score, a five. Sharma *et al*. [13] automates OSATS metrics in a simulated surgical training lab setting by extracting sequential motion textures. While they show improved performance compared to a bag-of-words approach (a codebook representation of common features), it is unclear if their features are relevant to the movements of the surgeon or spurious visual details. Abstracting to hand motions, Azari *et al*. [125] analyze videos of clinicians of varying experience levels performing a suturing task on three

separate tissue types. They approximate technical skill by measuring the characteristics of the motion such as the tracked path, acceleration, and total distance travelled. In [126], the authors also study how hand movements can be used to predict surgical maneuvers such as suturing, tying, and transition states. Similar to the last two works, we focus on hand poses and motions to move towards generalizing surgical skill estimation, However, in Chapter 6, we experiment and evaluate on videos of live open-surgery, particularly in cardiac surgery.

2.4 Semi-supervised Learning

Semi-supervised learning [127] is a loosely-defined term and its definition varies depending on the available data and target task. In Figure 2.2, we demonstrate the differences between fully-supervised, semi-supervised, and unsupervised data in image classification. In this instance, semi-supervised learning can use the appearance features of the labeled cat and dog images to identify those corresponding features in the unlabeled images. A common approach may involve domain shift adaptation by learning feature representations using GANs [128] or contrastive learning [129], followed by fine-tuning on labeled samples. Berthelot *et al*. [130] propose a MixMatch algorithm that minimizes the entropy of averaged predictions from data-augmented labeled and unlabeled images. In a slightly different way, multi-task learning [131] aims to learn an inductive bias through shared representations by learning multiple tasks in parallel. This can use samples with varying types of annotations such as segmentation masks, bounding boxes, or image captions. Weakly-supervised learning overcomes the need for dense annotations by leveraging coarser higher-level annotations for finer tasks, this can fall under semi-supervised learning as shown in Figure 2.2. Zhou *et al*. [132] utilizes this for object grounding from cooking videos through the use of instructional recipe captions. Lin *et al*. [133] produces semantic segmentation masks seeded from simpler scribble annotations on objects of interest. Cycle consistency is slightly closer to unsupervised learning, assuming that correspondences can be maintained between transformation of samples. This idea has been used in visual tracking, where many works [134]–[136] propose object trackers that use cycle consistency as a constraint. This enforces an object trajectory to be identical both forward and backwards in time, where deviations indicate error in the track. Zhu *et al*. [26] use GANs to perform unpaired image-to-image style transfer, while [137] extends this from images to video re-targeting with spatio-temporal constraints, and [138] learns self-supervised video correspondences that generalize to multiple tasks. Throughout this dissertation, specifically in Chapters 6 and 4, we explore ways to incorporate some of these semi-supervised techniques to overcome data limitation challenges in our problem spaces.

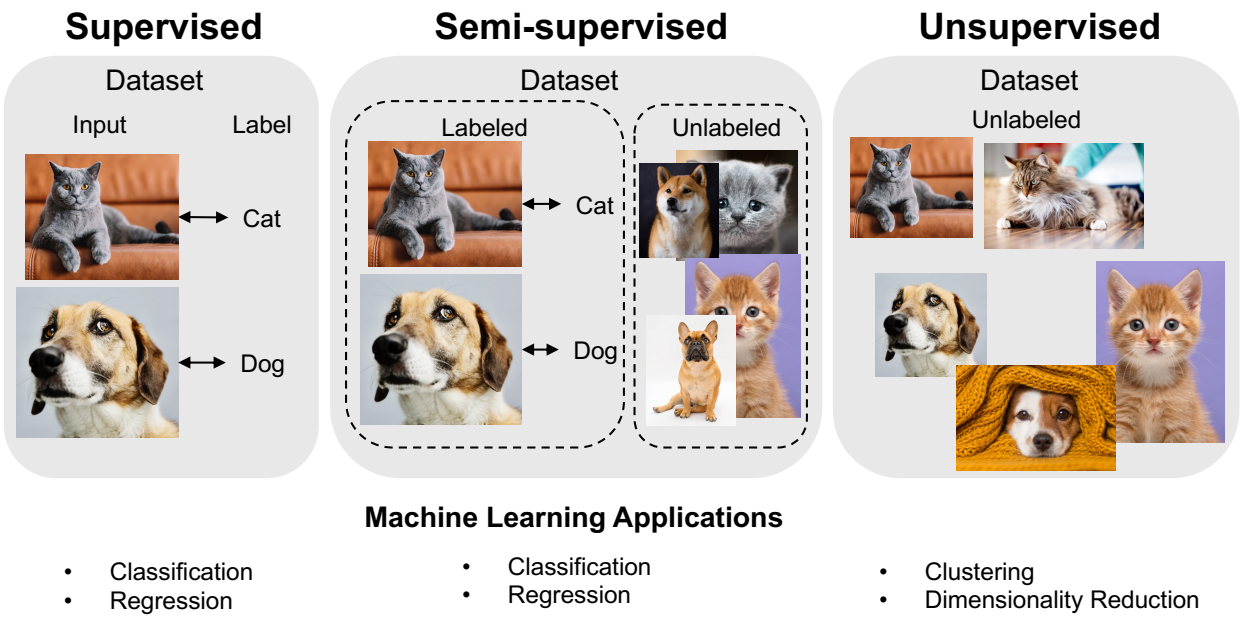


Figure 2.2: We show the range of data supervision available in machine learning. On the left, we have fully-supervised data which provides dense annotations for all samples. On the right, we have unsupervised data which only includes the raw data itself with no labels. While in the middle, semi-supervision is a combination of the two where labeled examples can inform us about unlabeled examples. The bulleted terms depict applicable machine learning applications for each label of supervision. Figure is inspired by [139].

2.5 Operating on Temporal Sequences

We define a temporal sequence as any form of sequential data with dependent or non-randomized relations between time steps. One form form of temporal sequential data is natural language text, where Long Short-Term Memory (LSTM) networks [140] and Recurrent Neural Networks (RNN) have been the defacto choice for natural language processing (NLP) tasks. These tasks range from language translation [141], image captioning [142], and next word prediction [143]. However, transformer networks [144] have since dominated the NLP space and more recently vision and language modalities [145]–[149]. A transformer is an attention-based deep learning model that uses multi-headed attention modules to learn relations between parts of an input sequence, without the use of recurrence. Compared to LSTM networks, transformers avoid issues such as backpropagating through time and vanishing gradients pitfalls. But despite the notable developments, transformer networks lack inductive biases and must often train on larger datasets to generalize well [147]. Examples are seen with BERT [143], training on English Wikipedia (2,500M words), and ViT [147], training on JFT (300M images). Recently, transformers have also been introduced for both 2D [150]–[152] and 3D [153] HPE tasks. Zheng *et al*. [75] presents PoseFormer, which argues

that learning separate spatial (intra-joint) and temporal (inter-pose) encodings in two transformers are important for 2D-to-3D pose estimation. Not only does this include an attention mechanism, lacking in the LSTM network, but it also serves a dual role in our 2D-to-3D HPE subtask.

2.6 Physical Simulation and Optimization

Along the lines of physical grounding in the observed space, we explore the usefulness of physical simulation for measuring the plausibility of 3D human poses. Physical simulators [38], [85] are generally black-box and largely non-differentiable, hence they require external gradient-free optimization methods to obtain optimal parameters. Previous works [51], [84], [86], [89] have used physics simulators to improve 3D human pose estimation, measuring improvements using plausibility metrics on the outputs. Gartner *et al*. [89] incorporate a fully featured physics engine into the pose estimation process to model self-contact and human-object contact. They recover plausible motion by performing trajectory optimization [154], [155] as a sub-process inside the simulation engine. Others, like Yuan *et al*. [51], utilize reinforcement learning to obtain optimal control parameters when training their policy. Recently, computationally efficient gradient-based optimization has also been emerging in differentiable simulators [86], [92], [156]. Our work in Chapter 5 uses trajectory optimization within the simulation engine itself to generate metrics about physical plausibility. We utilize the inherent gravity, friction, and collision constraints of physical simulation rather than measuring plausibility on the output skeletal pose.

CHAPTER 3

Articulated Hand Pose Tracking

3.1 Introduction

Machine learning and computer vision have become increasingly integrated with healthcare in the medical community. This is apparent in the myriad of tasks such as tumor segmentation [157], technical skill assessment [99]–[101], [158], [159], and tool detection and tracking [102], [104], [106], [109]. Here we study the problem of articulated hand pose tracking in the surgical domain. Tracking hand poses can facilitate other useful tasks such as technical skill assessment, temporal action recognition, and training surgical residents. Pose tracking in the computer vision community is primarily centered around human poses [47], [53]–[59], [160], while medical works focus on detecting and tracking surgical instruments [102], [104], [106], [109]. Tracking surgical instruments is useful but these instruments are inherent to the surgical procedures seen during training. Instead we abstract away the emphasis on surgical instruments where articulated hand tracking will be more applicable to broad surgical tasks. Articulated hand pose tracking can highlight important properties such as grip, motion, and tension that human experts often attend to when conducting gold-standard evaluation of technical skill.

A challenge in pose tracking is the temporal consistency of predictions between frames, the lack of which leads to flickering and improbable changes in estimated poses. Existing works [54], [57]–[59], [160] in articulated pose tracking use frame-wise independent predictions along with post-processing when tracking [47], [53], [55], [56] to gather temporal context. However, they do not integrate past inferences when localizing joints. We address this by proposing **CondPose**, a new model that performs predictions conditioned on the pose estimates from prior frames. In Fig. 3.1, we show a comparison of both approaches: the baseline using frame-wise independent predictions and our model using conditional predictions. The initial estimate may fluctuate due to varying factors such as lighting, hand orientation, or motion blur. But we find that using prior predictions as guidance, we can improve our localization accuracy. The internal representation of this object's state (position, appearance, and classification) is a function of its current state and previous states. By learning this Markovian prior for the prediction of hand joints, we can improve both pose estimation and consequently tracking accuracy.

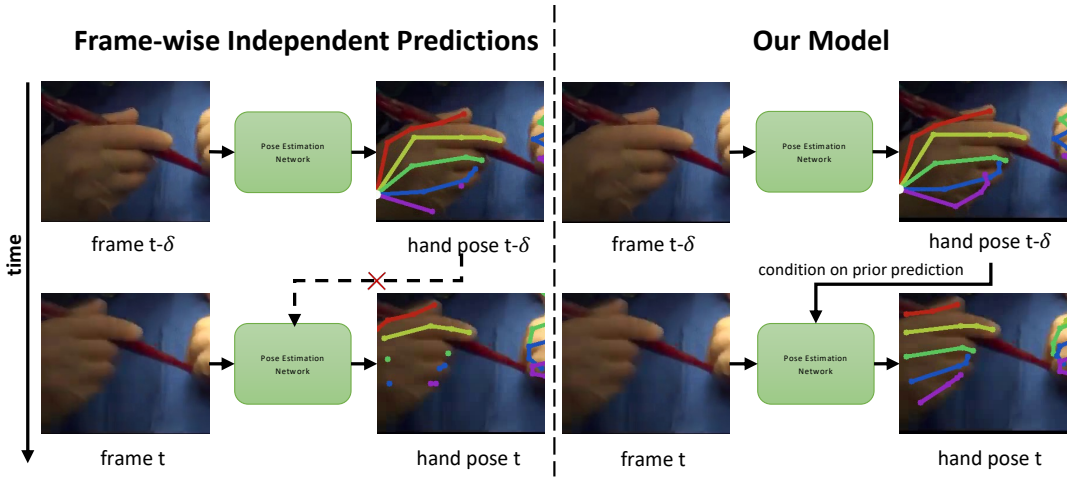


Figure 3.1: On the left, a method only performing frame-wise independent predictions may miss out on properly localizing joints, while on the right, temporally passing past predictions from previous frames improves the network's localization.

There is a lack of data and benchmarks for articulated hand pose tracking. To address this, we collect a novel dataset featuring intra-operative videos of real surgeries, *SurgicalHands*. We annotate the articulated hand poses of surgeons that subsumes both surgical instrument and non-instrument actions, e.g. suturing, knot-tying, and gesturing. We are, to the best of our knowledge, the first to introduce a labeled dataset for both detection and tracking of multiple articulated hand poses. We benchmark our dataset against existing tracking baselines and demonstrate the superiority of our proposed approach on both hand pose estimation and tracking.

Our contributions are as follows:

- We introduce **CondPose**, a novel deep network that takes advantage of confident prior predictions to improve localization accuracy and tracking consistency.
- We present *SurgicalHands*[1], a new video dataset for multi-instance articulated hand pose estimation and tracking in the surgical domain.
- We set new state-of-the-art benchmark performance on *SurgicalHands*.

3.2 Dataset

We introduce the *SurgicalHands* dataset for multi-instance articulated hand pose tracking. As shown in Table 3.1, many existing datasets support detection with no temporal coherence between

[1]Both the code and dataset are available at `https://github.com/MichiganCOG/Surgical_Hands_RELEASE`

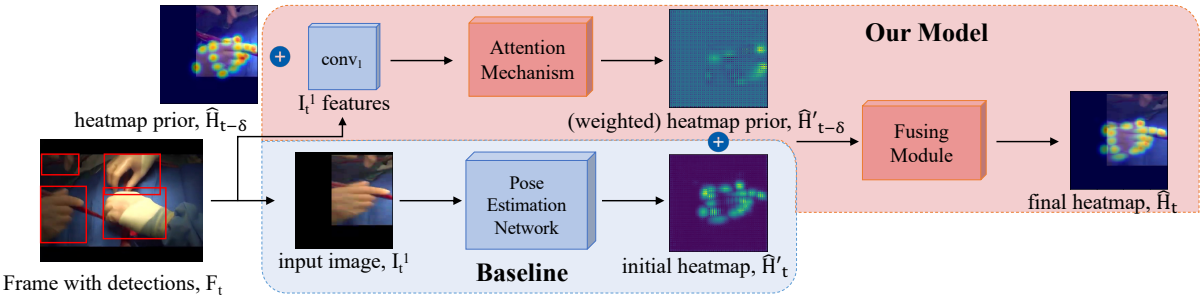


Figure 3.2: The baseline generates a heatmap, $\hat{\mathcal{H}}'_t$, for each detection using a pose estimation network. In our model, we provide additional information by incorporating a heatmap prior from $t-\delta$. Concatenating the image features at t with $\hat{\mathcal{H}}_{t-\delta}$, we pass this through our attention mechanism to produce a weighted heatmap prior, $\hat{\mathcal{H}}'_{t-\delta}$. Both $\hat{\mathcal{H}}'_t$ and $\hat{\mathcal{H}}'_{t-\delta}$ are concatenated and passed through the fusing module, using context from both heatmaps to produce the final articulated hand pose. The initial and final heatmaps represent real outputs from the network, while the heatmap prior (during training) shows ground truth at $t-\delta$)

Name	Labels	Environment	# dets	Detection	Multi-instance	Tracking
CMU Manual Hands (MPII+NZSL) [43]	Manual	In-the-wild + Same background	2.8k	✓	✓	✗
CMU Synthetic Hands [43]	Synthetic	Renderer	14.2k	✓	✓	✗
Panoptic Hands [43]	Trained Network	Multiview-camera studio	14.8k	✓	✓	✗
LSMV 3D [44]	Leapmotion sensor	Research office	184k	✓	✗	✗
Freihand [45]	Hybrid	Green screen + Indoor/outdoor	36.5k	✓	✗	✗
STB [46]	Manual	Indoor	18k	✓	✗	✓
SurgicalHands (Ours)	Manual	Operating rooms	8.1k	✓	✓	✓

Table 3.1: We compare our proposed dataset to other existing hand pose datasets. Our data supports multiple object instances, along with tracking, in each clip.

video frames. Our dataset includes varying lighting conditions, fast movement, and diversity in scene appearances. Distinctively, we also include gloved hands, which appear in contrasting colors such as latex and green.

We lack data for training and benchmarking models on multi-instance hand tracking. Therefore we introduce *SurgicalHands*, a novel video dataset for multi-instance articulated hand pose estimation and tracking in the surgical domain, the first of its kind. From publicly available data, we collect 28 videos with a view of the hands of surgical team members during the operation. From those videos, we extract 76 clips sampled at 8 frames per second and collect bounding box, class label, tracking id, and pose annotations using Amazon Mechanical Turk (AMT) and a modified version of Visipedia Annotation Tools [161]. We show samples of our annotations in Fig. 3.4. Each hand is labeled with the handedness (left/right), 21 joints, and properties for each joint: visible, occluded or non-available. Visible implies that the joint is visibly on screen, occluded means the joint is obstructed but its position can be estimated, not-available means the joint position cannot be inferred or it is off-screen. From our collected data, we have a total $2,838$ annotated frames and

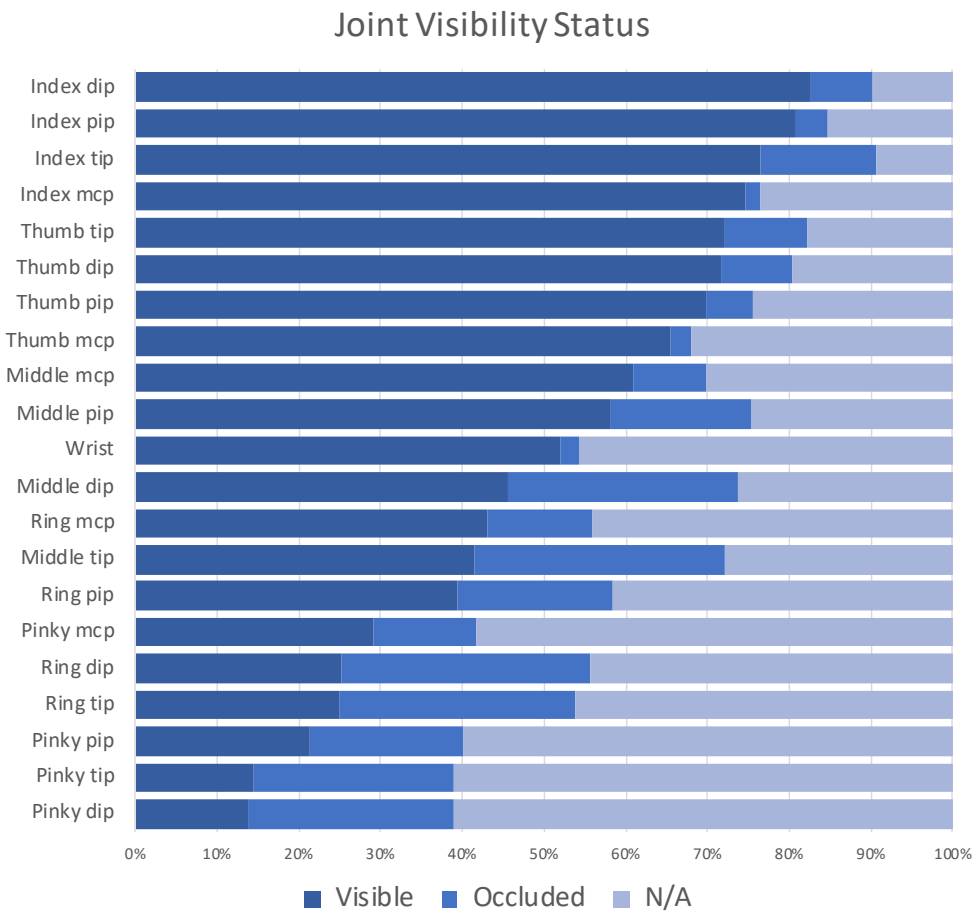


Figure 3.3: Statistics on visibility of each joint. The least visible joints belong to the 4^{th} and 5^{th} digits, this is expected as they are underutilized in most surgical actions.

$8,178$ unique hand annotations from 21 unique annotators. Each annotated frame contains a mean of 2.88 hands, median of 3 hands, and a maximum of 7 hands.

Fig. 3.3 shows the joint annotation visibility across our labeled frames. We see that across all instances, the joints from the ring and pinky finger show the highest rate of being visually obscured or not-available. In the majority of our video clips, position and orientation of the hand makes it extremely difficult to localize those joints. This is expected because the 4^{th} and 5^{th} digits are underutilized in many procedures, the first 3 digits are typically used to hold surgical instruments.

3.3 Method

We propose **CondPose**, to perform articulated pose detection and tracking by incorporating previous observations as prior guidance. We show our model in Fig. 3.2. While the baseline produces a heatmap from each hand using a pose estimation network, we leverage past predictions

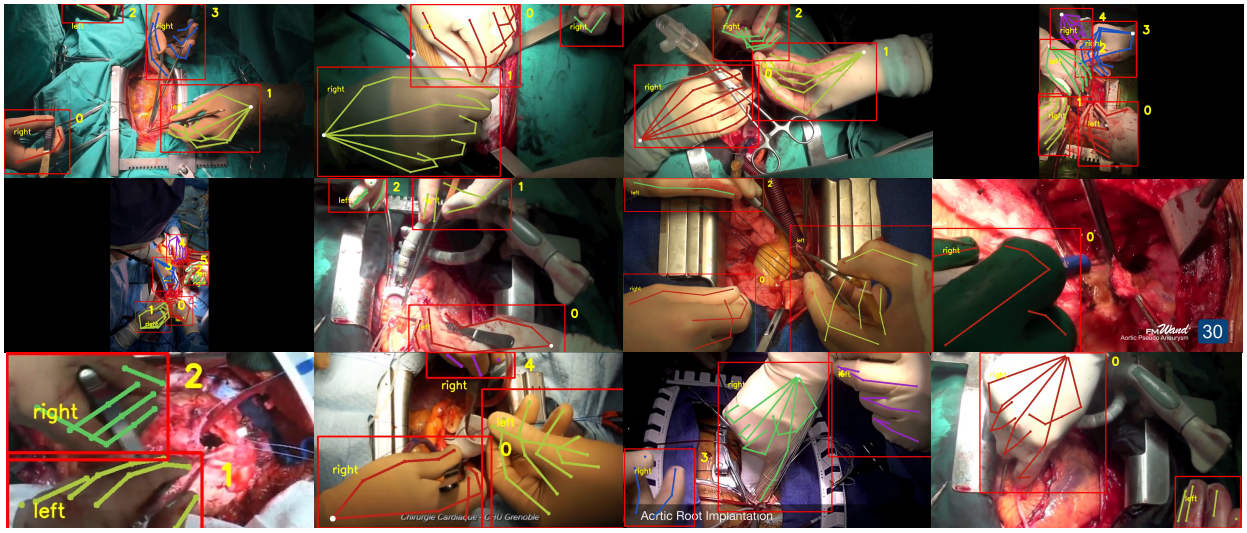


Figure 3.4: We show samples from our annotations. Each hand is labeled with a bounding box, handedness, tracking id, and visibility of joints.

to produce conditioned hand pose outputs, improving detection performance during inference. While we design **CondPose** with video data in mind, we begin with pretraining on image data, finetuning on our video dataset, *SurgicalHands*, and lastly, comparing between different tracking methods.

3.3.1 Hand Pose Estimation in Images

We first pretrain on image data, defining the input and output for the pose estimation network, P, as $\hat{\mathcal{H}} = P(\mathcal{I})$. The input is an image crop $\mathcal{I}$, $\mathcal{I} \in \mathbb{R}^{H \times W \times 3}$, and the output is a predicted heatmap $\hat{\mathcal{H}}$, $\hat{\mathcal{H}} \in \mathbb{R}^{H' \times W' \times J}$. Here H, W represents the input image height and width and H', W' are the output heatmap height and widths. J represents the number of predicted joints of each hand. Each image crop is scaled to 2.2 times the total area of the hand bounding box. We train using the mean squared error (MSE) between the ground truth and predicted heatmaps as $\mathcal{L} = \|(\mathcal{H} - \hat{\mathcal{H}}) \odot \mathcal{M}\|^2$. The ground truth heatmaps, $\mathcal{H}$, are generated from 2D Gaussians centered on each annotated keypoint. $\mathcal{M}$, is included to mask out un-annotated joints. The output joint locations are the max value positions in the third channel of $\hat{\mathcal{H}}$. After pretraining, we finetune our model on videos to learn conditional hand pose predictions.

3.3.2 Hand Pose Estimation in Videos

While image data cannot be used to learn our conditional hand pose predictions, we can initialize weights to speed up our training process and improve generalizability. We finetune **CondPose** on *SurgicalHands*, shown in the top portion of Fig. 3.2. To incorporate a prior branch, we introduce a

heatmap prior, $\hat{\mathcal{H}}_{t-\delta}$, a pose estimate of the same object from $t-\delta$. Our model performs conditional predictions, defined as

$$\hat{\mathcal{H}}_t = M_{fus}(P(\mathcal{I}_t); M_{att}(v_t; \hat{\mathcal{H}}_{t-\delta})). \tag{3.1}$$

In contrast to our previous definition of P, $\hat{\mathcal{H}}_t$ is now conditioned on predictions at a previous time step $t-\delta$. Our model is further composed of two branches: the attention mechanism, M_{att}, and the fusing module, M_{fus}. M_{att} contextualizes the prior heatmap prediction, $\hat{\mathcal{H}}_{t-\delta}$, with image features, v_t ($conv\_1$ in our experiments), at time t. This branch relates the visual representation and the localized heatmap prior, ideally learning to weight each joint prior accordingly. M_{fus} produces a merged final heatmap from the initial prediction, $\hat{\mathcal{H}}'_t$, and weighted heatmap prior, $\hat{\mathcal{H}}'_{t-\delta}$. M_{att} and M_{fus} are both composed of two convolutional layers, followed by transposed convolution, with ReLU non-linearities in-between.

During training the prior is selected from frame $t-\delta$. If the object does not exist at that frame, we use earlier frames up until the first occurrence. If a corresponding object does not exist on any previous frames, then the prior, $\hat{\mathcal{H}}_{t-\delta}$, is set as a zeros heatmap. This is expected behavior during evaluation, because priors do not yet exist at frame one. Also during evaluation, unlike training, the prior associated with the current detection is unknown. Given n priors from time $t-1$, $\{\hat{\mathcal{H}}^1_{t-1}, \hat{\mathcal{H}}^2_{t-1}, \ldots \hat{\mathcal{H}}^n_{t-1}\}$, and k detections at time t, $\{\hat{\mathcal{I}}^1_{t-1}, \hat{\mathcal{I}}^2_{t-1}, \ldots \hat{\mathcal{I}}^k_{t-1}\}$ we pass all pairs through the network to generate candidates. The heatmap with the highest average confidence score is selected as the output for that detection.

3.3.3 Matching Strategies for Tracking

Following the detect-then-track paradigm, we require a matching strategy to performing tracking. Given n hands at time $t-1$ and m hands at time t we use a similarity function to derive similarity measures between each pair at $t-1$ and t. Common methods are intersection-over-union (IoU) of bounding boxes, average L2-distance of the predicted joint locations, or L2-distance between the graph pose embeddings. Similar to Ning et al. [55] we train a GCN to output the embedding of each input hand pose, $\mathcal{X}$, defined simply as $\hat{p} = GCN(\mathcal{X})$. Here $\mathcal{X} \in \mathbb{R}^{J \times C}$, where J is the number of joints and C is the number of channels. For training, we use the contrastive loss [162], $\mathcal{L} = \frac{1}{2}(y * d + (1-y) * max(0, (m-d)^2))$. The contrastive loss places embeddings close in perceptual distance. For a pair of embeddings $\hat{p}^1_v$ and $\hat{p}^2_v$, the variable d represents the L2-distance between the two, $d = \|\hat{p}^1_v - \hat{p}^2_v\|^2$. y is a binary label indicating the same hand, 1, or different hands, 0. m is the margin variable, a hyperparameter used for tuning. For each item in our minibatch, positive pairs are selected between adjacent frames with probability $p = 0.5$ and negative pairs are selected from the same video with $p = 0.4$ or from a different video with $p = 0.1$. We evaluate our

trained GCN models using the classification accuracy between pairs of selected hands, achieving classification accuracies of $> 97\%$.

3.4 Experiments and Evaluation

3.4.1 Implementation Details

We adopt a ResNet-152 pose estimation model [47] to first train on hand pose image data, CMU Manual Hands and Synthetic Hands [43]. We use a batch size of 16, training for 30 epochs, with an Adam optimizer and a learning rate of $1e^{-3}$. When finetuning on *SurgicalHands* we use leave-one-out cross-validation and split our data into 28 different folds. Clips belonging to the same video are in the same validation fold, and the reported metrics are averaged across all folds. We employ a variant of curriculum learning that gradually transitions to predicted priors from ground truth priors. A predicted prior at $t - \delta$ is sampled with a probability of $p = 0.10 * epoch$, until only predictions are used for training at epoch 10 and onward. We empirically select $\delta = 3$ during training. For all training, we apply random rotations and horizontal flipping as data augmentation. When training the GCN for tracking, we using a batch size of 32 and train for 60 epochs and an initial learning rate of $1e^{-3}$. We normalize $\mathcal{X}$ to 0-1, relative to keypoint positions along the bounding box. The input dimension for each input is $J \times C$ where J represents the number of joints and C is the number of channels. We use $C = 2$ for x-y coordinates and $C = 3$ to include annotation state (0 = unannotated, 1 = annotated, or 0-1 for predicted keypoints). We adopt a two-layer Spatio-Temporal GCN [55], [163] to output a 128-dimensional embedding of each pose.

3.4.2 Detection Performance

We evaluate detection performance using mean Average Precision (mAP), the choice metric in human pose evaluation, on our *SurgicalHands* dataset. MAP is computed using the Probability of Correct Keypoints (PCK), measuring the probability of correctly localizing keypoints within a normalized threshold distance, σ. This threshold distance, σ=0.2, is empirically chosen to be roughly the ratio between the length of a thumb joint and the enclosing bounding box. Pose predictions are matched to ground truth poses based on the highest PCK and unassigned predictions are counted as false positives. AP for each joint is computed and mAP is reported across the entire dataset. In Table 3.2 we report the mAP at the highest MOTA score (defined in the next section) for each model. With our recursive heatmap strategy we are able to obtain higher average precision across the different joints in the hand. In Fig. 3.6 we show qualitative examples of our hand pose estimation on various frames from our *SurgicalHands* dataset. The top row clips are sampled from the best performing clips, while the bottom row are from the worst performing clips. We see that

Model	Wrist	Thumb	Index	Middle	Ring	Pinky	mAP
Baseline [47]	**67.23**	60.12	63.29	53.77	48.29	39.28	53.59
Our model	65.51	**62.66**	**64.99**	**57.88**	**51.40**	**44.26**	**56.66**

Table 3.2: Mean Average Precision (mAP). Performance is averaged across all folds

Model	MOTA Wrist	MOTA Thumb	MOTA Index	MOTA Middle	MOTA Ring	MOTA Pinky	MOTA Total	MOTP Total	Prec. Total	Rec. Total	F_1 Score Total
Baseline [47]	**36.7**	**45.83**	57.35	45.53	34.63	8.49	38.27	85	**78.3**	59.13	67.37
Our Model	30.99	44.74	**58.21**	**48.90**	**36.46**	**10.39**	**39.31**	**85.28**	77.61	**62.69**	**69.35**

Table 3.3: We optimize for the Multiple Object Tracking Accuracy (MOTA), each performance metric is averaged across all validation folds

the model suffers most in cases of heavy occlusion, where the camera view excludes the majority of the hand. Ambiguity in the position of the hand furthers the localization errors, e.g. top-down view with most fingers occluded. The best performing cases are those with balanced lighting and an unambiguous view of the first few digits.

3.4.3 Tracking Performance

To measure tracking performance, we use Multiple Object Tracking Accuracy (MOTA) which also takes into account the consistency of localized keypoints between frames. MOTA [110] is defined as:

$$\text{MOTA} = 1 - \frac{\sum_t (\text{FN}_t + \text{FP}_t + \text{IDSW}_t)}{\sum_t G_t} \tag{3.2}$$

$$\text{mAP} = \frac{1}{|\text{joints}|} \sum_{j \in \text{joints}} \text{AP}_j \tag{3.3}$$

This encapsulates errors that may occur during multiple object tracking: false negatives (FN), false positives (FP), and identity switches (IDSW). FN are joints for which no hypothesis/prediction was given, FP are the hypothesis for which no real joints exists, and IDSW are occurrences where the tracking id for two joints are swapped. G represents the total number of ground truth joints. The range of values for the MOTA score is $(-\infty$ to $100]$.

We measure perform tracking using three methods: IoU, L2-distance, and GCN. Intersection-over-union (IoU) measures overlap of two bounding boxes using the ratio: area of intersection over total area, between subsequent frames in our case. L2-distance measures the average L2 distance of

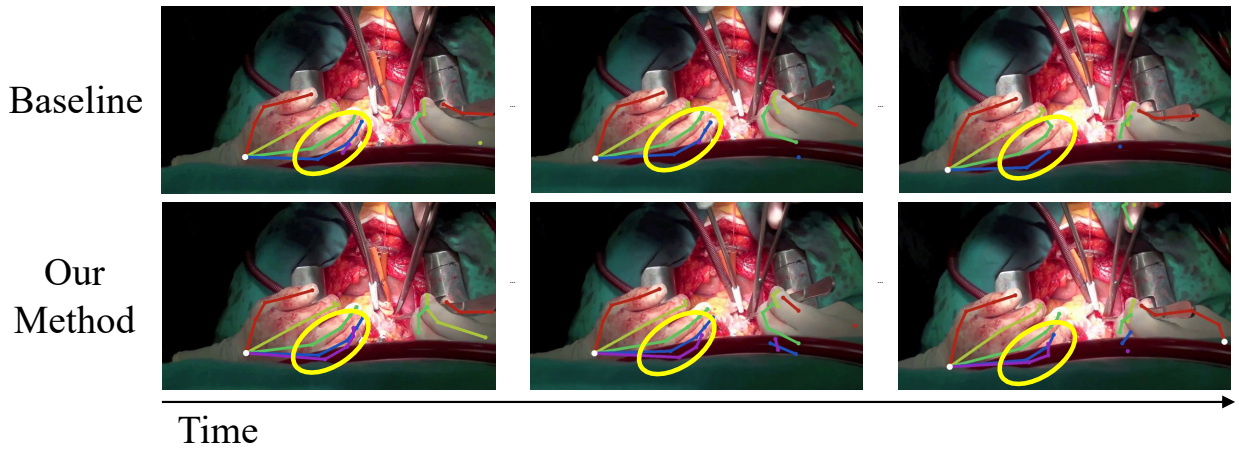


Figure 3.5: We show a qualitative comparison between the baseline model and our method. We note a higher recall and consistency between frames, as shown for the hand to the left. Even when the pinky finger is not visible, the past predictions reinforces those joint locations.

regressed keypoints between frames. GCN measures the embedding similarity between the encoded keypoints to determine matches. We show quantitative results from our experiments in Table 3.4 and the per-joint performance in Table 3.3. Each row is maximized for the highest MOTA score across all hyperparameters, shown along with its corresponding mAP. Our method has a higher MOTA score across all of the videos, but our corresponding mAP scores are greater by a much larger margin. This points to our advantage from temporally leveraging predictions from previous frames during the detection step. We show an example in Fig. 3.5, in a frame-by-frame comparison between the baseline and our method, we note a higher recall and improved localization. While the last digit is obstructed, its position can be reasonably inferred. In the last two columns of Table 3.3 we use an object detector to detect hands, the prior two columns (perfect detections) use the manual annotations. Training an object detector on 100 Days of Hands [164], we see a lower localization and tracking accuracy but a consistent trend from the baseline. The quality of the detections serve as a bottleneck, but the margins of improvements are very similar. While trained with perfect detections as priors, they are not required to maintain performance in practice.

3.4.4 Ablation Analysis

We perform an ablation analysis on the convolutional map in M_{att} and the fusing module M_{fus}. We experiment with no prior convolutional feature map (NC), no attention mechanism (NA), and removal of both (NC-NA), showing our results in Table 3.5. Our full model has the highest scores overall. The attention mechanism and convolutional feature maps have opposing effects on the mAP and MOTA scores. The NC model does not use a convolutional feature map from frame t, so the fusing module is applied directly to both un-altered heatmaps from $t - \delta$ and t. We found

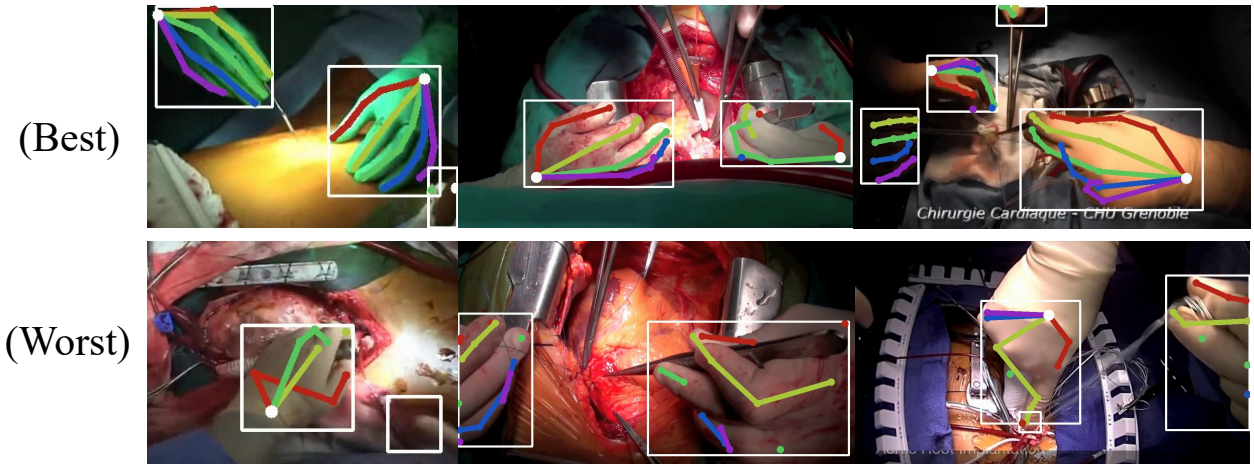


Figure 3.6: We show qualitative samples of frames from the best performing (top row) and lower performing (bottom row) videos. (Best viewed in color).

this increases the mAP value, but lowers the MOTA score. The NA model directly concatenates the convolutional features and the heatmaps, with no attention mechanism. This has the opposite effect, decreasing the mAP significantly but slightly increasing the overall MOTA score. Without contextual convolutional features (NC and NC-NA), the model can still learn to use the prior prediction and improve its detection score. On the contrary, no attention mechanism brings a drop in mAP, which may be attributed to an unrefined prior with noisy features. The small increase in the MOTA score is likely from fewer false positives produced by that model, due to a slightly lower mAP.

We also explore the value of our hyperparameter, δ, during training. We use values $\delta = \{1, 2, 3, 4\}$ and show our results in Table 3.6. Optimizing for highest MOTA score, we found $\delta = 3$ to be best with 39.31, followed by $\delta = 1$ with a smaller MOTA score (39.03) but a higher mAP (58.64 vs 56.66). We find a non-linear correlation between the mAP and MOTA scores, showing a trade-off in mAP when optimizing for the tracking performance. The best strategy is one that maximizes MOTA accuracy with minimal loss in localization precision.

3.4.5 Evaluation on Human Pose

We executed additional experiments on the PoseTrack18 dataset between our model and our re-implementation of the baseline. From Fig. 3.7, we show a narrowed gap in performance but our findings are consistent with our earlier experiments. Our model maintains a higher mAP score for the highest MOTA values. Given the trade-off that occurs between mAP and MOTA, this means our model is more likely to retain its localization precision at higher tracking accuracies.

Model	Matching Strategy	Perfect Det. mAP	Perfect Det. MOTA	Object Det. mAP	Object Det. MOTA
Baseline [47]	IoU	53.59	38.27	48.15	31.46
	L2	52.65	37.78	47.44	31.14
	GCN	52.65	36.78	47.44	30.03
Our Model	IoU	**56.66**	**39.31**	**50.04**	**33.19**
	L2	56.66	38.94	50.04	32.84
	GCN	56.66	38.22	50.04	32.24

Table 3.4: MOTA performance between matching strategies, averaged across all folds. Each row is optimized for highest MOTA performance. Matching strategies share the same base model, so it is possible for them to share the same mAP score.

Model Variant	Matching Strategy	Perfect Det. mAP	Perfect Det. MOTA
NC-NA	IoU	55.23	38.31
NC	IoU	56.00	38.13
NA	IoU	54.70	38.45
Full model	IoU	**56.66**	**39.31**

Table 3.5: Ablation analysis using IoU matching strategy ($\delta = 1$). NC = No convolutional feature map, NA = No attention mechanism.

3.5 Conclusion

In this chapter, we introduce *SurgicalHands*, the first articulated multi-hand pose tracking dataset of its kind. Additionally we introduce **CondPose**, a novel network that makes conditional hand pose predictions by incorporating past observations as priors. We show that when compared with a frame-wise independent strategy, we have better performance in localizing and tracking hand poses. More so, a higher localization accuracy for comparable tracking performance. While tracking drives the consistency of joints through time, the actual shape and characteristics of the hand is described by the localization precision. With a higher localization precision and better tracking still, we can guarantee a better representation of the hands in the scene. While not the focus of this work a reliable hand tracking method can provide a salient signal that can be used to approximate surgical skill or understanding actions.

Model Variant	Matching Strategy	Perfect Det. mAP	Perfect Det. MOTA
$\delta = 1$	IoU	**58.64**	39.03
$\delta = 2$	IoU	54.71	38.42
$\delta = 3$	IoU	56.66	**39.31**
$\delta = 4$	IoU	56.35	38.09

Table 3.6: Effect of δ. Each model is trained with a separate δ value

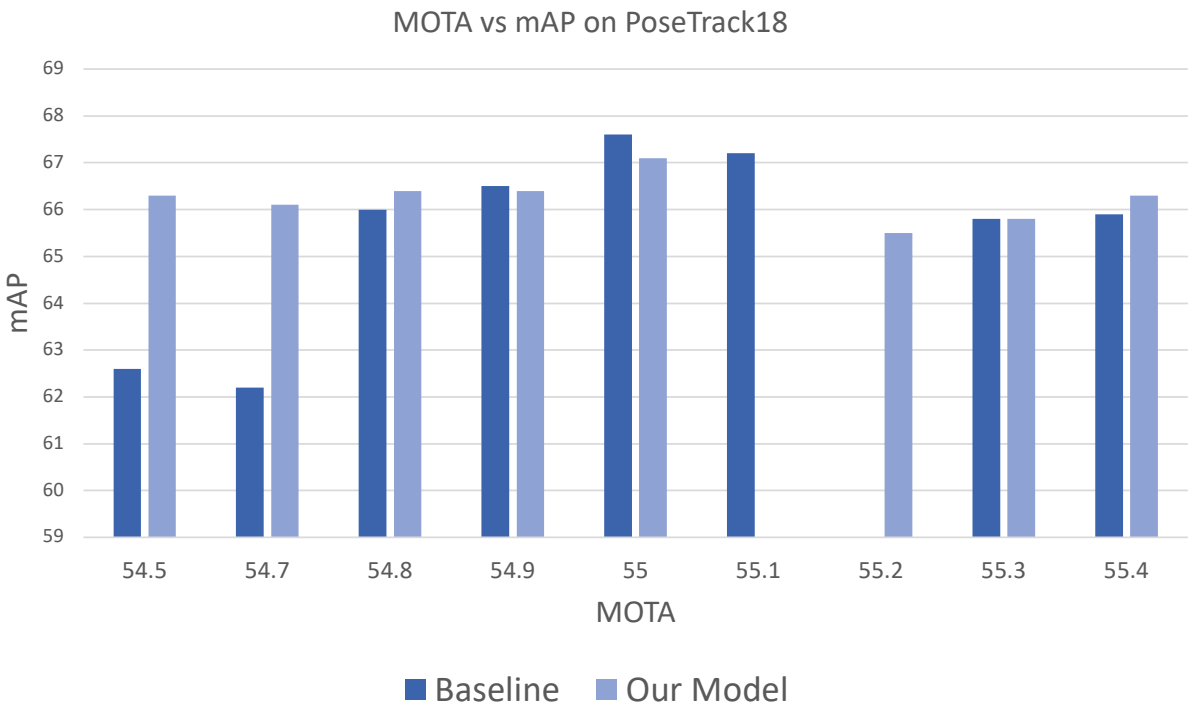


Figure 3.7: Optimized for maximum Multiple Object Tracking Accuracy (MOTA) score, we show the top performing models on PoseTrack18. Consistent with our earlier findings, our model maintains a higher mAP for comparable MOTA scores.

CHAPTER 4

Ground Reaction Force Estimation via Multi-task Learning

4.1 Introduction

For identifying factors in performance improvement or injury prevention of certain physical activities (e.g. running and jumping), an essential quantitative measure is the Ground Reaction Force (GRF) [165]. Used primarily in the field of biomechanics, GRF measures the three-dimensional contact force between the human body and the ground. When paired with calculated kinematics, GRFs can be used to compute inverse dynamics and estimate internal forces and interactions between the different joints, muscles, and bones within a subject [165], [166].

Traditionally [111], [112] this process requires a multi-camera marker-based tracking system and force plates in a controlled lab environment as shown in Fig. 4.1. Consequently, this time-consuming and expensive process cannot be replicated outside of a lab environment [111], [115]. However, advances in human pose estimation in 2D images and video has opened the doors for marker-less systems to replace this *antiquated* approach, like Morris *et al*.[34] who use detected keypoints and an LSTM network to predict GRFs directly from video. But, this work only experiments with a single lateral side-stepping maneuver, includes no implicit or explicit 3D representations, and is limited by the modeling and representation capacity of the LSTM. LSTMs suffer from vanishing gradients on long sequences and lack the capabilities of modern attention-based models.

In contrast, we use a transformer architecture to predict GRFs from video. Transformer networks [144] originated as a driving force in natural language processing (NLP) because of their computational efficiency and ability to learn longer range dependencies than LSTM and RNN networks. But they typically require orders of magnitudes more training data to generalize well [147]. We address this by leveraging features learned from 2D-to-3D human pose estimation (HPE) via pre-training and multi-task learning as a subtask, learning to infer a 3D pose from a 2D input. Because GRFs are predicted on a three-dimensional plane, subtle out-of-plane movements from a 2D view maybe harder to capture with a restrictive point-of-view. So we encode an approximate 3D representation from a 2D pose, along with context of the input sequence, as a joint feature representation transferable to the target force prediction task. Pre-training follows the train and

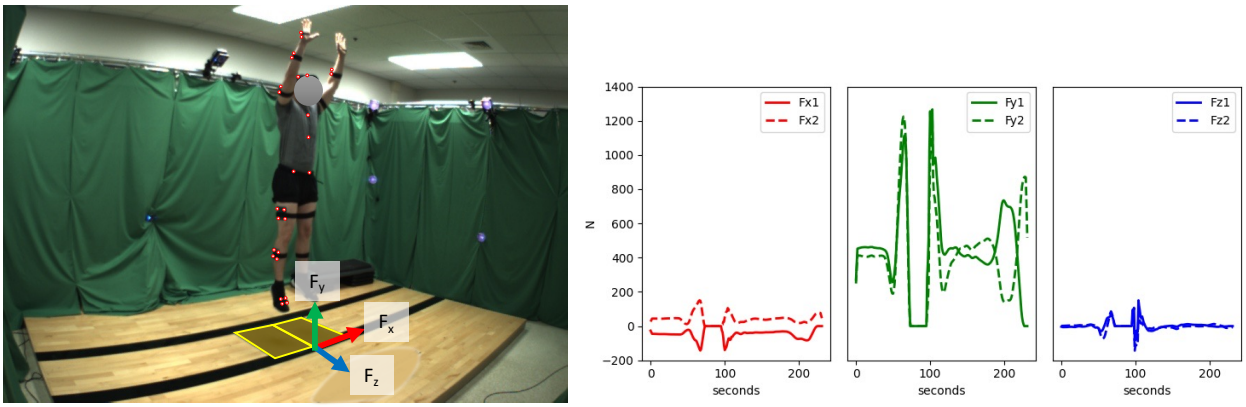


Figure 4.1: Standard practice for modeling human motion with forces requires physical reflective markers to capture kinematics and force plates to measure GRFs. In contrast, we propose a video-only approach that yields comparable performance yet does not require any physical apparatus to predict forces. Here the subject is performing a jumping movement on two force plates, measured forces are shown on the right. (Reflective markers are highlighted for visibility)

finetune paradigm, while multi-task learning optimizes both the 3D pose estimation and force prediction simultaneously.

The typical optimization recipe for GRF prediction uses mean-squared error (MSE) as a loss and root-mean squared error (RMSE) as a metric, but this overlooks a vital portion of the end task. For load analysis from motion, we care about the peak impact forces and how closely the predicted magnitude matches the ground truth. For example, an athlete is likelier to incur an injury during more active parts of a movements rather than standing still. While MSE and RMSE provide a good approximation for best curve fit, they unfairly weigh all parts of the force output. In periods of relatively no motion from the user, such as right before a jump, the predicted and ground truth forces will be easy to predict—relatively flat values. These "low activity" regions, in the learning process, often cause a muted (underestimated) approximation of peak impact forces to satisfy the flatter portions of the force curves. To account for this, we introduce a new loss function, gated-MSE, that prioritizes higher impact regions of the force prediction during training. We find that this leads to a small cost in RMSE but significantly decreases peak impact errors.

Last, we introduce *ForcePose*, a novel dataset to address the GRF prediction problem. There is an absence of publicly available data for this task, with LAAS Parkour [48] a single-view dataset of 28 videos only, being publicly available. In comparison, *ForcePose* is a multi-view dataset with eight subjects performing five movements for multiple trials, resulting in over $1,300$ videos with paired force plate data. We show that for extremely small datasets, such as LAAS Parkour, the 2D-to-3D pre-training provides up to 19% decrease in error compared to training from random weights. For small-to-medium-sized multiview datasets, such as *ForcePose*, employing a multi-task learning regime provides a moderate increase in performance when evaluating on the target task

and on unseen motions.

Our main contributions are as follows:

- We introduce gated-MSE as a new loss to minimize peak impact errors in GRF prediction.
- We show, using a transformer architecture, that pre-training and multi-task learning on 2D-to-3D human pose estimation:
  - Provides good initial weights for finetuning on small datasets;
  - Learns a useful representation for generalizing to novel motions.
- We provide the *ForcePose* dataset, a novel collection of multi-view tracked human motions with time-synchronized force plate data.

4.2 Data

4.2.1 ForcePose Dataset

We introduce the *ForcePose* dataset, consisting of eight subjects performing five movements for up to three different trials each. These movements are Counter Movement Jump (CMJ), Squat Jump, Squats, Single Leg Squat (SLS) and Single Leg Jump (SLJ) for both left and right legs. We show the camera views and screenshots of these movements in Fig. 4.2. To promote repeatability and consistency between all experiments, we train with six subjects while we reserving two for validation. The subjects are fitted with 47 reflective markers and there are two force plates to capture forces from the left and right foot. The average mass for training subjects is 88.37 ± 12.42 kg and 87.91 ± 10.74 kg including validation. There are 124 trials in training and 44 trials in validation, given eight camera views for each trial there are effectively $1,344$ videos. This gives us $227,640$ frames in training and $71,400$ frames in validation (including multiple views). The video data is captured using eight cameras at 50-Hz, with time-synchronized force plate data recorded at 600-Hz, and mocap data sourced from a 16 camera marker-based system captured at 100-Hz. We generate 2D pose COCO detections from the video data using a pre-trained Detectron [62] model, and then compose a pseudo-ground truth 3D pose label from the multiple views. With intrinsic and extrinsic parameters (K, T, R matrices) of the cameras, we can perform triangulation using a RANSAC-based algorithm to find the best point from all set of observations (viewpoints). The COCO 2D, 3D detections from all videos and time-synchronized force plate data are publicly available at `https://github.com/MichiganCOG/ForcePose`.

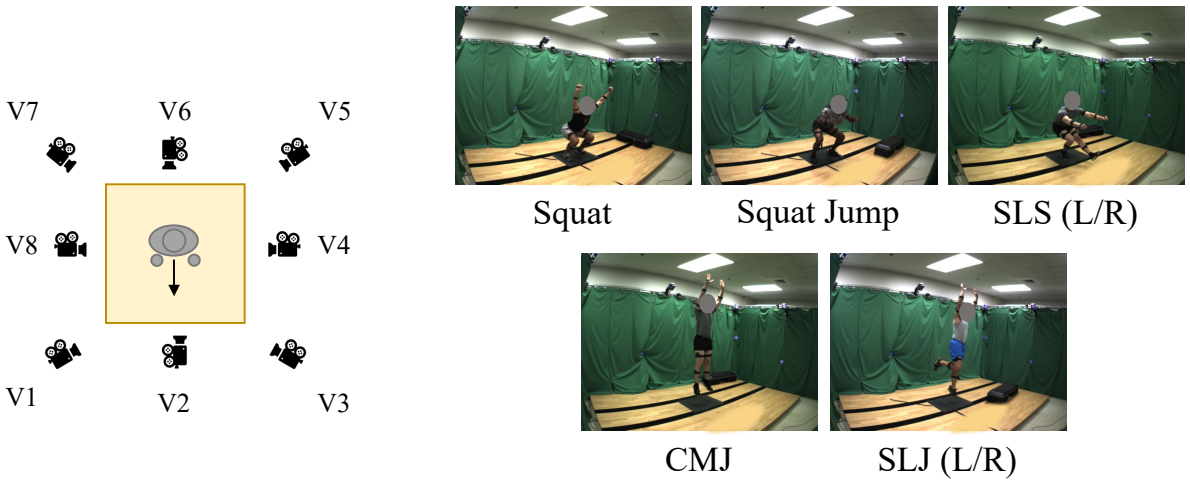


Figure 4.2: On the left, we show the (eight) camera views used from each trial and on the right, the (six) movements captured in the *ForcePose* dataset. (Single Leg Squat = SLS, Counter Movement Jump = CMJ, Single Leg Jump = SLJ).

4.2.2 LAAS Parkour Dataset

The LAAS Parkour dataset [48] is a publicly available dataset of five subjects performing four parkour movements: kong vault, safety vault, pull-up, and muscle-up. Each subject is fitted with 16 mocap markers, with force sensors capturing contact forces from the hands (L. Hand, R. Hand) and feet (L. Sole, R. Sole). Like prior work, a mass of 74.6 kg is assumed for all subjects. For fair comparison with existing work, and with only a single camera view available, we stick to 2D pose detections for these experiments. There are 28 videos in total, with an average of 83 frames per video. As we are only interested in GRFs, we report our performance on the L. Sole and R. Sole forces.

4.3 Method

We first propose a new loss function (gated-MSE) for learning GRFs, weighing significant regions heavier in the regressed curve. And we include an additional metric (mean k-peaks) to measure the error of impact peak forces, comparing the distances between k extrema points. RMSE and mean k-peaks distance metrics together provide a better picture of the characteristics of each prediction describing: (1) how well the overall force curve fits and (2) how closely peak magnitudes are approximated. Then we describe our process for training the GRF task on a transformer via pre-training and multi-task learning on the 2D-to-3D HPE subtask.

4.3.1 Losses and Metrics

Direct regression of forces using mean-squared error (MSE) as a loss function is the standard approach for learning to approximate ground truth signals in virtually all regression tasks. For measuring the best fit, root-mean-square error (RMSE) is most often used. MSE loss and RMSE metric tends to produce or capture a smoothed or averaged approximation of the ground truth signal. However, in GRF prediction, the instantaneous impact peaks of the force signal are more important than the averaged accuracy across the entire sequence. Therefore we introduce gated-MSE as a substitute for MSE loss, shown below as

$$\mathcal{L}_f = \sum_T w_t \cdot \frac{1}{|\mathbf{F}_{\delta_t}|} \|\mathbf{F}_{\delta_t} - \hat{\mathbf{F}}_{\delta_t}\|^2. \tag{4.1}$$

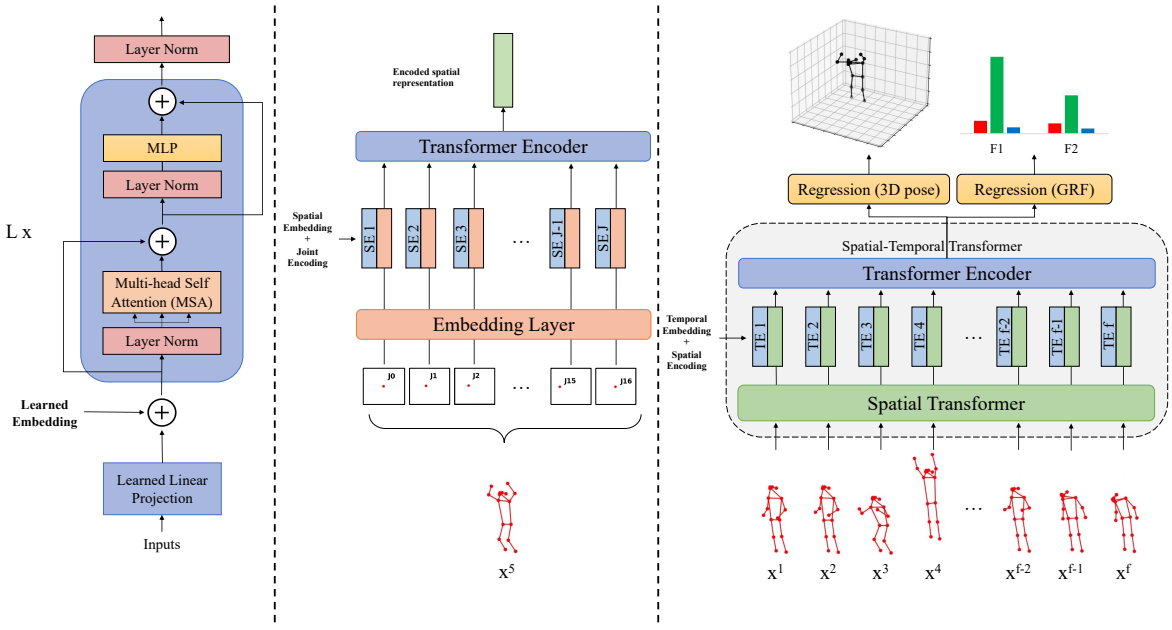
Gated-MSE can be viewed as a linear combination of weighted MSE at T thresholds. We define $\mathbf{F}_{\delta_t} = \{F_1, F_2, \cdots, F_f; \forall |F_f| \geq \delta_t\}$, which serves as an indexed array for all elements in the ground truth signal $\mathbf{F}$, with an absolute value above the threshold δ_n. In practice, T serves as a hyper-parameter where each threshold belongs to the sequence $\delta = [0, 1, 5, 10, 15, ...]$ in terms of N/kg. The summed loss is equivalent to the MSE when $T = 1$ and we weigh each contribution with $w_t = 1/T$, weighted by the number of thresholds.

RMSE is a useful measure for how well the overall force curve fits, but lacks insight in how closely peak magnitudes are approximated. We use mean k-peaks to provide this measurement. We extract k extrema (k peaks) and their temporal locations from the forces ($\mathbf{F}$ and $\hat{\mathbf{F}}$), then we compute mean k-peaks using averaged Euclidean distance, $\frac{1}{k}\sum_k \sqrt{x_i^2 + y_i^2}$, along each axis. Here, x represents the temporal distance and y the magnitude difference. This additional metric grades the matching capabilities to the high and lowest magnitudes of forces, in addition to a good fit.

4.3.2 Predicting Ground Reaction Forces

In our task, we take an input sequence, $X \in \mathbb{R}^{f \times (J \cdot C)}$, of 2D ($C = 2$) or 3D ($C = 3$) poses and predict the three-dimensional output force $\mathbf{F}$ at each frame. Here f is the number of frames, J represents the number of joints for the human pose, and C is the number of channels. We are predicting the magnitude forces on two force plates, so we use a 6D vector representation: $\mathbf{F} = [F_{x1}, F_{y1}, F_{z1}, F_{x2}, F_{y2}, F_{z2}]$. Here F_x, F_z represent the horizontal shear forces, F_y the vertical force, and $1, 2$ are for left and right force plates, see Fig. 4.1 for reference.

For training, we use a mini-batch of sliding windows and predict the force at the center frame $\lfloor f/2 \rfloor$. Forces are normalized by subject mass, hence units are in N/kg. But we report results in Newtons, multiplying the predicted value with the average training subject mass. We perform evaluation using mean k-peaks and average RMSE losses across sequences to obtain an Average



(a) Transformer Encoder (b) Spatial Transformer (c) Spatial-temporal Transformer

Figure 4.3: We show a brief overview of (a) the vanilla transformer encoder and the Spatial-Temporal architecture (b)-(c) [75] we used in our experiments for this work. This is composed of (b) a spatial encoder and (c) a temporal encoder. The input is a sequence of 2D poses and the output is a prediction of the 3D pose and corresponding GRF at the center of that sequence.

Sequence Loss shown as:

$$\mathcal{L}_{grf} = \frac{1}{|V|}\sum_{v}\frac{1}{|Cam|}\sum_{cam}\sqrt{\frac{1}{N}\sum_{i}\|\mathbf{F}_i - \hat{\mathbf{F}}_i\|^2}. \tag{4.2}$$

We average the RMSE across all camera views, Cam, within a sequence and then average across all video sequences, V.

Transformer Encoder We use a transformer encoder model to predict the forces, following the spatial-temporal architecture proposed by [75]. Inputs are tokenized at each time step, so a spatial transformer attends to intra-pose coordinates while a temporal transformer attends to inter-pose embeddings as shown in Fig. 4.3 (b)-(c). Like standard transformers, first each input is embedded using a learned linear projection layer, $\mathbf{E} \in \mathbb{R}^{(J\cdot C)\times D}$ summed with a learned positional embedding, $\mathbf{E}_{pos} \in \mathbb{R}^{f\times D}$, as shown in Eq. 4.3. Here D is the dimension of the embedding layer. Within the encoder, we then pass this embedding through L Multi-head Self Attention (MSA) layers [144] and L multilayer perceptrons (MLP) with layer normalizations (LN) and skip connections [138], [167] in-between, as shown in Eqs. 4.4 and 4.5. The final output, Y, is the Lth layer output of the encoder

after layer normalization.

The operations of the transformer encoder are shown below and in Fig. 4.3 (a):

$$Z_0 = [\mathbf{x}^1\mathbf{E}; \mathbf{x}^2\mathbf{E}; \cdots; \mathbf{x}^f\mathbf{E}] + \mathbf{E}_{pos} \tag{4.3}$$

$$Z'_\ell = MSA(LN(Z_{\ell-1})) + Z_{\ell-1}, \qquad \ell = 1 \ldots L \tag{4.4}$$

$$Z_\ell = MLP(LN(Z'_{\ell-1})) + Z'_{\ell-1}, \qquad \ell = 1 \ldots L \tag{4.5}$$

$$Y = LN(Z_L) \tag{4.6}$$

Eqs. 4.3-4.6 are repeated sequentially for both the spatial and temporal transformer. The temporal dimension of the final output $Y \in \mathbb{R}^{f\times(J\cdot C)}$ is reduced using a 1D convolution layer as learned weighted averaging, and is passed through a single MLP for the end task prediction.

Pre-training and Multi-task Learning Optimizing the performance of a transformer typically requires a lot of training data. To leverage data from a different task, we tightly couple 2D-to-3D HPE and GRF prediction, both benefiting from an implicit 3D representation of the input 2D pose. Pre-training on the 2D-to-3D HPE task will provide good initial weights and converge to a better solution on the GRF task, especially in situations with scarce amounts of data. The 2D-to-3D HPE task learns to estimate a 3D pose, $\mathbf{y} \in \mathbb{R}^{J\cdot C}$, at the center of an input sequence.

The loss function is MPJPE (Mean Per Joint Position Error), shown as

$$\mathcal{L}_p = \frac{1}{J}\sum_{k=1}^{J} \|y_k - \hat{y}_k\|_2, \tag{4.7}$$

between the ground truth and predicted 3D coordinates, averaged over the number of joints. Here k represents each joint index.

After training on 2D-to-3D HPE, we discard the final MLP layer and replace it with one to predict our 6D GRF vector. We repeat the same steps from the encoder shown in Eqs. 4.3-4.6, where the temporal elements of Y are again weighted averaged and passed through a single MLP layer to generate $\mathbf{F}$, $F \in \mathbb{R}^6$.

With multiple views we use multi-task learning (MTL) in place of the "train then finetune" paradigm. We optimize the predicted 3D pose concurrently with the 6D ground reaction force (with a separate MLP head), reducing the total training time while retaining solvability of the 2D-to-3D subtask. Our loss function includes both the 3D pose label and the force vector as a label for

Pose Format	Input	Prediction Network	RMSE (N)	1-peak (N)	3-peak (N)	5-peak (N)
-	Movement Category	Naive baseline (GRF Exemplar)	153.84	585.28	327.35	247.86
COCO	Ankle Keypoints	Naive baseline ($\mathbf{F} = m * \mathbf{a}$)	206.58	624.62	393.92	317.74
Mocap markers	Triangulated 3D keypoints	LSTM	**77.04** ± 0.51	322.10	199.20	165.09
Mocap markers	Triangulated 3D keypoints	LSTM (our loss, T=2)	77.74 ± 0.43	**313.34**	193.60	160.19
Mocap markers	Triangulated 3D keypoints	Transformer	78.15 ± 1.00	373.48	217.58	175.92
Mocap markers	Triangulated 3D keypoints	Transformer (our loss, T=2)	83.50 ± 2.97	328.21	**193.11**	**153.72**
COCO	Triangulated 3D keypoints	LSTM	81.12 ± 1.52	347.46	209.14	172.72
COCO	Triangulated 3D keypoints	LSTM (our loss, T=2)	82.91 ± 1.95	**317.63**	191.62	158.04
COCO	Triangulated 3D keypoints	Transformer	**77.90** ± 1.66	340.95	199.38	157.79
COCO	Triangulated 3D keypoints	Transformer (our loss, T=2)	79.36 ± 2.19	336.59	**186.43**	**148.35**
COCO	2D keypoints	LSTM	98.14 ± 1.81	358.17	219.51	184.02
COCO	2D keypoints	LSTM (our loss, T=2)	103.02 ± 2.88	302.73	192.44	163.26
COCO	2D keypoints	Transformer	**77.95** ± 0.36	321.19	190.14	154.03
COCO	2D keypoints	Transformer (our loss, T=2)	83.20 ± 3.80	**285.72**	**171.71**	**139.23**

Table 4.1: Average Sequence Losses and mean k-peaks on the *ForcePose* dataset, measured in Newtons

supervision. This is directly a summation of Eq. 4.1 and Eq. 4.7 shown as:

$$\mathcal{L}_{mt} = \mathcal{L}_f + \alpha \mathcal{L}_p. \tag{4.8}$$

Here L_f is the loss for the GRFs, L_p is for 2D-to-3D HPE, and α is a hyper-parameter. We find $\alpha = 1$ to provide the lowest Average Sequence Loss overall.

4.4 Experiment Details

We experiment with three types of keypoint inputs: 2D, 3D (mocap), and 3D (triangulated). 2D keypoints are generated using a pre-trained Faster-RCNN (ResNet-101) network from Detectron [62]. We use the default COCO pose format (17 keypoints) with no additional finetuning on the detector. 3D keypoints from motion capture (mocap) physical markers are unique to each dataset, hence they do not have a one-to-one correspondence with the COCO format. With multiple camera views, we construct a triangulated 3D pose from 2D detections as a pseudo-ground truth 3D pose label. These labels are directly associated with their corresponding COCO keypoints.

When training the LSTM baseline, like [34], we implement a simple bi-directional network with a hidden dimension of 64-1024 (selected through hyper-parameter search), and an MLP encoding layer with a hidden dimension of 256. We also vary the input sequence length for the network and report only the hyper-parameters leading to the lowest Average Sequence Loss. The LSTM networks are trained for 40 epochs using an Adam optimizer, a learning rate of $1e^{-4}$, and a batch size of 64. We use the transformer architecture PoseFormer [75] as the base model for our experiments in predicting GRFs from 2D and 3D inputs. For the Transformer encoder, we retain many of the

default settings using an embedding dimension $D = 32$, 8 attention heads, and 4 depth layers. From this point, the training details change slightly as we optimize for each input data type and target task, typically until overfitting occurs. 3D (mocap) inputs are trained for 25 epochs and a learning rate of $4e^{-4}$, 3D (triangulated) inputs for 50 epochs and a learning rate of $4e^{-4}$, and 2D inputs for 50 epochs with a learning rate of $4e^{-4}$. When pre-training on the 2D-to-3D HPE, we run our model for up to 100 epochs and a learning rate of $4e^{-6}$. All training was conducted with an exponentially decaying learning rate, decaying by a factor of 0.99 every epoch, and a static batch size of 512. We include horizontal flipping of the poses as data augmentation.

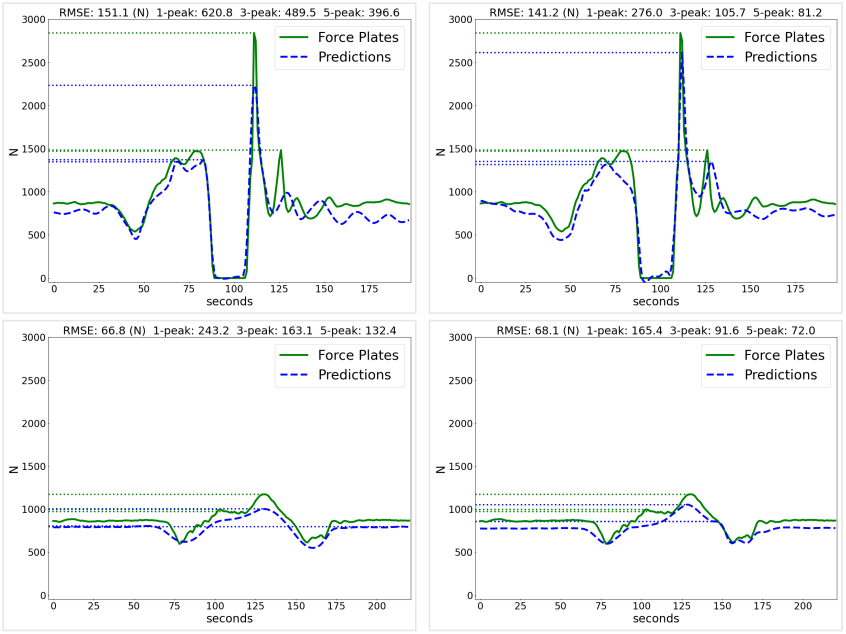
4.5 Results

4.5.1 *ForcePose*

We show results on the *ForcePose* dataset, measuring performance using RMSE for Average Sequence Losses and mean k-peaks as shown in Table 4.1. Each experiment is run for three trials and we report the mean and standard deviation errors. As mentioned in Sec. 4.4, we conduct training with three types of keypoints: 3D (mocap), 3D (triangulated), and 2D. The mocap markers represent the industry standard, hence should result in the lowest error (*i.e.* a soft upper bound). The triangulated 3D keypoints (from 2D detections) require at least two viewpoints and the 2D keypoints require only a single view.

We include two naive baselines for comparison. The first generates an exemplar force profile by averaging the ground truth force of each movement on the training set. But due to high inter-subject variance in force magnitudes and motion timing, the exemplars often do not resemble the typical force profiles. The second baseline uses Newton's Second Law, $F = m * a$, to estimate the total force. With the known frame rate, we estimate the acceleration of the subject using the triangulated 3D ankle keypoint positions. This method is very susceptible to noise and world coordinate scaling errors, and single leg movements introduce difficulties in the computation. These baselines show very high error on mean k-peaks and an overall worse fit in terms of RMSE.

We compare the performance of the LSTM and transformer architectures across different keypoint input types and then with our gated-MSE during training. The LSTM initially outperforms the transformer on mocap markers, but as the keypoints get noisier (less perfect) and data samples increases with multiple views, the transformer architecture outperforms an LSTM. This difference is much greater on 2D keypoints, with 20 N lower error than the LSTM, a reduction of 20.5%. The most significant result here is that the performance between the mocap markers and detected COCO keypoints are quite competitive. This highlights that physical markers can be forgone in lieu of pose detections, which are easier to produce and readily available. Next, we analyze the impact of our loss across these experiments. Our primary goal is to lower the error in impact peaks, as



(a) MSE Loss (b) Our Loss

Figure 4.4: We compare the net GRF outputs on the Single Leg Jump (R) (top row) and Squat (bottom row) trained using the (a) MSE Loss and (b) gated-MSE. The force plates are shown in green and predicted forces are shown in blue. We show much smaller differences across the mean k-peaks when compared to the ground truth force plates. Horizontal lines mark the detected peaks in each plot, for brevity we only show the top-3.

measured by mean k-peaks. We note that while we significantly lower the peak errors in the GRF prediction, this generally comes at a cost in RMSE. But we find that this cost is very low about $1-5$ Newtons, while the average decrease in just a single peak ranges from $10-56$ Newtons. Through a hyper-parameter search, we find $T=2$ provides the best trade-off between RMSE and minimizing k-peak errors. We reserve additional details on loss thresholds and range of input receptive fields for the Supplementary Material.

There is currently no definitive research on acceptable GRF error for downstream biomechanical analysis, as motivated in our introduction. However from US Customary Units we can use the conversion 1 lb. $=4.448$ N, to better understand the significance of the errors. For example in Fig. 4.4, we compare the summed GRF outputs from a model trained on MSE loss versus our new gated-MSE. In Fig. 4.4 (a) Single Leg Jump, we decrease the peak magnitude error by 344.8 N which is approximately 77.5 lbs. of force and the decrease of the average 5-peaks is still very high, about 70 lbs. of force. This drastic difference shows how vital minimizing these instantaneous errors can be, especially when being used for analysis on the human body. In the following section, we compare our results on the transformer when pre-training and multi-task learning on a different subtask.

Input	Prediction Network	RMSE (N)
2D Keypoints	Transformer random weights	80.45 $\pm$1.08
2D Keypoints	Transformer pre-trained (ours)	80.51 $\pm$0.61
2D Keypoints	Transformer MTL (ours)	**77.95** $\pm$0.36

Table 4.2: Average Sequence Losses on the *ForcePose* dataset. We compare results from transformers using 2D-to-3D HPE as a subtask

Method	L. Sole (N)	R. Sole (N)
Li *et al.*[48]	144.23	138.21
LSTM	99.47	94.38
Transformer random weights	103.48	95.08
Transformer pre-trained (*ForcePose*)	91.47	89.67
Transformer pre-trained (H36m)	**83.78**	**85.79**
Transformer MTL	91.74	99.78

Table 4.3: LAAS Parkour Dataset. Estimation errors of forces (in Newtons). Each subject has an assumed mass of 74.6 kg

2D-to-3D subtask We introduce 2D-to-3D HPE as a subtask in training our transformer via pre-training and multi-task learning (MTL), results are shown in Table 4.2. Leveraging the multiple views of *ForcePose*, we can use the triangulated 3D poses as pseudo-ground truth labels. Our MTL implementation performs best across the three training strategies, while training using pre-trained and random weights show similar results. Optimizing for the 3D pose simultaneously with 3D GRFs provides an advantage because the model maintains a constant three-dimensional representation from the 2D keypoint inputs. This additional constraint puts it on par with the other models that operate directly on 3D keypoints, as shown in Table 4.1. On the contrary pre-trained and random weights appear to converge to a similar solution, which we can attribute to the variance and size of the *ForcePose* dataset. Because the data is large and varied enough for the target GRF task, when trained for enough iterations, the initialization from the pre-training becomes less significant. We see in Sec. 4.5.2 that when the target dataset is much smaller, the pre-training has a huge impact on the final results.

Zero-shot learning With 2D keypoint inputs on the transformer, we measure our ability to generalize to unseen motion using zero-shot learning. We employ leave-one-out cross-validation; training on four movement classes and testing on the unseen class. We compare training with random weights against pre-training and multi-task learning (MTL) on the 2D-to-3D HPE task,

Method	CMJ	SLS	SLJ	Squat Jump	Squat
Random weights	117.40 $\pm$2.91	137.76 $\pm$4.82	219.95 $\pm$8.41	104.28 $\pm$4.34	65.77 $\pm$6.22
Pre-trained weights	111.62 $\pm$2.45	140.17 $\pm$4.03	**207.01** $\pm$3.68	**104.23** $\pm$1.19	55.15 $\pm$1.29
Multi-task learning	**110.29** $\pm$4.59	**119.62** $\pm$1.54	216.92 $\pm$2.57	104.60 $\pm$6.76	**53.37** $\pm$7.79

Table 4.4: Zero-shot learning, RMSE measured in Newtons

reporting our results on Table 4.4. Each column represents the testing (leave out) class. The advantages of using pre-trained weights or MTL are apparent in all classes except Squat Jump, with Squats having the lowest error overall. Grouping the classes into squatting and jumping movements, we notice smaller margins between training strategies for jumping, with the exception of single leg. The jumping motions may appear visually similar, so the additional 2D-to-3D HPE subtask may not provide much more discriminative power, over random weights, on the unseen class. However, across the variations in the classes, single leg movements are most visually distinct and difficult to generalize to. The subject may position the leg far out front when squatting or further behind them when jumping, deviating far from what has been seen during training. Interestingly enough, while the performance between pre-training and MTL is generally close, we see a swap between Single Leg Squat and Single Leg Jump. But when generalizing to newer movements, we find that MTL provides the most advantages and lowest errors across the board. We include additional comparisons and analysis between these training strategies in the Supplementary Material.

4.5.2 LAAS Parkour

We report results on the LAAS Parkour dataset, shown in Table 4.3. Given the small dataset size, we train and evaluate using leave-one-out cross-validation by subject across all movements. For comparability with other GRF experiments, we only predict the contact forces for the L. Sole and R. Sole. Following [48], the final accuracy is the average L2 distance across all videos. Using the same transformer we perform training on random weights, then finetuning from pre-trained weights and multi-task learning on the 2D-to-3D HPE task (*ForcePose* and Human3.6m [37] datasets). The Human3.6m (H36M) dataset is a large mocap dataset with 3.6 million frames and 17 actions but no force data. We use this pre-training only on LAAS to showcase the impact of much larger datasets when finetuning on smaller datasets. But we maintain COCO detections throughout other experiments for consistency. There is not a one-to-one mapping between the COCO keypoints and the H36M mocap markers, hence the transformer must learn some keypoint re-mapping during the finetune stage.

We compare to prior work by Li *et al.*[48], where they re-project detected 2D joint locations to 3D positions and solve a large-scale trajectory optimization problem to learn contact forces and 3D

Pose Format	Input	Prediction Network	RMSE (N)
COCO	3D Keypoints	LSTM	75.95 ± 0.43
COCO	3D Keypoints	Transformer	**68.66** ± 2.24

Table 4.5: Updated results on *ForcePose* version 1.1.

motions. We show that the baseline LSTM network outperforms the previous SOTA baseline with a 30% decrease in error. Motivating the strategy of pre-training on the 2D-to-3D HPE task, we note a $9-19\%$ decrease in error between that and a transformer trained from random weights. With the LAAS Parkour dataset being very small, unlike the results shown in Sec. 4.5.1, pre-training on 2D-to-3D HPE demonstrates a tremendous advantage from using the *ForcePose* and Human3.6m datasets. This trend is sustained even with the keypoint mismatch between H36M and COCO pose formats, showing the lowest loss overall. The transformer training from random weights performs worse than the LSTM baseline, due to the small number of data samples paired with a more complex network architecture. And MTL training performs well on the L. Sole, but much worse on the R. Sole. We believe this may originate from restrictions in the dataset, which only supports the same single view for all videos. In 68% of the trials, the subject is facing to the left which may bias the 3D pose estimation branch of the MTL.

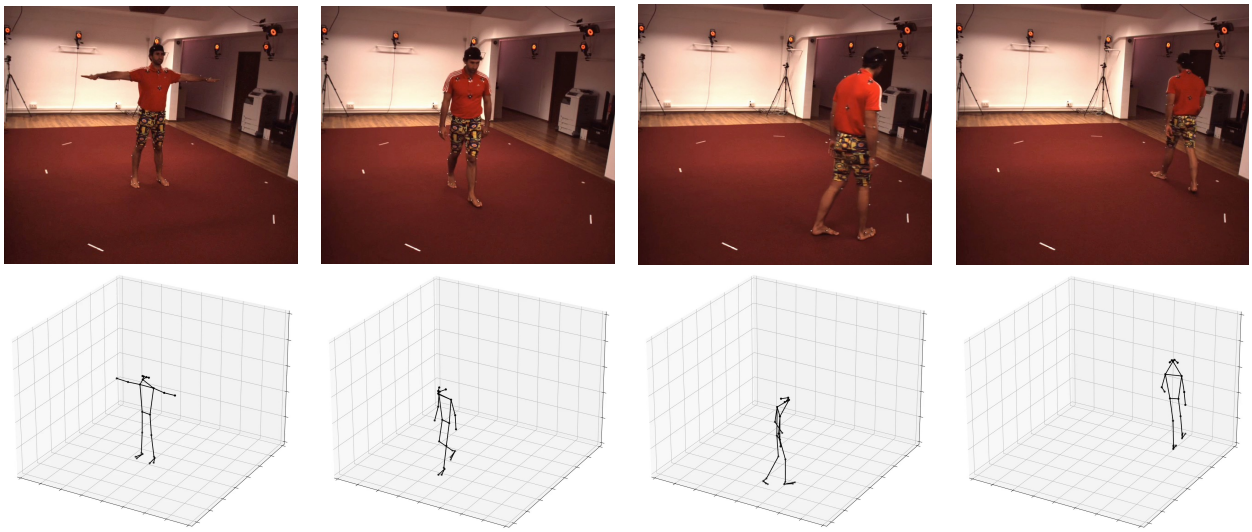
4.6 Extension to Experiments

4.6.1 ForcePose version 1.1

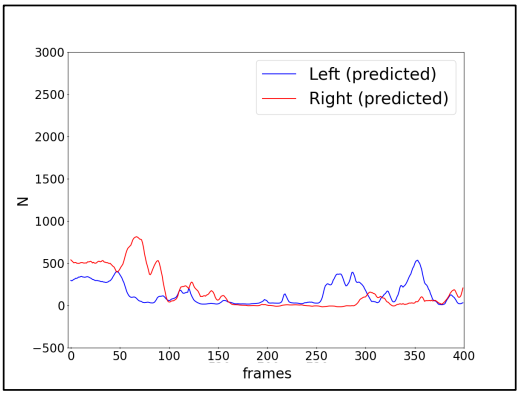
We identified a small misalignment between the RGB frames and force plates. Due to minor timing differences in hardware, the RGB video has delayed start by 200ms, or 10 frames, from the force plates and marker-based camera system. We resolved this by truncating the force plate data and re-aligning with the video frames. Additionally, we updated the pose format with toe and heel key-points using Whole-Body detections from [168]. With two key-points for the big toe and small toe and one key-point for the heel on each foot, this increases the total joints of the triangulated 3D pose from 17 to 23. Our updated results using the new sequences are shared in Table 4.5. This demonstrates a proportional decrease in error for both models.

4.6.2 Data augmentation for generalization

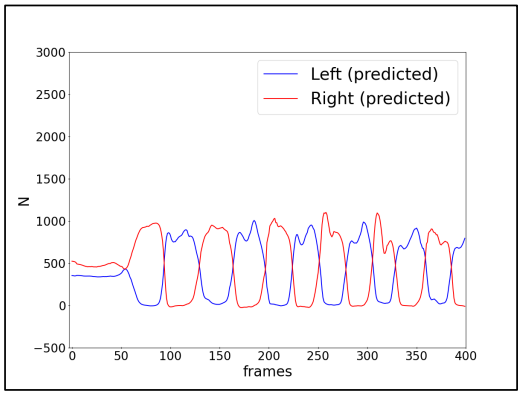
We analyze our ability to produce plausible outputs on the force-less video sequences in Human3.6M. These video sequences do not have ground truth forces. However in Figure 4.5 (b), we see that the training of our models produces a bias for force prediction based on the spatial



(a) *S9 – Walking 1* sequence



(b) No data augmentation



(c) Translation and Rotation Augmentation

Figure 4.5: We demonstrate qualitative contact force results (b)-(c) on the *S9 - Walking 1* motion from the Human3.6m dataset (a). In (b), we see predicted begin to attenuate as subject moves away from the starting position or rotates around the y-axis. While in (c), including translation and rotation augmentations on the poses produces a plausible force magnitudes.

position of the subject. We apply this to the *S9 - Walking 1* motion from Human3.6M. To address this, we experiment with translation and rotation data augmentations on the input poses during training. This consists of a random rotation about the $y-$axis (vertical) between 0-360 degrees and a random translation of 2000 units along the x and z-axes (horizontal plane). In Figure 4.5 (c), we see that this results in more plausible contact forces. However, this has a negative impact on the performance on *ForcePose*. For the LSTM model, the RMSE increases from 75.95 to 81.19. This may indicate that our limited dataset may not be generalizable enough to represent performance on a wider range of motions.

4.7 Conclusion

In this chapter we present a new approach for solving the GRF prediction from video problem. First, we introduced the *ForcePose* dataset, a large collection of multi-view tracked human motions with paired force plate data. Then we addressed the minimization of peak impact errors by introducing gated-MSE which drastically reduces peak impact errors at a low cost in RMSE. We also show, using a transformer architecture, that pre-training and multi-task learning on 2D-to-3D human pose estimation induces better performance when fine-tuning on small datasets and has better generalization to unseen motions. This work takes a great step towards analyzing the quality of human motions and actions. By estimating these values directly from video we can use them as apart of an automated process for comprehensive analysis.

CHAPTER 5

Physical Plausibility as a Metric for 3D Human Pose Estimation

5.1 Introduction

Physically grounding objects from video can be vital for understanding spatial relationships [169], geometric properties [170], and body forces [171], [172]. For humans in particular, this allows us to understand how a person interacts within an environment and models their actions to be evaluated for therapeutics, sports, or entertainment purposes (*e.g.*motion capture) [123], [173], [174]. To physically ground a person from video, the common learning-based approach is to capture a 3D representation using state-of-the-art 3D human pose estimation (HPE) methods [65], [67], [72], [75].

But even with the impressive pose estimation progress on computer vision datasets [37], [175], [176] and attaining lower MPJPE—the primary quantitative metric—there are notable flaws in the pose estimates, such as floating, foot-skating, ground penetration and unnatural position [50]. Qualitatively, in these situations, the pose estimates often appear to be physically implausible. However, the standard MPJPE metric does not take into account global translation in world space and often displays little correlation between low errors and visual quality of results [49].

These discrepancies led us to question the physical plausibility, hence physical grounding, of these predicted 3D human poses. Per prior studies [177], [178], humans are particularly sensitive to motions that lack perceptual realism or contain slight physical errors. Consequently, applications and systems involving augmented or virtual reality and physical understanding of a scene may be negatively impacted by how humans perceive these implausibilities.

To overcome these limitations in the current practice of evaluating pose estimates, we propose a unique approach to evaluate the physical plausibility of 3D HPEs using rigid body physics simulation. Physically plausible 3D poses obey the laws of physics and produce stable, realistic motion. Contrarily, physical simulation computes the effects of friction, gravity, collisions, and temporal consistency to encourage realistic motion.

We hypothesize a positive correlation between the physical plausibility of a sequence of 3D poses and stability during physical simulation. The more stable the simulation, the greater the

likelihood of it being physically plausible. Assume a 3D skeleton from an existing 3D HPE method, estimated reference joint angles, and a controller to actuate the joint angles on a simulated body. We apply trajectory optimization on the reference joint angles to find the optimal trajectory that maintains the reference motion. This process then facilitates concrete evaluation of the physical plausibility of the input 3D HPE. We measure the simulated body stability from the result of this trajectory optimization. We analyze the distance of the center-of-mass trajectory, assessing its deviation from the reference, and evaluate both static and dynamic stability by determining the extent to which a pose can be simulated before reaching an irrecoverable failure. These new metrics have the potential to unlock a greater understanding of real progress in HPE.

Current HPE Evaluation MPJPE uses the Euclidean distance to measure the closeness of two 3D poses and is regarded as the *gold-standard* for evaluating 3D HPE. However, it requires ground truth 3D poses for evaluation and is not indicative of plausibility. Ideally, a physical plausibility metric does not require ground truth 3D poses. Identifying these issues, there has been existing work [50]–[52], [65], [89], [93] on physics-based metrics for 3D HPE. To measure realistic contact with the ground, footskate [50], foot slide [51], and ground penetration [51], [52] have been proposed.

To highlight unnatural jittering, some works [51], [52], [65], [89] have computed smoothness losses from the velocity or acceleration of joints. For stationary stability, Shimada *et al*.[52] and Tripathi *et al*.[93] introduce terms to measure balanced and unbalanced static postures. However, these terms do not model the progression of instability and implausibility errors, or how earlier faults impact the stability of latter poses. In Figure 5.1, we show an example of how a slight unnatural lean in a predicted pose can eventually lead to a loss of balance in simulation. Rigid body physics simulation can model these dynamic instabilities, and subsume the previously proposed metrics through gravitational forces to prevent floating, collisions with the ground plane, self-collisions, and inherent temporal consistency.

Contributions A plausible 3D HPE output should account for physical phenomena such as contact, collisions, friction, and gravity. We use a rigid body physics engine to simulate these interactions and model physical dynamics, without relying on training data. This makes it an ideal candidate for measuring the physical plausibility of 3D HPEs. In addition to the stability of a dynamic pose, we can also examine the quality of produced external contact forces. These components give us a holistic view of the 3D HPE within a real environment, in comparison to prior methods. Our contributions are two physical simulation-based metrics for evaluating the physical plausibility of 3D human pose estimations and a comprehensive analysis of existing publicly available state-of-the-art approaches and baselines. All simulation and evaluation code will be publicly released upon publication.

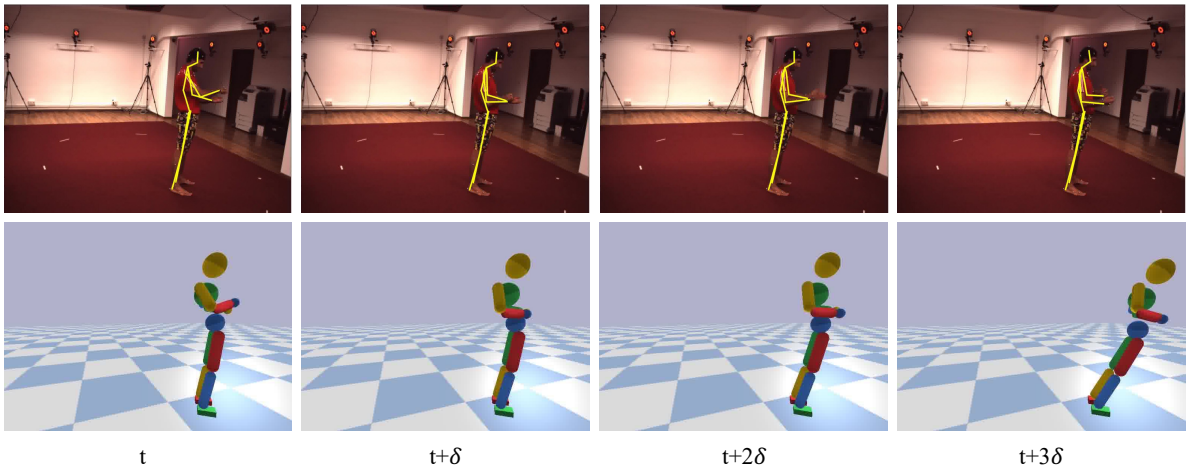


Figure 5.1: The top row shows a reprojected 3D prediction on the Human3.6M dataset (*S9 - Directions 1*). While the error is relatively low on current metrics (MPJPE = 49.0mm, FS = 0%, GP = 0.59mm), with our physical plausibility simulator (bottom row), we see that a slight unnatural lean eventually causes a loss of balance.

5.2 Method

Our main focus is a new evaluation method for the physical plausibility of 3D human pose estimates (HPE). To do that, we assume a given 3D HPE. Evaluating the plausibility of this HPE under physical simulation itself requires a computational method. We first describe this method, which is a two-stage process of initializing the kinematics of the simulation from the input and then optimizing the pose-trajectory under physical constraints. Finally, in Section 5.2.3 we describe new metrics that leverage this physical simulation to quantitatively summarize the input HPE's plausibility. An overview of our method is shown in Figure 5.2.

5.2.1 Kinematic Initialization

We begin with a sequence of 3D skeletal poses, $X \in \mathbb{R}^{T\times J\times 3}$ from any 3D HPE method, with J joints and T frames. For models generating a mesh such as SMPL [80], we can use a pre-trained regressor to estimate the joint locations. We apply minimal preprocessing to reduce noise in X, first a median filter ($w = 15$ frames) and then constraining bone lengths to their averaged value. We assume a horizontal ground plane in our sequences and estimate the floor height to the pose by averaging the $k(= 0.05T)$ lowest joint values.

Next, we must convert the 3D poses, X, into a sequence of target joint angles, $\mathbf{q}^k_{1:T}$, and joint velocities, $\dot{\mathbf{q}}^k_{1:T}$, to be applied to our simulated body. Each $\mathbf{q} = (q_1, q_2, ..., q_D)$ is a kinematic representation of the pose as a concatenation of all joint angles, parameterized as quaternions, for D degrees-of-freedom (DOF). Solving an inverse kinematics problem is a common approach for

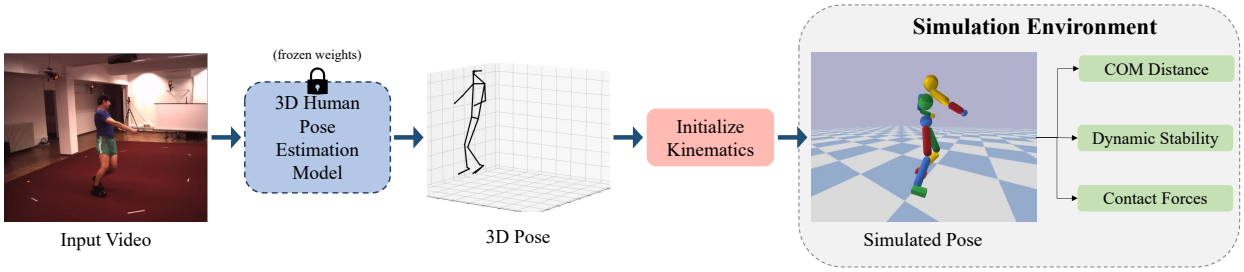


Figure 5.2: To analyze the physical plausibility of 3D human pose estimates, we first extract a 3D skeletal pose from the output of a 3D HPE method. We then initialize kinematics through the estimation of reference joint angles, floor height, and normalizing joint segment lengths. We apply the joint angles directly to a simulated body within a simulated environment and optimize the joint angles to imitate the reference motion under simulator constraints. We measure the approximate plausibility of this optimized output by analyzing COM trajectory distance, dynamic stability, and contact forces.

estimating $\mathbf{q}^k_{1:T}$, however, the ill-posed nature of this problem may result in substantially different and even implausible poses. Similar to [90], we adopt a simpler method of first defining a kinematic tree and then using change of basis rotations, from the root to the end effectors, to approximate the joint rotations. This assumes both the skeletal pose and the body have similar kinematic trees. The resulting output, $\mathbf{q}^k_{1:T}$, is a direct kinematic representation of the 3D pose sequence without alteration from an inverse kinematics solver. The joint velocities, $\dot{\mathbf{q}}^k_{1:T}$, are computed using finite differences.

Our simulated body is a humanoid with 28 degrees-of-freedom (DOF) and 12 controllable joints, and an assumed mass of 45kg. This includes eight spherical joints with 3 DOFs each and four revolute joints (knees and elbows) with 1 DOF. Each joint is paired with a Stable PD controller [179], which takes as input target joint angles, joint velocities, and gains (k_p,k_d) to output a torque that actuates each joint. At each time step, the torques are computed as

$$\tau = k_p(\mathbf{q}^k - \mathbf{q}) + k_d(\dot{\mathbf{q}}^k - \dot{\mathbf{q}}). \tag{5.1}$$

Here, $\mathbf{q}^k$ represents the target kinematic pose while $\mathbf{q}$ is the current kinematic pose, and k_p, k_d are the proportional and derivative gains. The gain values are fixed across all experiments. The root node (pelvis) is excluded as a controllable joint, since applying forces directly to the root of the kinematic tree is not realistic and lacks physical meaning [171].

5.2.2 Trajectory Optimization

Even with accurate kinematic initialization, differences in physical and environmental attributes will cause the simulated motion to deviate from the reference motion [90], [155]. Following

Method	Input	Model	Phys.	MPJPE
PoseFormer [75]	S	Skel.	✗	44.3
NeuralPhysCap [84]	S	Skel.	✓	76.5
Baseline	MV	Skel.	✗	-

Table 5.1: 3D human pose estimation methods used for comparison and published scores on the Human3.6M dataset. S = Single image, MV = Multiview.

best practices in physical simulation [89], [154], [155], we perform trajectory optimization to optimize the kinematic joint angles to mimic the original reference motion on the simulated body, this can be viewed as a variant of motion retargeting [155]. We run the simulator at 1kHz and set controller targets to 25Hz. Given that physical simulators are highly advanced and generally non-differentiable, we use the derivative-free Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [180] algorithm for optimization. Optimization is performed on overlapping windows of 0.5s, for 8 windows in total, 200 iterations, and a population size of 100.

We reduce the search space by representing the targets, as Euler angles, in a cubic B-spline and optimizing the knots instead of all time steps simultaneously. We use the following cost functions:

The center-of-mass (COM) loss,

$$L_{COM} = \sum_t \left(\mathbf{c}_t^k - \mathbf{c}_t\right)^2, \tag{5.2}$$

is the distance between the COM of the kinematic target, $\mathbf{c}^k$, and the current COM, $\mathbf{c}$, from the simulated body.

The center-of-mass velocity loss,

$$L_{COMv} = \sum_t \left(\dot{\mathbf{c}}_t^k - \dot{\mathbf{c}}_t\right)^2, \tag{5.3}$$

analogously is the distance between COM velocities of the kinematic target and the current velocity.

The root orientation loss,

$$L_{orn} = \sum_t \arccos\left(\left|\left\langle \mathbf{q}_{root}^k, \mathbf{q}_{root} \right\rangle\right|\right), \tag{5.4}$$

constrains the pelvis orientation of the body to align with that of the kinematic reference. The distance is computed from the $\arccos$ of the dot-product between these two quaternions. Where $|.|$ denotes the absolute value.

The kinematic pose loss,

$$L_{pose} = \sum_t W_j * \left(\mathbf{q}_t^k - \mathbf{q}_t\right)^2, \tag{5.5}$$

and the kinematic pose velocity loss,

$$L_{vel} = \sum_t W_j * \left(\dot{\mathbf{q}}_t^k - \dot{\mathbf{q}}_t\right)^2, \tag{5.6}$$

minimize the pose and pose velocity difference. We optimize these two losses by parameterizing the joint angles using Euler angles rather than quaternions. W_j is the joint weighting, with higher weights given to the hips and shoulders due to their bigger impact on the end effector positions.

And finally the joint acceleration loss,

$$L_{acc} = \sum_t \|\ddot{\mathbf{q}}_t\|^2. \tag{5.7}$$

to encourage a smoother trajectory and avoid jittery motions. We introduce a contact loss,

$$L_{feet} = \sum_t \|\mathbf{p}_t^k - \mathbf{p}_t\|_1, \tag{5.8}$$

to encourage alignment of foot contacts with those of the input pose. We estimate the kinematic foot contacts, $\mathbf{p}^k$, (and ground plane) as described in Section 5.3.3. For the simulated body foot contact, $\mathbf{p}$, we use a small height threshold (0.0005) from the position of the feet. We observe that this constraint helps to maintain balance when shifting weight between feet.

The total optimization objective function is a linear combination of the aforementioned losses with the following weights $w_{COM} = 20$, $w_{COMv} = 0.5$, $w_{orn} = 1.0$, $w_{pose} = 1.0$, $w_{vel} = 5e^{-3}$, $w_{acc} = 1e^{-10}$, $w_{feet} = 1.5$.

5.2.3 Simulation-based Metrics

After completing the trajectory optimization, we measure physical plausibility within the physical simulator using COM trajectory distance and dynamic stability.

Metric 1: COM Trajectory Distance The COM trajectory distance is relatively straightforward. We measure the L2 distance between the kinematic COM trajectory and the final optimized COM,

$$D_{COM} = \frac{1}{T} \sum_t \|\mathbf{c}_t^k - \mathbf{c}_t\|_2, \tag{5.9}$$

in millimeters (mm). Comparing the final trajectory with the kinematic reference indicates how well the original 3D pose estimate can be simulated. A low value suggests the reference is likely plausible and easy for the simulated body to follow, while a high value indicates significant deviations, possibly due to simulation failure or necessary adjustments for a similar result.

Metric 2: Dynamic Stability For the instance in which failure occurs, we introduce dynamic stability, defined as

$$\text{Dyn}_T = \min(T - n, t_d). \tag{5.10}$$

T is the total number of frames in each sequence, t_d is the frame at which the dynamic instability occurs, otherwise $t_d = T$, and n is the number of static instability instances. By examining both stationary poses and poses undergoing locomotion, we assess the maximum number of simulated frames before a catastrophic failure, *i.e.*, an unrecoverable deviation from the reference motion. We use the kinematic COM velocity at each time step, $|\dot{\mathbf{c}}_t^k| <= 250$mm/s, to classify a stationary or dynamic pose. From biomechanics literature [94]–[96], it is commonly accepted that a stationary pose is considered stable or balanced if its center of gravity falls within its base-of-support (BoS), *i.e.*the convex hull of all ground contacts. We compute the BoS using coordinate locations of the foot, with a small boundary buffer, and the center of gravity by projecting the COM onto the ground plane. Occurrences where the simulated body is stationary but the center of gravity is beyond the convex hull, are marked with n, acknowledging that minor balance loss can be corrected through slight adjustments. The second case of dynamic instability involves the simulated body falling to the ground, indicating a catastrophic failure. We acknowledge exceptions where this may require prior knowledge of a motion. For instance, the hands and torso touching the ground during push-ups is an example case that is not considered a failure. This can be addressed by adjusting constraints derived from all points of contact within the kinematic reference. Nevertheless, our experiments exclusively focus on upright poses.

5.3 Experiments

Here we discuss the effectiveness of our new physical plausibility metric on state-of-the-art 3D HPE data and methods.

5.3.1 Evaluation Dataset

We evaluate the 3D HPE models on Human3.6M [37], a video dataset capturing human actions using 4 cameras and a motion capture system. We use the validation subjects *S9, S11*, and the same subset of actions, *Directions, Discussion, Greeting, Posing, Purchases, Photo, Waiting, WalkDog,*

WalkTogether, and Walking, as prior work [52]. We downsample the videos from 50fps to 25fps and run our evaluation on 100 frames. The full camera projection matrices are assumed known. While the intrinsic parameters are often estimated, the extrinsic parameters are required to accurately transform the 3D predictions from the camera frame to the world frame.

5.3.2 Evaluation Models

We use three methods for comparison, shown in Table 5.1, along with published MPJPE results on H36M (if applicable). PoseFormer [75] is a spatio-temporal transformer that produces a 3D pose estimate from a sequence of 2D pose detections. We use the transformer weights trained on 2D CPN pose detections [181] and 81 frame sequences. PoseFormer does not output a global trajectory for the 3D HPE, therefore we estimate the 3D global position by minimizing the 2D re-projection loss from a differentiable projection function. NeuralPhysCap [84], proposes a differentiable framework that generates physically plausible poses through contact estimation, force estimation, and physics-aware optimization. This method generates a 16-joint skeleton, slightly different from the 17-joint skeleton in H36M. We only use the mutual joints for computing MPJPE. We chose both PoseFormer and NeuralPhysCap for their publicly available code, and to cover two aspects of monocular 3D HPEs methods. One being purely kinematic, while the second is physics-aware. Last, we include a multi-view triangulated baseline. This generates a global 3D pose by applying RANSAC triangulation from 2D wholebody MSCOCO detections [182], [183] using all available camera views. This produces a 23-joint skeleton, including heel and toe keypoints; however, we only compute MPJPE using the mutual joints.

5.3.3 Evaluation Metrics

We report MPJPE for each method, measured in mm, on our validation subset. This follows *Protocol 1*, computing the pose error relative to the pelvis (root) position. We compute MPJPE-2D, the difference between the projected ground truth and projected 3D prediction averaged across all camera views, to evaluate the 2D image alignment. These metrics capture the spatial alignment of the 3D pose but disregard the absolute 3D positioning. To account for the global 3D position error, we also report the global MPJPE-G. For measuring physical plausibility, we include Footskate (FS%) [50] and ground penetration (GP) [51], [52] in addition to our metrics. For FS%, we measure the percentage of frames where the foot moves more than 2cm while in contact with the ground. For GP, we compute the average distance to the ground for joints below the ground plane. To estimate ground contact on each sequence, we first approximate the floor height as described in Sec. 5.2.1. With the provided floor height, we follow the same heuristics as [50] employing a height threshold of 5cm and velocity threshold of 2cm/s on the foot joints to identify ground contact.

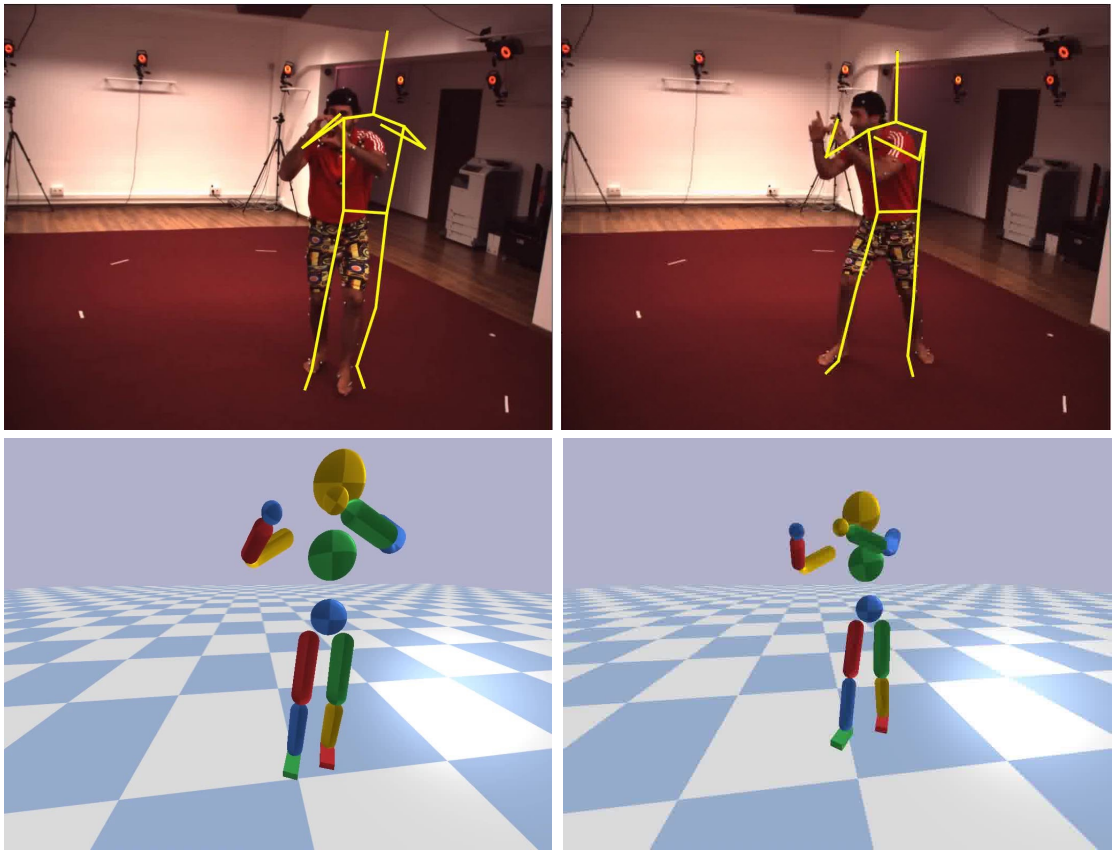


Figure 5.3: 2D re-projected (top row) and corresponding simulated body (bottom row) from NeuralPhysCap [84]. While this example displays a low MPJPE-2D=50.0mm, we measure dynamic stability to be adequate, $\text{Dyn}_{100} = 71.9$.

Method	MPJPE ↓	MPJPE-G ↓	MPJPE-2D ↓	FS (%) ↓	GP ↓	COM (Ours) ↓	Dyn_{100} (Ours) ↑
GT 3D	-	-	-	**0.0**	1.08	33.7	63.1
NeuralPhysCap [84]	81.6	439.6	36.3	29.7	2.62	29.6	62.3
PoseFormer [75]	**42.5**	299.4	**10.6**	4.4	0.30	36.2	64.8
Baseline	55.6	**57.2**	12.0	2.6	**0.26**	**27.3**	**70.6**

Table 5.2: We share results on our validation subset of the Human3.6M dataset. We compare standard 3D pose evaluation metrics as well as physical plausibility metrics.

5.4 Results

We share our results from H36M in Table 5.2. We incorporate GT 3D as an exemplar, referring to the ground truth surface markers from H36M that define the 3D pose. The first three columns focus on MPJPE-derived metrics which emphasize spatial alignment of poses, while the latter columns evaluate physical plausibility. As expected, PoseFormer [75] boasts the best (lowest) MPJPE, 42.5mm, and MPJPE-2D, 10.6 pixels. The skeleton differences with NeuralPhysCap [84] and our baseline introduce some spatial variance because the H36M skeleton is composed of markers on top of the skin, rather than body joint centers. Our baseline has the lowest MPJPE-G with 57.2mm because it utilizes multiple views and known camera projection matrices to regress a global pose. On the other hand, NeuralPhysCap has the highest amount of error because of its erroneous estimation of the camera intrinsic parameters. However, we observe little impact with our proposed plausibility metrics, discussed briefly in the next section.

For physical plausibility analysis, the baseline produces the lowest amount of foot skating, 2.6%, excluding GT 3D, and the lowest GP error, 0.26mm. This demonstrates a consistent estimation of the ground plane and aligned contact. Surprisingly, GT 3D has more ground penetration than both PoseFormer and our baseline. Although referred to as "ground truth", these exemplar keypoints are derived from a motion capture system with inherent errors. So it is likely that variance of the foot heights are within its margin of error. From these results, we expect the COM and Dyn_{100} of our baseline to be most physically plausible and NeuralPhysCap to be much worse than the others. While our baseline has the lowest COM distance, 27.3, and the highest dynamic stability, 70.6 frames, we observe that NeuralPhysCap is not too far behind with 29.6mm and 62.3 frames, respectively. Surprisingly, NeuralPhysCap is mostly on-par with the other methods in terms of plausibility within the simulator. The low COM indicates that the predicted pose is likely more stable, requiring the simulated body to deviate less from the kinematic reference.

To understand the dynamic stability, we examine the per-class performance in Table 5.3, here we underline classes with the largest amount of displacement. We note that the lowest performing classes are *Greeting, Purchases, Waiting, and WalkDog*. These sequences contain crouching and bending over movements and most frequently cause a loss of balance on the simulated body. While

Method	Dir.	Disc.	Greet	Photo	Pose	Purch.	Wait	WalkD.	WalkT.	Walk	Avg.
GT 3D	**91.6**	**83.8**	10.5	**94.5**	**82.7**	37.3	37.6	36.9	**80.0**	76.1	63.1
NeuralPhysCap [84]	86.5	82.3	73.1	89.6	62.7	35.8	**70.4**	52.1	26.2	44.3	62.3
PoseFormer [75]	69.8	77.8	**81.0**	82.8	80.5	13.3	42.2	**55.8**	66.1	**78.3**	64.8
Baseline	88.2	82.4	66.0	87.6	72.9	**72.4**	53.5	27.7	77.8	77.2	**70.6**

Table 5.3: We break down the per-class performance for the dynamic stability (Dyn_{100}) across methods. We underlined the classes with the most spatial displacement in the world plane.

NeuralPhysCap appears on par with other methods, its dynamic stability is worse on motions requiring substantial motion displacement, likely due to its errors in camera estimation. However, it still displays much better plausibility on the static actions and classes containing the bending over and crouching motions.

In Figure 5.4, we provide a qualitative example on *S11 - WalkTogether 1*. Our baseline and PoseFormer show comparable performance on physical plausibility metrics and produce similar results from the simulator output. NeuralPhysCap, on the other hand, fails early on due to poor camera parameter estimation and the large displacement required, generating physically implausible motion. The red arrow shows where the 3D pose attempts to penetrate the ground plane and on the right column, we see the clear misalignment in the image plane.

2D Image Alignment We do note; however, that spatially misaligned poses can still generate physically plausible poses. In Figure 5.3, we provide an example of NeuralPhysCap on the *S9 - Photo 1* sequence. We show low spatial alignment scores: MPJPE, 110.4mm, and MPJPE-2D, 50.0mm, but adequate physical plausibility according to our metrics COM, 0.381, and Dyn_{100},71.9. Conversely, FS, 32%, and GP, 0.07mm, errors are still high. NeuralPhysCap explicitly corrects poses for physical implausibilities, so for relatively little motion it makes sense that the output pose is plausible, in spite of poor spatial alignment. This provides evidence that we are assessing the plausibility of the pose itself and not strictly pose alignment, which can be improved with proper camera parameter estimation.

Contact Force Estimation In Figure 5.5, we demonstrate the ability to analyze contact forces from the simulated body. We generate the output contact forces from *S11 - WalkTogether 1* sequence, shown in Figure 5.4. The top row shows the estimated ground contacts from GT 3D, while the remaining rows are the measured contact forces. We see that our baseline and PoseFormer both agree in their estimated forces and with the ground contacts, while we can observe the point at which NeuralPhysCap falls over.

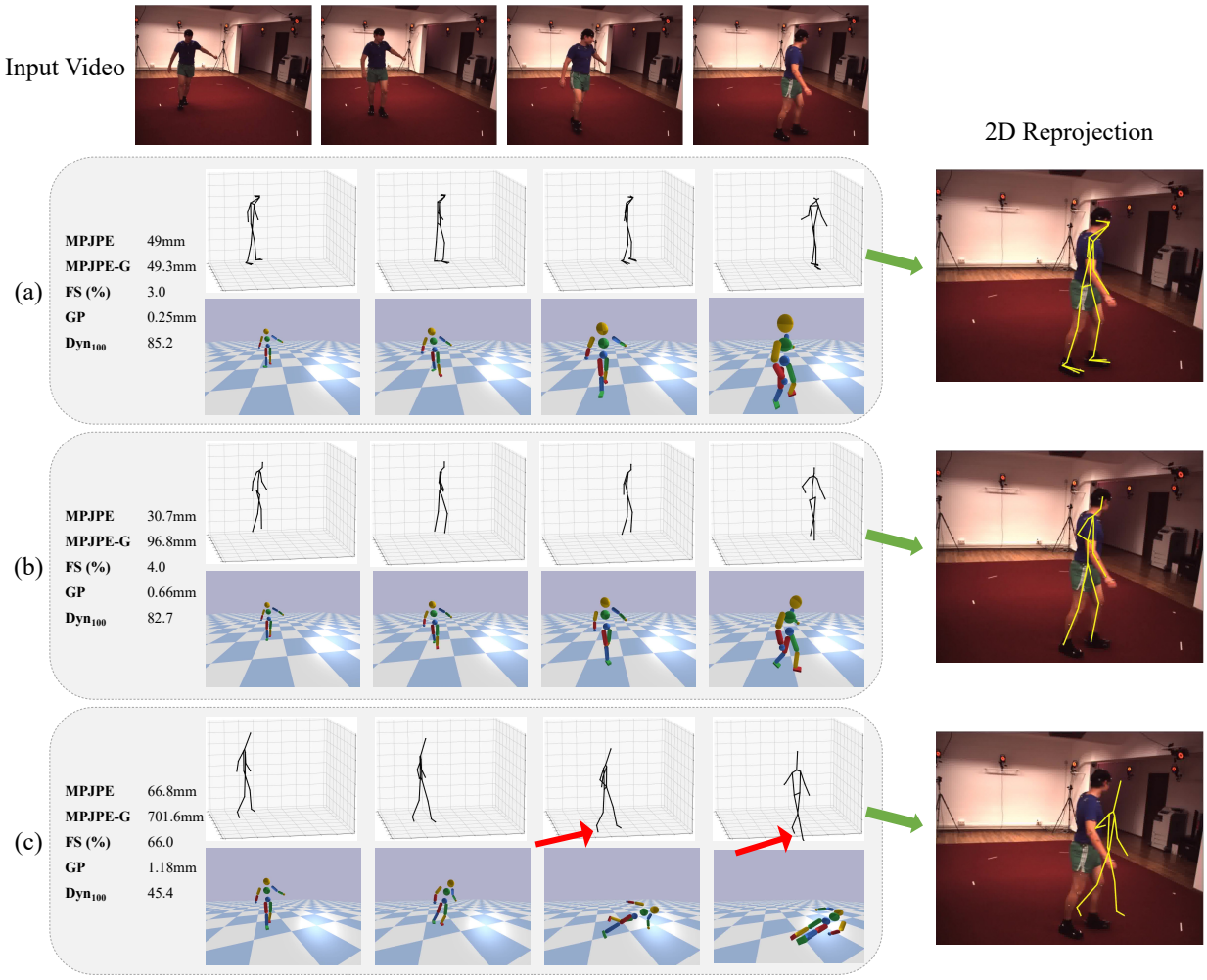
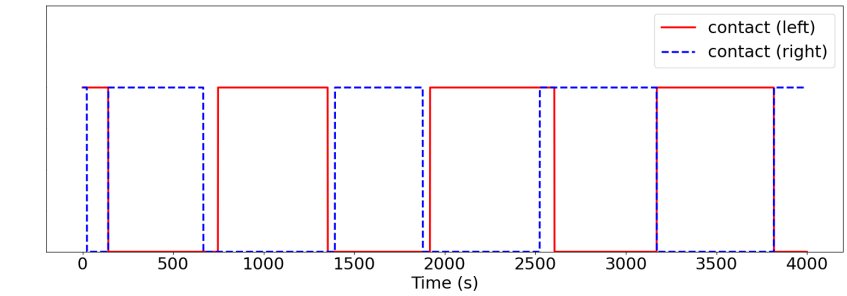
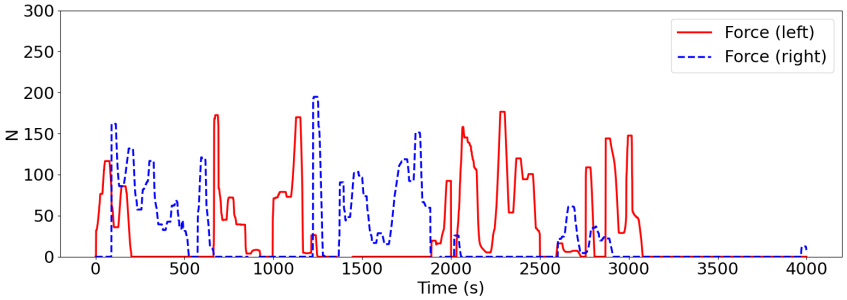


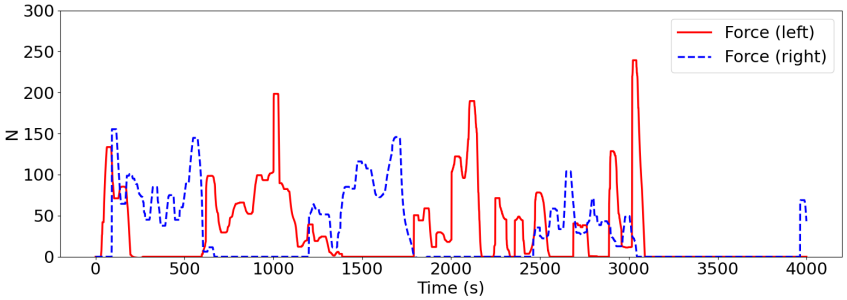
Figure 5.4: For the *S11 - WalkTogether* sequence, we show 3D prediction and simulated output results between (a) Baseline (b) PoseFormer [75] (c) NeuralPhysCap [84]. Inaccurate camera and ground plane assumptions (shown with the red arrows) in (c) causes the motion to fail early on as the simulated body tries to step through the ground plane. The right column shows the 2D re-projected predictions on the final frame.



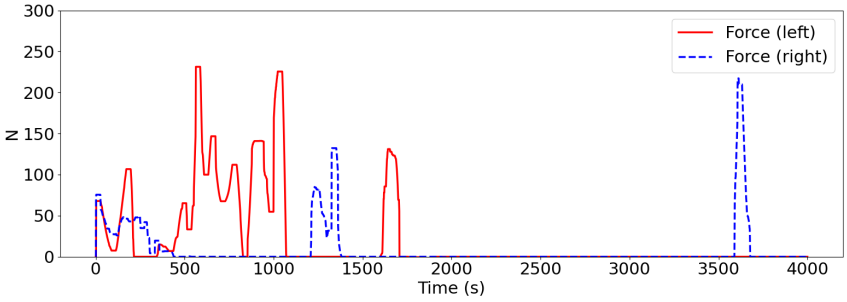
(a)



(b)



(c)



(d)

Figure 5.5: External contact force results on *S11 - WalkTogether 1*. The row, (a), represents the estimated contact using GT 3D keypoints. The remaining rows are results from (b) Baseline (c) PoseFormer [75] (d) NeuralPhysCap [84]

Method	FS (%)	GP ↓	COM ↓	Dyn_{100} ↑
Baseline-17	1.6	**0.11**	28.9	**74.7**

Table 5.4: We show results on our validation subset on Human3.6M dataset. Baseline-17 removes the toe and heel keypoints. The results here are comparable to Table 5.2.

Method	Dir.	Disc.	Greet	Photo	Pose	Purch.	Wait	WalkD.	WalkT.	Walk	Avg.
Baseline-17	84.3	75.7	77.2	**98.5**	75.2	82.4	37.8	55.4	**83.3**	77.0	**74.7**

Table 5.5: Here we show the per-class performance for the Dyn_{100} metric on Baseline-17. Baseline-17 removes the toe and heel keypoints. The results here are comparable to Table 5.3.

5.5 Impact of Toe and Heel Joints

Our main results in Table 5.2 may suggest that the baseline outperforms other methods because of additional toe and heel joints. To identify the impact of these joints, we run our baseline again with only 17 joints, removing the toes and heels from the kinematic initialization. We include a full table for the existent of joints in Table 5.6. We show results in Tables 5.4 and 5.5. In Table 5.4, we observe comparable plausibility metrics, but with noticeable improvements for GP and Dyn_{100}. These gains instead suggest that while the toes and heels provide more information about the orientation of the foot, it can also introduce additional variance into the pose estimation. Clearly, the observed physical plausibility would improve when omitting the predicted toe and heel positions. In Table 5.5, we note similar per class performance to the baseline counterpart, with the most increases coming from the lower-performing classes that contain significant crouching or bending over movements.

When the toes and heel keypoints are not detected, the orientation of the foot for the kinematic initialization is unknown. Instead, we initialize the foot joint angles with a neutral pose but impose no constraints during the optimization process. Therefore, the foot joint angles act as free variables throughout.

5.6 Additional Qualitative Examples

In Figures 5.6 and 5.7, we show qualitative examples on two of the lower performing classes, *Purchases* and *WalkDog*. We run both examples on the PoseFormer architecture. While both results show reasonable MPJPE scores, the *Purchases* sequence has much more ground penetration throughout. This suggests an inconsistent estimation of the floor which leads to some instability. While we have observed that this is generally fine with a stationary pose, this causes the simulated

body to lose balance and fall forward when combined with bending over. The *WalkDog* sequence is more stable, but struggles when the simulated body does a turnaround, possibly due to unknown foot orientation and suboptimal optimization. In Figures 5.8 and 5.9, we show qualitative examples on two higher performing class, *Walking* and *Waiting*. Coincidentally, both of these sequences appear identical. We run both of these examples on our multi-view baseline, where we note much lower MPJPE-G scores and no ground penetration. The consistent ground estimation, accurate multi-view estimation of the limbs, and the linear movement result in much more stable motion of the simulated body.

5.7 Discussion

In this work, we propose a new approach to measure the physical plausibility of 3D HPE using physical simulation. While prior approaches capture independent instances of physical implausibilities, they do not demonstrate the ability to understand the progression of these instabilities. Mainly, how do earlier errors in a pose affect latter poses? To address this, we introduce two simulation-based metrics, COM trajectory distance and dynamic stability, to understand how well a pose can be simulated and at which point it fails catastrophically. We evaluate our metrics on Human3.6m and show agreement with previous plausibility measures in most instances, but also demonstrate some invariance to spatial alignment of 3D poses.

Limitations Our work is limited by the accuracy of the trajectory optimization. If the optimization falls into a local minimum, it may deviate from the reference motion. While we mitigate this through our constraints, future work can consider alternative optimization algorithms or emergent differentiable simulators [86], [156]. In our experiments, we retarget all humans to the same simulated body. This can potentially introduce modeling errors when the shape and size vary drastically between subjects. Future work may resolve this by modifying the limbs of the simulated body to reflect the approximate shape and size attributes of the detected humans.

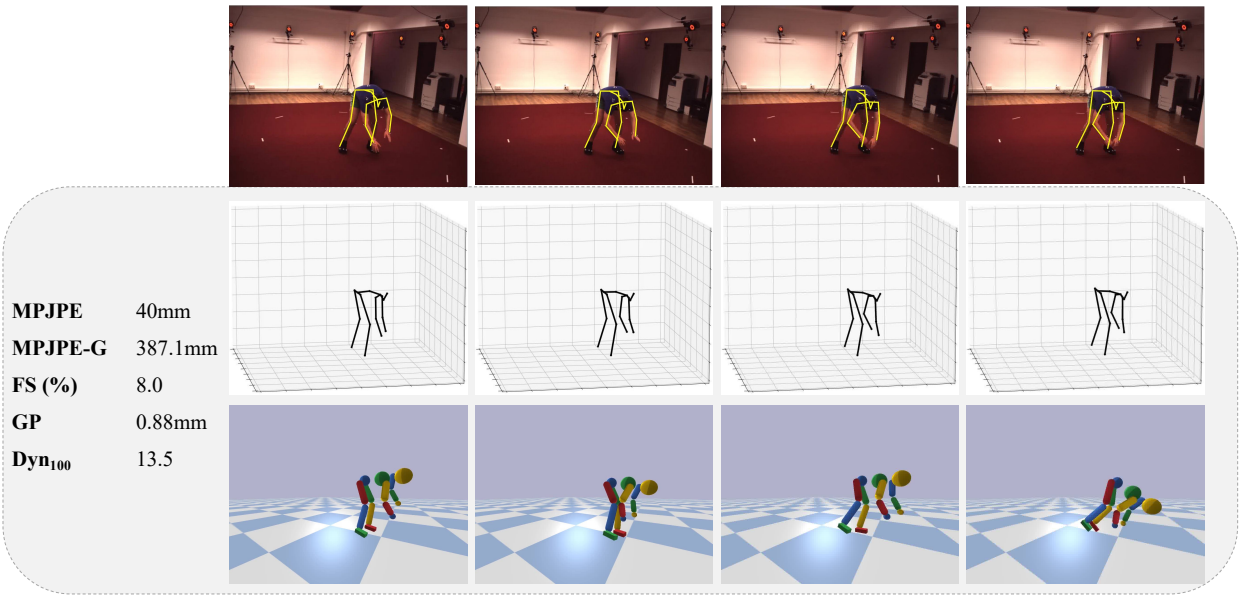


Figure 5.6: We show results on PoseFormer[75] for the *S11 - Purchases 1* sequence.

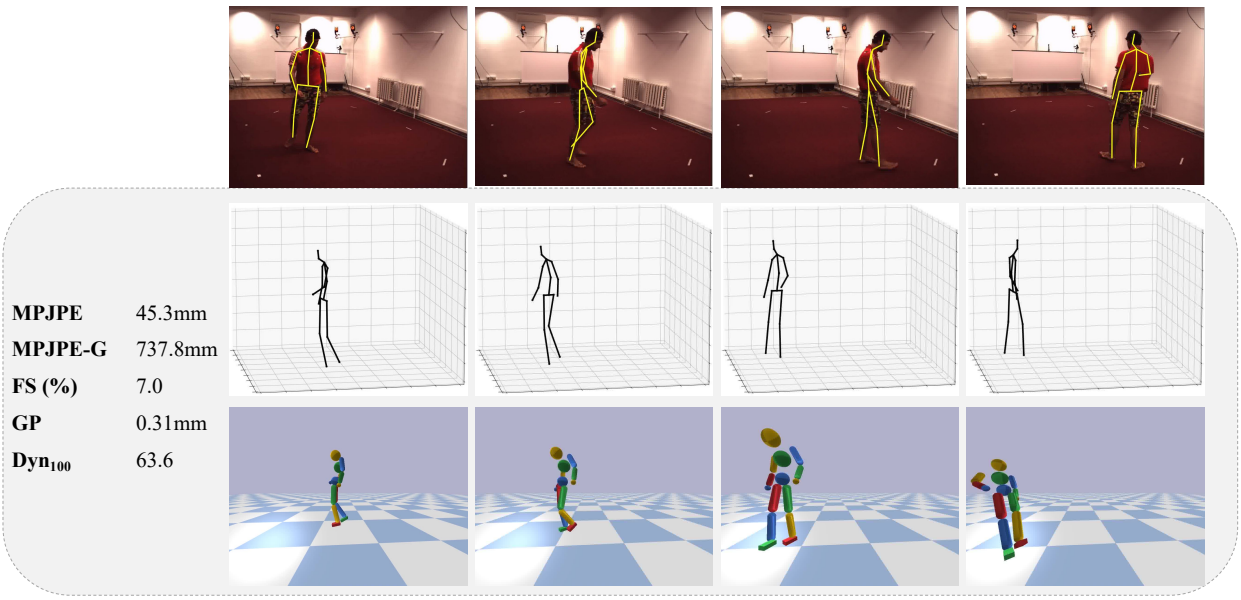


Figure 5.7: We show results on PoseFormer for the *S9 - WalkDog 1* sequence.

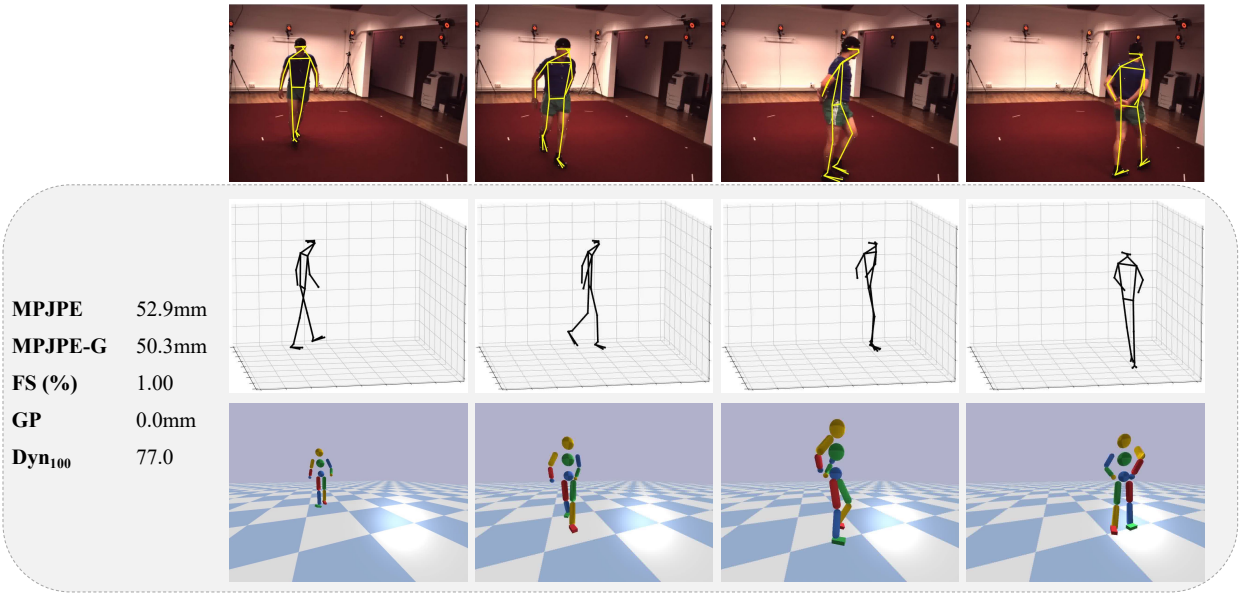


Figure 5.8: We show results on our baseline for the *S11* - *Walking 1* sequence.

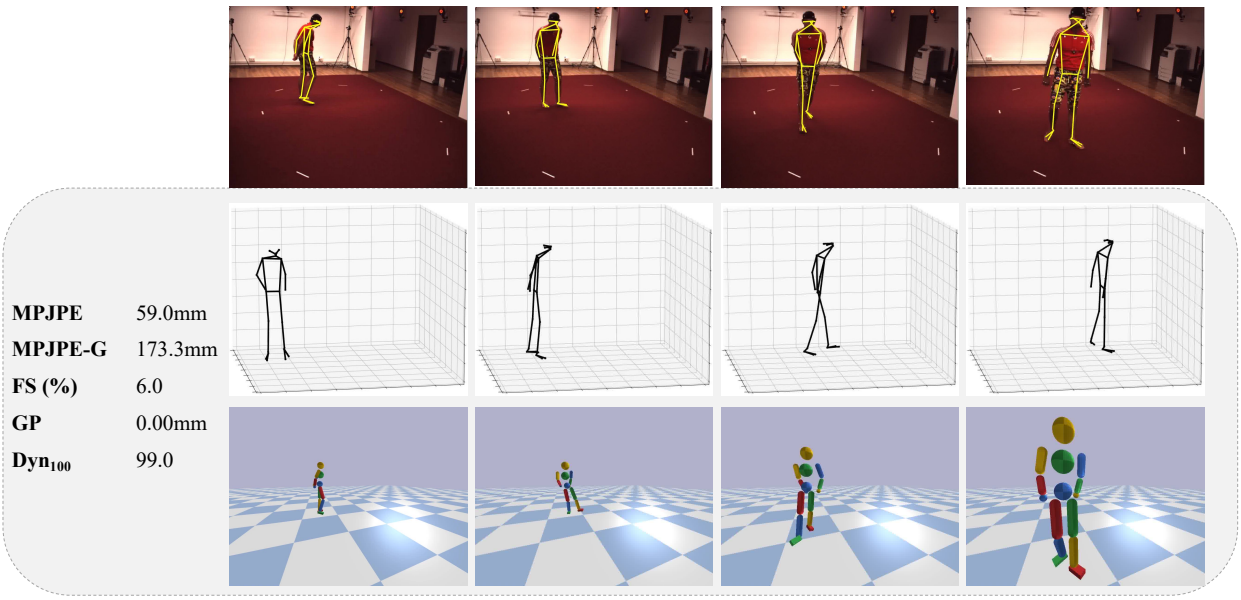


Figure 5.9: We show results on our baseline for the *S9* - *Waiting 1* sequence.

Joint Name	Baseline	PoseFormer	NeuralPhysCap
Head (top)	✗	✓	✓
Nose	✓	✓	✗
L Eye	✓	✗	✗
L Ear	✓	✗	✗
R Eye	✓	✗	✗
R Ear	✓	✗	✗
Neck	✗	✓	✓
Pelvis	✗	✓	✗
Thorax	✗	✓	✗
L Shoulder	✓	✓	✓
L Elbow	✓	✓	✓
L Wrist	✓	✓	✓
R Shoulder	✓	✓	✓
R Elbow	✓	✓	✓
R Wrist	✓	✓	✓
L Hip	✓	✓	✓
L Knee	✓	✓	✓
L Ankle	✓	✓	✓
R Hip	✓	✓	✓
R Knee	✓	✓	✓
R Ankle	✓	✓	✓
L Big Toe	✓	✗	✓
L Small Toe	✓	✗	✗
L Heel	✓	✗	✗
R Big Toe	✓	✗	✓
R Small Toe	✓	✗	✗
R Heel	✓	✗	✗

Table 5.6: The detected joint format used for each method, GT 3D and PoseFormer use the same exact joints. If the pelvis joint is not detected, it is estimated from the left and right hip joints.

CHAPTER 6

Contrastive Learning for Video-Based Skill Assessment in Open Cardiac Surgery

6.1 Introduction

Measuring the performance of human actions is often subjective, but standardized rubrics like the ISU Judging System [5], for Olympic-level skating, or OSATS [39], for surgical skill, aim to improve objectivity. Despite these efforts, human assessors may still introduce bias [4], [5] and scalability issues. Naturally, automated methods for assessing human actions have emerged across domains such as sports [6], [119], [184], surgery [9], [10], [13], or generic technical skill tasks [14], [15].

However, these videos are more difficult to obtain in surgical domains where medical expertise and regulatory processes are required. Limited existing work on open surgery videos computes kinematic features on 5–30 second videos from manually initialized regions [40] or performs dimensionality reduction analysis on top-down kinematic hand attributes [42]. Consequently, present research in surgical skill assessment is primarily focused on readily obtainable videos of bench-top models [13], [21]–[23] or robotic surgery [9], [10], [12], [24].

In contrast, in this work we study skill classification—specifically Coronary Artery Bypass Grafting—from open surgical videos in live operating room settings. These settings present inherently challenging, unstructured complexities along numerous axes. First, the camera placement must not interfere with the operation, therefore heavy occlusion by heads and torsos of the surgical team members frequently occurs. Second, the presence of multiple surgical team members entangles detected objects and actions, adding distractors while increasing difficulty of individualized assessment. Last, each step of the surgical procedure may vary significantly in appearance and duration within the same operating room. Furthermore, the available data resources for open surgery are significantly smaller than necessary for contemporary approaches in deep learning.

Hence, we propose clip-level contrastive learning on hand features to maximize the utility of the available open surgery data. Our approach has two key elements. First, we propose a time-shift augmentation to ensure that the temporal dynamics of the videos are maintained while still being

able to leverage unsupervised contrastive learning to pretrain a suitable embedding representation; these dynamics would be lost if we used standard image augmentation techniques for contrastive learning [129]. This pretraining occurs on randomly sampled clips from our video corpus with the loss driven to bring embeddings for clips from the same video (and hence skill level) closer together than embeddings for clips from different videos (depicted in Fig. 6.1). Following the pretraining we append a classifier atop the network and use supervised training.

The second key element is the actual feature encoding we use on these clips. Although raw pixel features are plausible (and we compare against them), we instead use hands as proxies for surgical skill [22], [40], guided by surgical experts. For a given clip, we generate instance-level features using fine-tuned hand bounding box and pose detection networks [148], [185].

We evaluate our method on the skill classification task, classifying novice or expert surgeons from video. We share results on the published simulation dataset, VTS [21] and our newly collected dataset, Cardiac Open Surgery Skills Assessment (**COSSA**), a novel dataset consisting of 61 videos from 26 real surgical operations. We compare our approach to supervised training of an identical multi-layer network, a large pretrained video transformer [149], and a linear Support Vector Machine Classifier (SVM-C). Our experiments show that clip-level contrastive pretraining provides a consistent improvement in performance for the downstream skill classification task.

6.2 Surgical Skill Assessment with Contrastive Learning

Given a video of an open cardiac surgery, our goal is to assess the skill of the surgeon captured in the video. Here, we aim to classify the skill level into one of two buckets—novice or expert—noting the numerous potential future enhancements to this initial classification [39].

Concretely, assume we have a collection of videos $\mathcal{V}$ such that each video $\mathbf{v}_i \in \mathcal{V}$ captures a relevant segment of the surgical procedure and there is an associated skill classification label $s_i \in \mathcal{S} = \{\text{novice}, \text{expert}\}$. As common in surgery, procedures typically have one surgeon leading a segment of the operation with a second surgeon assisting. We assume that each such video of the procedure subtends one skill classification label, assigned to the surgeon leading that segment of the procedure. Ultimately, our task is to learn a classifier g that will map an input video clip to a classification; however, for clarity, let us assume that we define a feature-mapping f from the raw clip to an embedding space suitable for skill assessment:

$$\hat{s} = \arg\max_{\mathcal{S}} g(f(\mathbf{v})) \ . \tag{6.1}$$

In the following subsections, we expand on how we implement these two pieces. Importantly, obvious feature embeddings, including the identity (using the raw video) as well as obvious ones

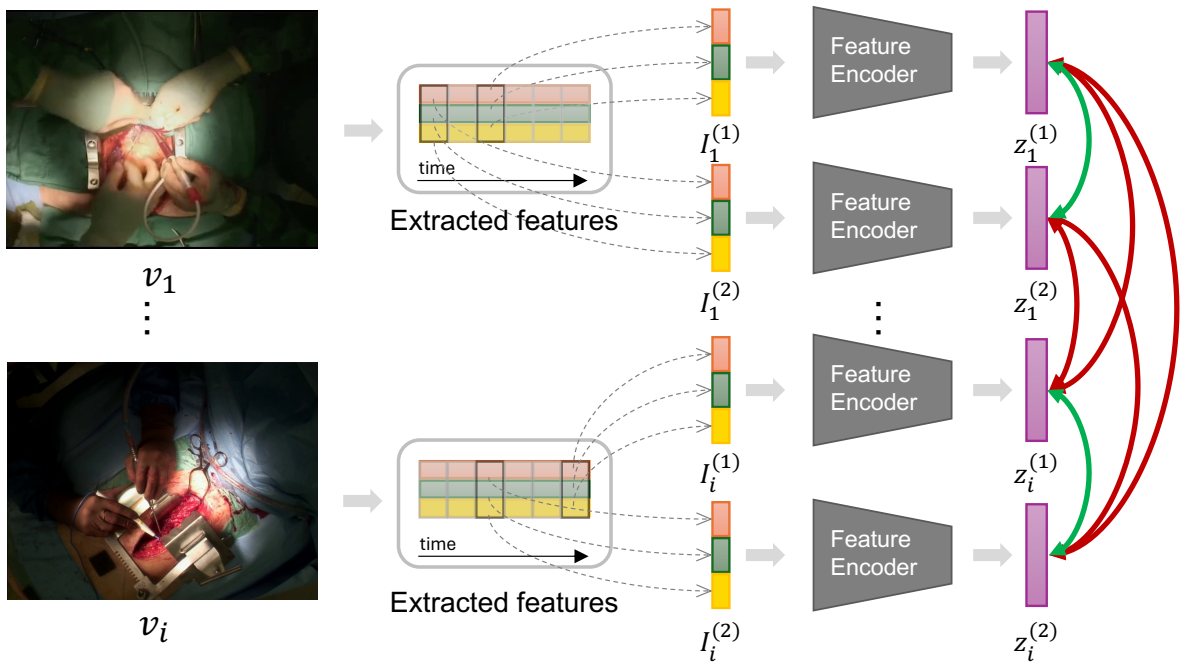


Figure 6.1: For unsupervised contrastive pretraining, we temporally sample clip-level features, I_i, from each video, which could be space-time patches, learned features or those tuned to the domain. We further embed them into a latent representation, z_i, which is obtained through a constrastive loss-based pretraining. Green lines indicate positive pairs (collected from the same video v_i) and red lines indicate negative pairs (collected from different videos). After pretraining, we append the final layer of the contrastively-trained feature encoder and fine-tune for the actual skill classification task.

like extracted motion inspired by domain expertise [42], are unable to capture the variability of the open surgery context in the face of relatively limited data, which our experiments show. We hence adopt contrastive learning to encourage the feature embedding f to learn a representation capable of distinguishing novice from expert surgeons.

6.2.1 Clip-level Contrastive Learning

We propose a two part structure to learn the surgical skill assessment model. In part 1, we pretrain the model by using unsupervised contrastive learning. Then in part 2, we fine-tune this model with supervised labels for the actual skill classification.

Open surgery videos are complicated by diverse camera placement, self-occlusions, multiple surgeons, and high content variability from the specific patient and procedure. Yet, the volume of available data to leverage is comparatively small. Hence augmentation is necessary; however, we cannot adopt standard image-level augmentations because of the risk that the dynamics of the procedure—the very thing we expect to be important in surgical skill assessment—would be corrupted.

Part 1: Unsupervised Contrastive Pretraining with Time-Shift Augmentation

We propose a time-shift augmentation to support learning a useful feature embedding through contrastive learning [129]. For each video v_i, we randomly sample N non-overlapping clips $c_i^{(1)}, \ldots, c_i^{(N)}$ and compute their corresponding features $I_i^{(1)}, \ldots, I_i^{(N)}$ from the extracted features (see §6.2.2). As depicted in Fig. 6.1, we induce labels for the contrastive learning, observing that, for, say, clips $c_i^{(\cdot)}$ and $c_j^{(\cdot)}$, intra-videos clips should be treated as positive pairs (*i.e.* when $i == j$) and inter-video clips should be treated as negative pairs (*i.e.* when $i \neq j$). We then drive pretraining using an InfoNCE-based loss [186]:

$$l_{m,n} = -\text{sim}(\mathbf{z}_m, \mathbf{z}_n)/\tau + \log \left[\sum_{k=1}^{B} \mathbb{1}_{[s(k) \neq s(m)]} \exp(\text{sim}(\mathbf{z}_m, \mathbf{z}_k)/\tau) \right] \quad , \tag{6.2}$$

where B is the batch size, m and n index into any two clips, $s(\cdot)$ maps to the source video index for any clip, $\mathbf{z}_m$ is the contrastive embedding for clip m that we are learning, sim is the standard cosine similarity metric between two vectors, and τ is the temperature scalar. This loss drives the representation to encourage clips from the same video to have a high similarity and, at the same time, to be distant from clips from other videos (in the batch, at least).

Part 2: Fine-Tuning for Skill Classification

After completing the contrastive pretraining, we append a linear layer to the feature encoder to serve as our classifier g and fine-tune for skill classification. For clip-level predictions, we reformulate Eq. (6.1) to operate on a clip c directly:

$$\hat{s} = \arg\max_{\mathcal{S}} g(f(c)), \tag{6.3}$$

with f serving as our contrastively-trained encoder on features extracted from clip c. Following this, we generate a consensus from the clip predictions. For clip predictions $\hat{s}_q^{(1)}, \ldots, \hat{s}_q^{(N)}$ for query video $\mathbf{v}_q$, we compute an empirical density P and choose its mode:

$$\hat{s} = \arg\max_{\hat{s}_k \in \mathcal{S}} P(\mathbf{Z} = \hat{s}_k) \quad . \tag{6.4}$$

6.2.2 Feature Extraction

For determining skill classification in surgical settings, we focus on the hands of surgeons. Hence, we first produce tracked hand bounding box and hand poses followed by computed summary statistics from those features. A hand detector first generates bounding boxes for all hands in the scene, followed by confidence thresholding($= 0.8$) and a non-maximal suppression (NMS) with an

IoU threshold($= 0.25$). Tracking is performed using a linear sum assignment algorithm [187] on the results, with costs computed from IoU overlap and 0.5 seconds of prior frame history. We estimate hand joint poses from the final hand tracks. Afterwards, we compute summary statistics from the T longest hand tracks from each video which serves as our feature representations. This assumes that the longest appearing hands will be the objects of interest, allowing for tracking failures such as out-of-frame motion or severe occlusion. We explore this parameter in Supplementary Material, but results in Section 6.3.3 use the best T value. Summary statistics, instead of sequential frame-wise features, allows us to isolate salient parts of the extracted data, while retaining low-dimensionality for a smaller network. For hand boxes, we collect the total path length, average and max velocity. We include the average and standard deviation of the area, which can approximate z-axis distance from the camera. For hand poses, we use the total path length and wrist-relative changes of the joint positions.

6.3 Experimental Setup and Results

6.3.1 Data

COSSA—Cardiac Open Surgery Skills Assessment To evaluate our contrastive learning method for skill in open surgery, We collect 61 video segments of cardiac surgical procedural steps from 26 open surgical cases. This data consists of 19 subjects (thirteen resident, six attending). The surgical cases range between $3 - 7$ hours but each video segment represents a key procedural step meant to exemplify proficiency of technical skill. The key procedural steps we study are: preparation and initiation of cardiopulmonary bypass (CPB), pre-bypass transesophageal echocardiography (TEE) assessment, harvest of the left internal mammary artery (LIMA), performance of distal arterial and vein anastomosis, and performance of proximal arterial and vein graft anastomosis. A senior cardiac surgery resident reviews the cases, temporally segments the clip containing the step, and labels the segment as performed by a resident (novice) or attending (expert). The key action is performed by one while the other is assisting. From the 61 video segments, 21 are labeled attending and 40 labeled resident surgeon. The videos have a mean length of 15 minutes, but we limit evaluation to the first 10 minutes. For the skill classification task, we use 3-fold cross-validation, grouping the videos by case.

Variable Tissue Simulator We also use the Variable Tissue Simulator (VTS) [21]. It is a dataset of two simulated suturing tasks: tissue paper material, to resemble friable tissue, and rubber balloon, to imitate arteries. The dataset has 25 subjects (eleven medical, one resident, thirteen attending surgeons) completing two trials of each task for a total of 100 videos. Each video is between 2–6 minutes long. We group the resident and attending surgeons as experts and the medical students as novices. We use a similar 5-fold cross-validation set as Bkheet *et al*. [41], grouping the videos by

user.

6.3.2 Implementation Details

We use DeTr [148] fine-tuned on hand bounding boxes from SurgicalHands [185] and 100DOH [164] for the hand detection feature extractor. We replace the final layer to output two classes, left and right hand, and fine-tune with a learning rate of $1e^{-5}$ for 100 epochs. For the hand pose estimation we use a ResNet trained on the poses from Louis *et al*. [185], each pose contains 21 joints. We encode all inputs using a 3-layer fully-connected network with ReLU non-linearity and a hidden dimension of 128. We run the contrastive unsupervised training step for 500 epochs followed by 100 epochs of fine-tuning on the target task. We use a batch size of 128 for training and 64 for validation. For the learning rate, we run a hyper-parameter search between $1e^{-2}$ to $1e^{-5}$ to discover the optimal value for each input. We employ a weighted random sampler to reduce class imbalances within batches. We refer to this concrete instantiation of our proposed method as **NN-Contrastive**. For comparison to other prediction models, we include a supervised training counterpart, an identical 3-layer network without the contrastive pretraining, which we call **NN-Supervised**. We include a video activity recognition transformer, TimeSformer [149], to operate on raw RGB frames. We fine-tune this with a learning rate of $1e^{-6}$ for 100 epochs. And last, a linear Support Vector Machine (SVM) classifier.

6.3.3 Surgical Skill Classification Results

For evaluating skill classification, we use classification accuracy and the multi-class definitions of Precision and Recall, the average of both classes. We discovered a video length bias on the VTS dataset, namely novice videos are noticeably longer than expert videos. Hence, video length as input achieves an accuracy of 84% (Pr=0.86, Rc=0.85) on an SVM classifier, but completing a task faster does not equate to higher skill. To mitigate this bias, we use 60-second clips as inputs.

We share our results on VTS in the first set of results in Table 6.1, using mean k-fold cross-validation. In the first row, we use the most frequent label in the training fold as a baseline. This achieves an accuracy of 58.0%, about the exact ratio of experts to novices, and a recall of 0.5, indicating only a single class selection. The TimeSformer model reports an accuracy of 77.8%, however, hand box and pose features, separately, performs better than RGB features. The highest accuracy comes from the hand box features, 85.5% for our NN-Contrastive model and 80.8% for the NN-Supervised model. We suspect the higher performance from the RGB frames and NN-supervised is due to the simplicity of the VTS dataset. Only single pair of hands are present and the well-lit, unchanging environment makes it easier to isolate relevant features or attend to subtle visual biases between trials.

Table 6.1: Skill classification results on VTS (left) and COSSA(right) datasets with comparison to baselines. Averaged across 5-folds.

		VTS			COSSA		
Inputs	Model	Acc	Prec	Rec	Acc	Prec	Rec
One label	Baseline	58.0%	0.29	0.50	60.9%	0.30	0.50
RGB frames	TimeSformer [149]	77.8%	0.78	0.77	48.4%	0.34	0.46
Hand Box	SVM-C	77.5%	0.83	0.83	64.1%	0.62	0.60
	NN-Supervised	80.8%	0.84	0.84	66.0%	0.74	0.57
	NN-Contrastive (Ours)	**85.5%**	0.86	0.86	**70.7%**	0.69	0.65
Hand Pose	SVM-C	57.8%	0.57	0.57	-		
	NN-Supervised	71.3%	0.79	0.79	64.1%	0.48	0.54
	NN-Contrastive (Ours)	**79.8%**	0.83	0.83	**69.1%**	0.73	0.61
Hand Box + Pose	SVM-C	63.8%	0.63	0.63	-		
	NN-Supervised	76.0%	0.80	0.80	66.4%	0.69	0.60
	NN-Contrastive (Ours)	**76.5%**	0.80	0.80	**68.1%**	0.71	0.66

This is apparent in Table 6.1, showing results on COSSA on the right, with increasingly complex scenes. Here, we run the SVM classifier only on the hand box metrics. In contrast to the VTS dataset, the transformer appears to struggle with classifying skill based on the video features. This maybe because the VTS data is easier to over-fit to scene-specific biases rather than action characteristics. The contrastive training makes the biggest difference across all three input modalities. But again, we note a higher performance with the hand box and hand pose features separately than together.

6.3.3.1 Analysis of Motion Characteristics

In Figure 6.2, we examine extracted features on correctly classified video clips comparing novices and experts between the VTS dataset [21] and COSSA. In both environments, we note the motion of the experts are localized within small regions, while the path of the novices tend to traverse a wider area and at a higher velocity as shown in Table 6.2. COSSA contains more complex scenes which entangles both the attending and resident surgeons, that can only be manually disentangled. However, by combining all detected metrics and analysing the scene holistically, we can still find separation between the expertise levels.

6.4 Discussion

In this work we study surgical skill assessment for open surgery cardiac procedures. We introduce a new clip-level contrastive learning method displaying superior performance particularly

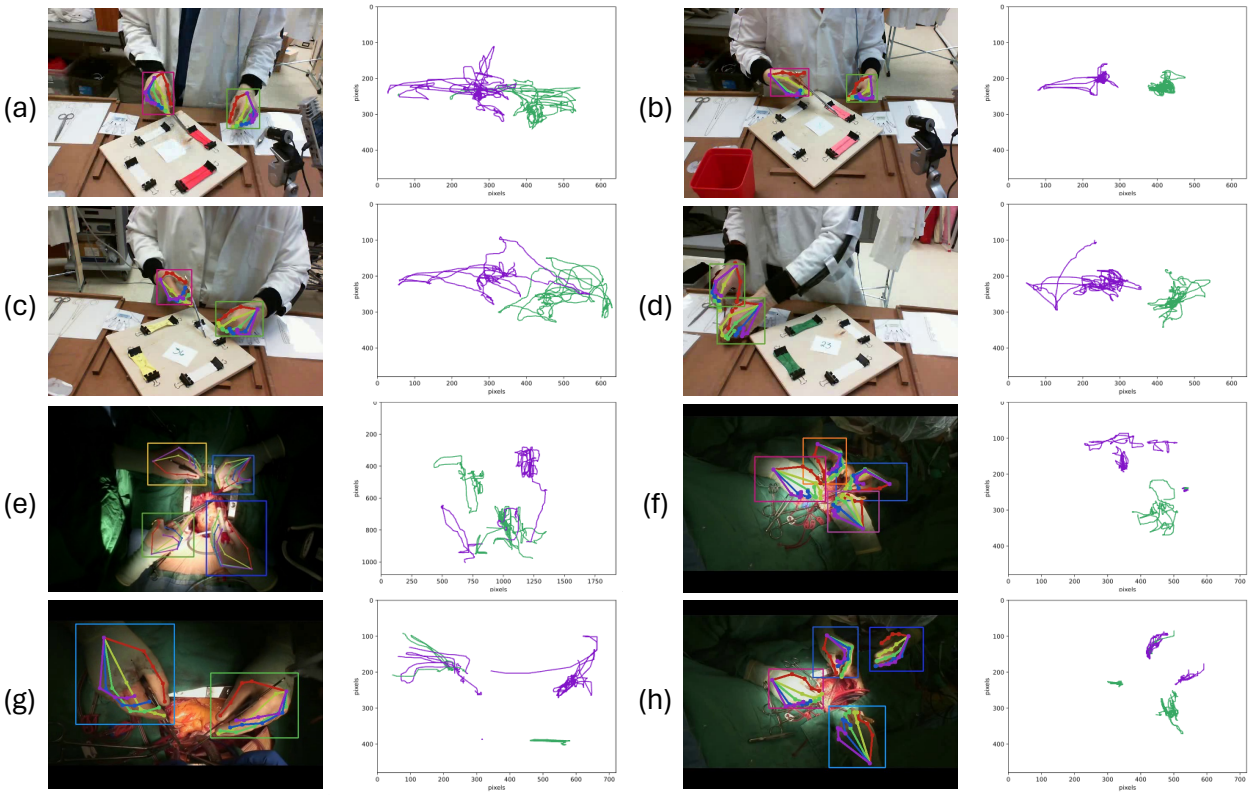


Figure 6.2: We show correctly classified results on VTS [21] (a)-(d) and COSSA (e)-(h), using $T = 10$ object tracks. Each examples shows a detected frame and corresponding hand tracks. We note that novice motions traverse a wider area while expert motions are more localized and precise. Purple tracks denote detected right hands and green tracks are detected left hands.

Table 6.2: Corresponding summary statistics computed from samples in Figure 6.2. Measures below are shown as Left Hand(s) / Right Hand(s).

Data	Sample	Length (px)	Avg. vel. (px/sec)	Max vel (px/sec)
VTS	(a) Novice	$2.1e^3$/$2.3e^3$	0.10/0.08	0.80/1.49
	(b) Expert	$2.6e^3$/$2.5e^3$	0.05/0.05	0.46/1.11
	(c) Novice	$3.9e^3$/$3.5e^3$	0.07/0.07	1.11/0.74
	(d) Expert	$2.6e^3$/$3.7e^3$	0.05/0.07	0.46/0.88
COSSA	(e) Novice	$1.9e^3$/$1.7e^3$	0.21/0.24	2.97/2.72
	(f) Expert	430/389	0.06/0.04	0.70/0.89
	(g) Novice	547/589	0.14/0.15	2.54/1.81
	(h) Expert	530/338	0.05/0.06	1.00/0.69

on the open surgery data with respect to standard contemporary approaches, such as a video transformer. Rather than coarse frame-level features, we generate our representations from instance-level hand features to highlight characteristics of skill. And we are able to quantify certain motion characteristics that are indicative of expertise, even with multiple surgical team members. We evaluate on new a set of data, COSSA, to account for the absence of dynamic, unstructured data representing live surgical scenes. Future work can extend to other important surgical non-technical skills including decision making and team coordination.

CHAPTER 7

Conclusion and Future Directions

7.1 Conclusion

In this dissertation, we have proposed several building blocks aimed at solving the qualitative human assessment problem. Broadly, we seek to automate this process using computer vision and AI to overcome the bias and scalability issues associated with manual assessors. Specifically, in relation to current literature, our primary aims are to improve extraction of image-level information, ground physical components from the scene, and discover correlations to skill-based assessments. Chapter 3 introduces *CondPose*, a network that improves hand pose predictions by leveraging past observations as priors, resulting in enhanced localization and tracking accuracy. This approach ensures a more precise representation of hand poses within the scene. In Chapter 6, we demonstrate the efficacy of embedded hand features for surgical skill classification and assessment in open-surgery cardiac procedures. These hand embeddedings surpass coarse frame-level features and showcases the superiority of our proposed clip-level contrastive learning over its supervised counterpart. In Chapters 4 and 5, we work towards grounding physical understanding of full body poses. We propose a novel approach for predicting Ground Reaction Forces (GRF) from video in Chapter 4, by minimizing peak impact errors through the introduction of a gated-MSE loss. Additionally, we demonstrate that pre-training and multi-task learning on 2D-to-3D human pose estimation reduces error on smaller datasets and improves generalization to unseen motions. Finally, in Chapter 5 we introduce two simulation-based metrics, COM trajectory distance and dynamic stability, to gain a measure of plausibility of 3D human poses.

An essential part of our work is the acquisition of domain-specific data. To leverage our proposed methods or measure performance, we require some annotated data within the target domain or problem space. Video data was severely lacking for articulated hand pose tracking, rendering pose tracking metrics virtually impossible. To address this, we collect *SurgicalHands*, the first articulated multi-hand pose tracking dataset of its kind, notable for its annotated labels of hands in surgical settings, presenting numerous appearance and environmental complexities. Also within that domain, we evaluate surgical technical skill on *VARSITY-Surgery*, to account for the absence of

dynamic, unstructured data representing live surgical scenes. These data were obtained through Michigan Medicine and collaborating institutions. And last, we publish the *ForcePose* dataset, a large collection of multi-view tracked human motions with paired force plate data. This equips researchers with a structured benchmark for measuring contact force estimation for a series of human motions.

The work in this dissertation touches the surface for human-centric analysis. We have shown that by using limited labeled video data and semi-supervised learning techniques, we can infer quantitative evaluation characteristics of human actions. This encourages future work to employ creative learning strategies and leverage inherent biases from existing data for niche tasks.

7.2 Limitations

Failure at detection Our approach begins with detecting and extracting relevant visual components (hand and full body poses) from video, closely aligning with methods that identify predefined classes such as human poses [6], [8] or surgical instruments [10], [12]. While this may remove background distractors and noisy features from the decision framework, it can worsen performance when detection itself fails. This can occur for a myriad of reasons such as motion blur, occlusions, or even multiple objects in close proximity. In Chapter 6, we account for this by considering multiple object tracks. But if an object is missed completely, then it cannot be incorporated into decision making. To resolve this limitation, future work may consider a multi-modal transformer to apply attention across image patches [147] and regressed poses.

Extrapolation of contact forces In our contact force estimation work, our quantitative evaluation relies heavily on supervised force plate data. Evaluation on force-less video sequences can only be performed qualitatively. While we have shown improvement within our distribution of motions, we suffer from extrapolation of reasonable contact forces as shown in Figure 4.5 (b). However in Figure 4.5 (c), we show reasonable force outputs with rotation and translation data augmentations on the input pose. This has a negative impact however, increasing the RMSE error on the *ForcePose* validation set. This can be addressed by increasing the variety of the validation set to be more representative of motions or to use inherent biases of contact forces to constrain them during training.

Computational Efficiency and Accurate Simulation Modeling For physical plausibility, our simulation-based metrics are limited by the computational efficiency and accuracy of the trajectory optimization. We utilize the CMA-ES [180] algorithm for trajectory optimization, however this only operates on the CPU with no GPU acceleration. This makes this optimization very computationally

inefficient, requiring several hours for a few seconds of video. Additionally, the stability appears to deteriorate with time when the optimization falls into a local minimum and fails to recover. While we mitigate this through our constraints, future work can consider alternative GPU-ready optimization algorithms or emerging differentiable simulators [86], [156]. Additionally, within the simulator, we retarget all humans to the same simulated body. Because most videos do not contain any information on the ground truth shape of the person, we believe a rough approximation for determining generally physical plausibility is sufficient. However, for more accurate extraction of metrics or re-targeting of complex actions, modeling the shape and size of each limb attributes accurately may be required.

7.3 Future Work

Constraining force embeddings with self-supervision As implied in limitations, the output space for the contact force estimation is under-constrained for videos without force labels. We seek to overcome this by introducing a cyclic loss. In Robotics, Wang *et al*. [27] approaches this similarly with abstracted annotations to weakly supervise unpaired data and regularize the correspondences. For our work, this would involve producing a force output for all videos and reconstructing the input pose from the predicted forces, forming a cyclic loss with self-supervision. But we can strengthen the relationship between poses and GRFs by including the supervised contact loss for the labeled examples. This would result in the following objective function for examples with labeled forces

$$\mathcal{L} = \mathcal{L}_{\text{force}} + \lambda \cdot \mathcal{L}_{\text{cycle}}, \tag{7.1}$$

while videos without labeled forces use only the cyclic loss,

$$\mathcal{L} = \lambda \cdot \mathcal{L}_{\text{cycle}}, \tag{7.2}$$

with a λ hyper-parameter to control the impact of the cyclic loss. This can enable us to improve our force prediction, by injecting our training data with multi-view data without force supervision or an auxiliary task, regularizing our force prediction through cycle consistent correspondences. Additionally, we could include a softer supervision on all samples for $\mathcal{L}_{\text{force}}$ based on inherent biases in nature. For example, when in contact with the ground, the summation of gravitational forces must be greater than or equal to mass $\times$ gravity.

Informing force embeddings with physical simulation Other works have sought to use cycle consistency across different modalities such as image to physics parameters or simulation agents [27], [188]. However without strictly paired supervised examples, cycle consistency may suffer

from misalignment issues. This occurs when instances between two domains are cycle consistent, but uncorrelated within their respective domains. When reconstructing the pose, there may still be practical limitations using just GRFs. Namely a many-to-one relationship, where different reconstructed motions may share the same exact force vector. We can consider enhancing our force representation directly with physical properties such as joint angles, orientations, and velocities estimated from a rigid body simulator. Many works [51], [89], [154], [155] have used physical simulation to approximate internal dynamics (*i.e*. joint torques) needed to reproduce a reference motion. But still these estimated torques and forces are un-grounded, and it remains unexplored if they map to realistic physical measures. Future work may explore their usefulness for mapping to supervised forces, while indirectly helping our network understand force-less data as well.

Enhancing latent skill representations In Chapter 6, we showed that hand features can outperform raw video features when used for surgical skill analysis. While competitive within certain skill assessment datasets [184], raw video features often lack interpretability because the model attends to all elements of the visual space. This can lead to over-fitting to dataset-specific environment and appearance factors, rather than characteristics of the actions. However, if detection fails then the entire pipeline is compromised, as discussed in limitations. While we experimented with concatenation of hand box and hand pose metrics, these metrics were manually computed. A learning-based model integrating multiple modes of data may offer the advantages of low-level pixel-based features and higher-level hand poses regressions. Future work should explore a multi-modal transformer for its capability to attend to all parts of lengthy sequences, although addressing the drawback of data size is necessary beforehand.

APPENDIX A

Articulated Hand Pose Tracking

A.1 Inference on a video

During training, all priors are from the dataset are known at time $t - \delta$. But, as mentioned in Sec. 3.3.2, we do not have the correspondence between each prior and current image crop during validation or testing. So we filter out improbable priors using two steps:

1. Reprojection of predictions back onto the full frame

2. Thresholding the average peak Gaussian across joint predictions

When we reproject the spatial position of the prediction back onto the full frame, the relevant cropped region gives us an easy of way deducing whether this is for the same object. For extremely large displacement, the prior will have values of all zero and hence can be tossed as a potential prior for the current object. We also use a minimum Gaussian peak threshold hyperparameter, where we average the peak prediction values for each joint. This removes the lower confident predictions that may not serve to improve the current prediction. In practice, we found this value to be between 0.20-0.25 for our best performing models, on a scale from 0 to 1.0. This means, to maximize performance, the average prior peaks must be at least that high.

We show how our model functions during inference in Fig. A.1. Time step t has only one prior and detected hand, so this pair is used to perform a prediction at time t. At time $t + 1$, a new object is introduced hence its prior is just a heatmap of zero values. And at time $t + 2$, we use each appropriate prior with its image crop reducing the number of priors by referencing our filtering steps again.

A.2 Model Architecture and Training Details

We show some additional details of each branch in our model shown in Table A.1. Each layer details: input features, output features, kernel size, and stride. All other parameters are set to default. For our hand dataset, the number of joints is 21.

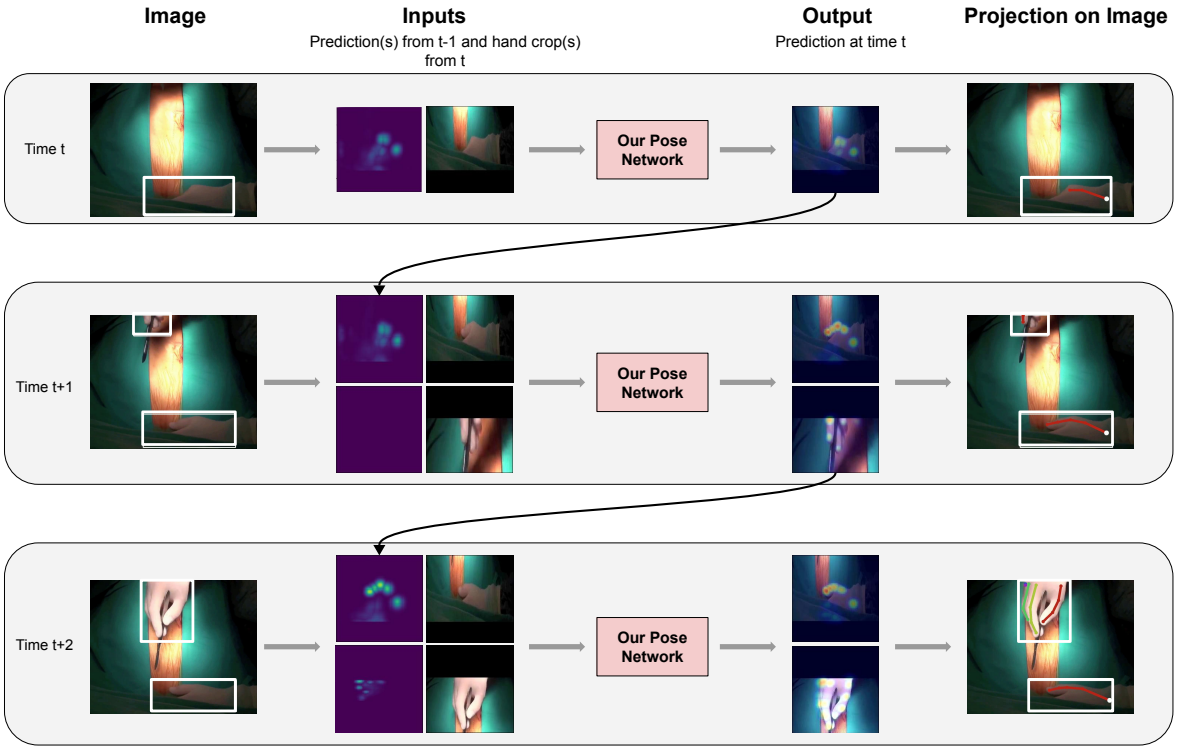


Figure A.1: A detailed overview of our model during evaluation on videos. Like training, newly introduced objects are accompanied with a blank, zero heatmap prior. We use two post-processing steps to filter out improbable priors when assigning matches. Each image crop is padded and centered on the detected hand.

Branch	Layer details
Atten. Mechanism	Conv2d(64 + num joints, 256, 3, 1) ReLU Conv2d(256, 256, 2, 2) ReLU ConvTranspose2d(256, num joints, 4, 2)
Fusing Module	Conv2d(2 $\times$ num joints, 256, 3, 1) ReLU Conv2d(256, 256, 2, 2) ReLU ConvTranspose2d(256, num joints, 4, 2)

Table A.1: Network architecture details on the branches in our model. Each layer shows the number of input features, output features, kernel size, and stride.

Det. Thresh	Num Det.	AP	AP50	AP75	AR1	AR10
0.9	49.9k	0.021	0.053	0.013	0.149	0.293
0.75	53.9k	0.022	0.056	0.013	0.153	0.311
0.5	57.6k	0.023	0.059	0.013	0.157	0.325
0.25	61.7k	0.024	0.061	0.014	0.159	0.335

Table A.2: Object Detection Metrics of Hand Detections across detection thresholds

During our experiments, we found that training only using the ground truth as prior biased the model to expect perfect data. Conversely, training the model on its predictions only caused those inputs to be completely ignored because it never provided any useful data. Therefore, we found it beneficial to bootstrap our model with the ground truth priors but adopting a linear scheduling strategy to gradually introduce the model's own predictions. At each epoch, we select the model's prediction as a prior with a probability of $p = 1 - 0.10 * epoch$. By the 10th epoch, the model is training will all its predictions. We found this to be a good balance that slowly taught the model to use prior data even when imperfect.

A.3 Performance of Object Detector on dataset

In our results, we showed scores using both a "perfect detection" and "object detection" system. A bottleneck in our system is the performance of the pre-trained hand object detector [164], which shows room for improvement on our data. Table A.2 follows the detection evaluation format of MSCOCO [182]. The most notable issue is the low recall across all detection thresholds. The detector used was not trained on our data, which introduces many novel and difficult scenes. We show the number of detections filtered through for each detection threshold in the second column. AP computes the average precision at IoU thresholds 0.5 to 0.9 (in increments of 0.05). While AP50 and AP75 use only IoU thresholds 0.5 and 0.75, respectively. AR1 and AR10 measure the average recall on images with a maximum of 1 and 10 detections per image.

A.4 Added optical flow

We re-implemented part of the baseline [47], to include optical flow during tracking, although this has no impact on the output of the pose estimation model. We show these results in Table A.3.

Unexpectedly, adding optical flow to the tracking has a generally negative impact on the overall scores. We suspect that this is because the average size of our objects in relation to the image resolution. Our dataset consists of close-up views of hands in the surgical scene. The average

Model	Matching Strategy	Perfect Det.		Object Det.	
		mAP	MOTA	mAP	MOTA
Baseline	IoU	53.59 (-0.93)	38.27 (-0.16)	48.15 (-0.71)	31.46 (-0.07)
	L2	52.65 (+0.0)	37.78 (-0.01)	47.44 (+0.0)	31.14 (-0.04)
	GCN	52.65 (+0.0)	36.78 (+0.0)	47.44 (-0.79)	30.03 (-0.28)
	GCN-Joint Visual	52.65 (+0.93)	36.64 (-0.1)	47.44 (+0.0)	30.17 (-0.12)
Our Model	IoU	**56.66** (+0.0)	**39.31** (-0.18)	50.04 (+0.0)	**33.19** (-0.07)
	L2	**56.66** (+0.0)	38.94 (-0.01)	50.04 (+0.0)	32.84 (-0.04)
	GCN	**56.66** (+0.0)	38.22 (+0.04)	50.04 (-0.66)	32.24 (-0.25)
	GCN-Joint Visual	**56.66** (+0.86)	38.25 (-0.06)	**51.28** (-1.25)	32.39 (-0.09)

Table A.3: MOTA performance between matching strategies, averaged across all folds. Each row is optimized for highest MOTA performance. Matching strategies share the same base model, so it is possible for them to share the same mAP score. Parenthesis show added optical flow.

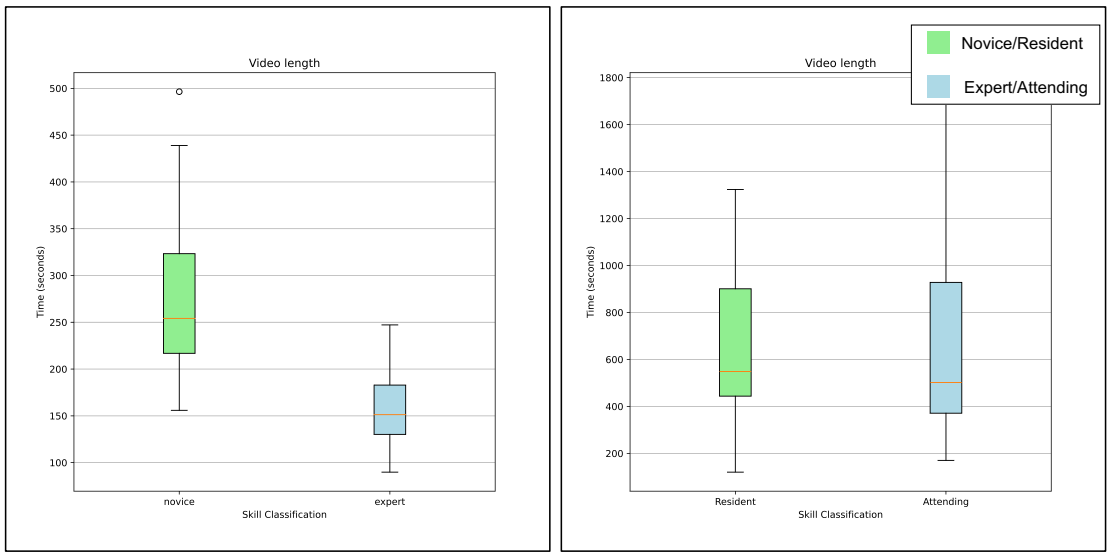
size of each hand bounding box is 334^2px while the average area of our frames are 891^2px. We hypothesize that the small optical flow shifts of the detected boxes and poses, on average, is more likely to incorrectly adjust these positions.

APPENDIX B

Contrastive Learning for Video-Based Skill Assessment in Open Cardiac Surgery

B.1 Bias in video length

There exists a bias on the VTS dataset, where novice videos are noticeably longer than expert videos. In Figure B.1, we share a box plot showing the range (in seconds) of videos labeled as novice (green) or as expert (blue). Learning-based methods may isolate this feature and ignore others factors that contribute to surgical skill. Unlike VTS (a), the total length of each class in **COSSA** (b) is roughly comparable.

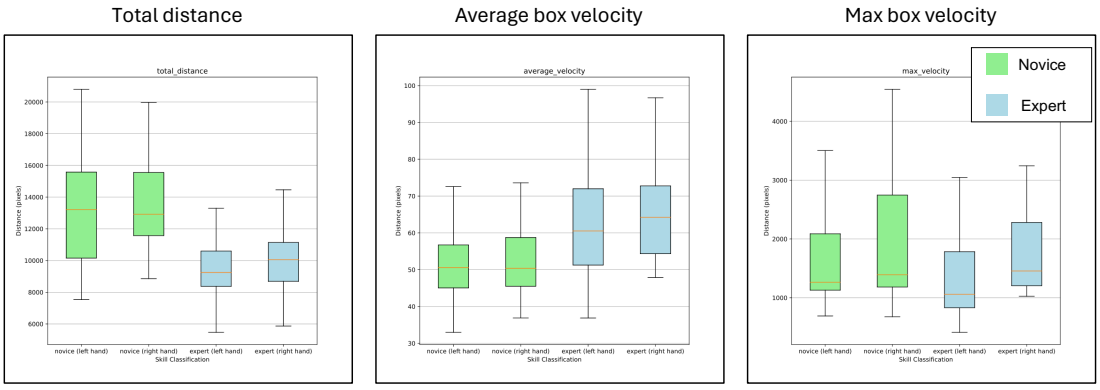


(a) VTS dataset (b) COSSA

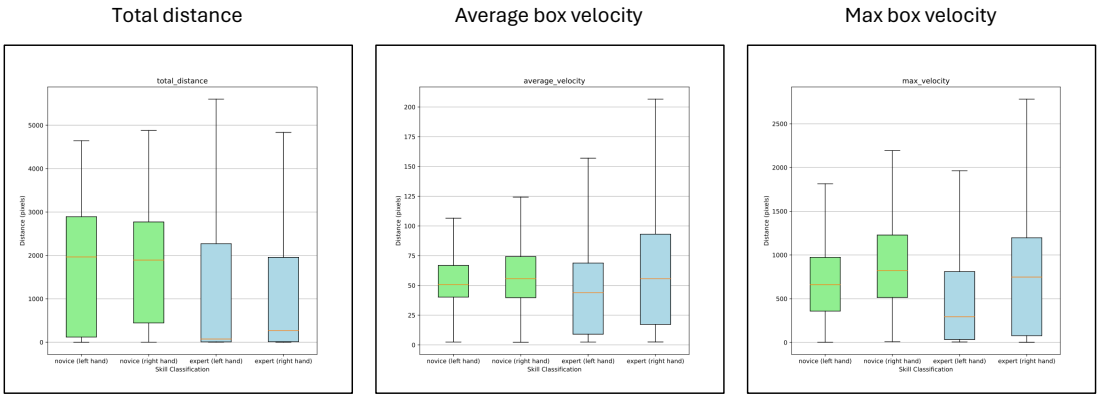
Figure B.1: Video lengths between novices and experts are easily separable on VTS (a) than they are on **COSSA**. This bias can prevent learning-based methods from discovering meaningful characteristics correlated with technical skill.

Clip-level statistics, in Figure B.2 (b), shows a significant reduction in length bias but retains meaningful differences with average and maximum velocity of the hand tracks. Interestingly, for

the full videos the experts have a higher average velocity than novices but in equal length clips they have a lower average velocity. This may indicate that the majority of expert movements are tempered, with distinct phases of swift motions.



(a) Full video statistics



(a) Clip-level statistics

Figure B.2: We compare full video statistics (a) and clip-level statistics (b) for the left and right hands between novices and experts. We note a higher total distance bias in full videos which we significantly reduce with clip-level sampling, while maintaining notable differences in average and maximum tracking velocity.

B.2 Longest T hand tracks

We show results using the the T-longest tracks from each clip in Tables B.1 and B.2. This allows for multiple relevant hands and accounts for tracking failures. In the main text, we report only the T with the highest accuracy.

Table B.1: Detailed skill classification results on the VTS dataset.

		NN-Sup.			NN-Con. (Ours)		
Inputs	Num Tracks	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Hand Box	2	**80.8%**	0.86	0.83	81.8%	0.83	0.83
	10	**80.8%**	0.84	0.83	80.3%	0.83	0.83
	30	80.0%	0.83	0.83	**85.5%**	0.86	0.87
Hand Pose	2	60.5%	0.62	0.62	**79.8%**	0.83	0.82
	10	64.0%	0.67	0.65	71.8%	0.76	0.75
	30	**71.3%**	0.79	0.75	66.8%	0.77	0.71
Hand Box + Pose	2	62.3%	0.74	0.67	71.8%	0.77	0.74
	10	**76.0%**	0.80	0.78	**76.5%**	0.80	0.78
	30	75.3%	0.81	0.78	70.3%	0.75	0.72

Table B.2: Detailed skill classification on the **COSSA** dataset.

		NN-Sup.			NN-Con. (Ours)		
Inputs	Num Tracks	Accuracy	Precision	Recall	Accuracy	Precision	Recall
Hand Box	2	62.5%	0.48	0.52	**70.7%**	0.69	0.65
	10	62.8%	0.48	0.52	66.0%	0.60	0.57
	30	**66.0%**	0.74	0.57	67.2%	0.55	0.60
Hand Pose	2	57.5%	0.48	0.52	62.6%	0.67	0.58
	10	55.6%	0.48	0.48	**69.1%**	0.73	0.61
	30	**64.1%**	0.48	0.54	61.0%	0.50	0.54
Hand Box + Pose	2	60.6%	0.41	0.52	63.4%	0.62	0.60
	10	62.6%	0.72	0.56	66.0%	0.55	0.58
	30	**66.4%**	0.69	0.60	**68.1%**	0.71	0.66

B.3 Surgical Skill Assessment

We show results for surgical skill assessment in Table B.3, measuring performance using Mean Absolute Error (MAE) on the first three OSATS categories. Due to the narrow distribution of the OSATS scores, the mean training label baseline is very strong. However, we demonstrate a clear improvement over NN-Supervised using our NN-Contrastive approach. Although we do not explicitly measure tissue deformation or track instruments, the hand track metrics are tightly coupled to these OSATS categories. The lowest errors are achieved using hand pose features which indicates that the nuanced assessment task requires finer detection information. We suspect as more varied OSATS labels are collected, our learning based methods will surpass this simple baseline. For surgical skill assessment, we perform better than our supervised counterpart but nearly on par with the naive baseline.

Method	Respect for Tissue	Time and Motion	Instrument Handling
Baseline - Mean	**0.42**	**0.48**	**0.49**
NN-Supervised	0.78	0.91	1.04
NN-Contrastive (Ours)	0.51	0.52	0.51

Table B.3: MAE for OSATS prediction on **COSSA**. Averaged cross 5-folds. (Lower is better).

B.4 OSATS Distribution

In Table B.4, we show the distribution of OSATS scores from **COSSA** dataset is skewed towards the higher end. A primary reason is that only surgeons with a certain skill level, including novices who have completed medical school training, perform live operations.

Table B.4: Distribution of OSATS scores ($n = 14$).

OSATS Category	Min	Median	Mean	Max
Respect for Tissue	3.5	4.5	4.54	5
Time and Motion	3	4	4.14	5
Instrument Handling	3	4.5	4.39	5
Knowledge of Instruments	4	5	4.75	5
Flow of Operation	3	4.75	4.5	5
Use of Assistants	3.5	4.5	4.43	5
Knowledge of Specific Procedure	3.5	5	4.64	5

B.5 Video Transformer Attention

In Figure B.3 and B.4, we show visualization of attention from the Transformer on the VTS dataset. The heavy attention on the corner regions of the image, for novice predictions, or only two sections of simulation device may indicates some visual bias not related to skill.

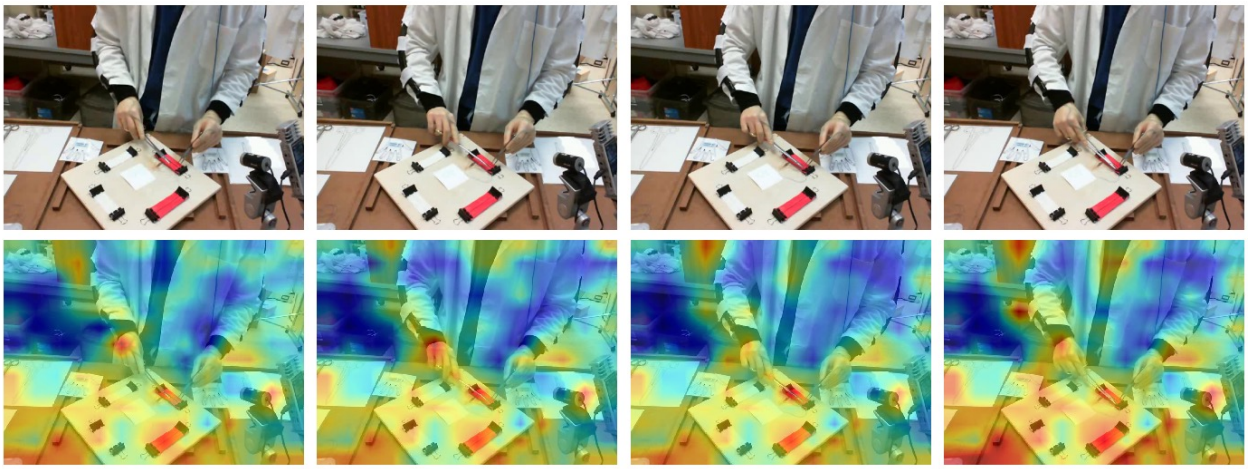


Figure B.3: Visualization of spatio-temporal attention from the transformer.

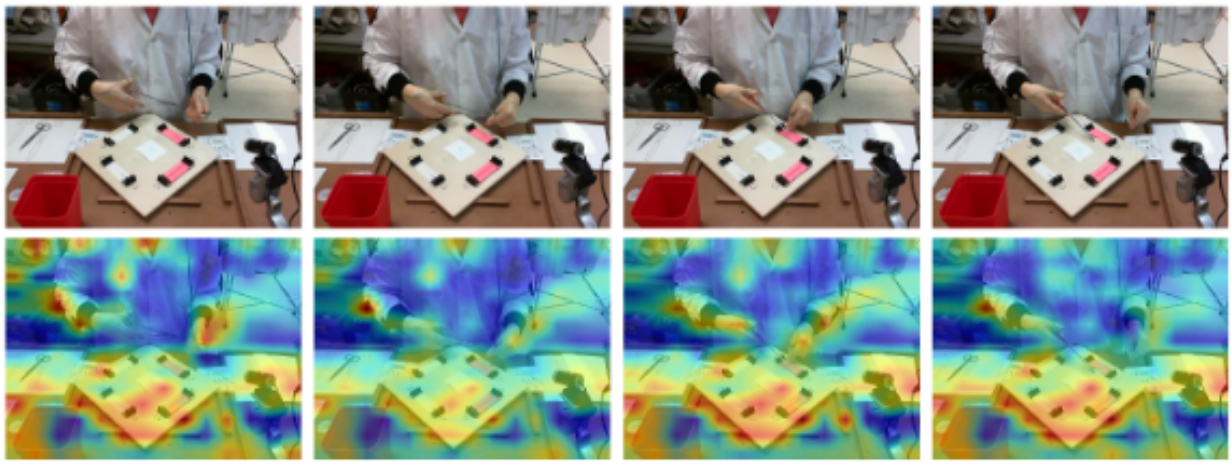


Figure B.4: Visualization of spatio-temporal attention from the transformer.

APPENDIX C

Ground Reaction Force Estimation via Multi-task Learning

C.1 Hyper-parameters

C.1.1 Input receptive field

When training all transformer models, we experiment with four input receptive fields on the *ForcePose* dataset; $9, 27, 43, 81$ frames. As shown in Table C.1, the longer sequences result in lower average sequence losses across all trained models. We report only the best (lowest) loss in the manuscript, hence all transformer models shown are trained on 81 frames. Across the different training strategies, our multi-task learning (MTL) consistently produces the lowest average sequence loss.

We also include the results over the various receptive fields for zero-shot learning in Table C.2, our results are consistent with the transformer models trained on all classes.

C.1.2 Loss function thresholds

The gated-MSE loss includes a threshold hyper-parameter, T, in the loss function:

$$\mathcal{L}_f = \sum_{T} w_t \cdot \frac{1}{|\mathbf{F}_{\delta_t}|} \|\mathbf{F}_{\delta_t} - \hat{\mathbf{F}}_{\delta_t}\|_2^2. \tag{C.1}$$

T specifies the number of terms to sum in the total loss, from the sequence $\delta = [0, 1, 5, 10, 15, ...]$ (N/kg). Again, $T = 1$ is equivalent to the MSE and $w_t = 1/T$. In Table C.3 we show the results of

Training Strategy	9 frames	27 frames	43 frames	81 frames
Transformer random weights	92.67 ± 1.97	85.82 ± 1.45	80.91 ± 0.51	80.45 ± 1.08
Transformer pre-trained	92.71 ± 1.01	84.00 ± 0.65	80.99 ± 0.39	80.51 ± 0.61
Transformer MTL	**90.69** ±0.80	**81.73** ±0.59	**78.51** ±0.55	**77.95** ±0.36

Table C.1: Sweep input receptive field

9 frames	CMJ	SLS	SLJ	SJ	Squat	Average
Random Weights	**139.69**	105.90	228.47	129.34	**78.88**	136.45
Pre-trained Weights	146.86	111.72	**217.26**	129.96	101.75	141.51
Multi-task Learning	140.42	**102.07**	223.66	**127.52**	87.69	**136.27**
27 frames						
Random Weights	132.00	**97.19**	222.82	120.63	72.59	129.04
Pre-trained Weights	126.92	111.40	224.67	**115.28**	73.47	130.34
Multi-task Learning	**120.80**	117.44	**217.74**	116.91	**68.85**	**128.35**
43 frames						
Random Weights	114.37	**104.99**	215.66	**104.34**	76.14	123.10
Pre-trained Weights	109.96	116.25	215.90	105.74	68.22	123.21
Multi-task Learning	**109.67**	109.05	**211.29**	108.70	**67.53**	**121.25**

Table C.2: Zero-shot learning, sweep input receptive field (single run)

sweeping through the $T = 1...3$. Across the trained models, we find $T = 2$ provides the greatest reduction in mean k-peaks while minimizing the additional cost in RMSE fit.

Prediction Network	RMSE (N)	1-peak (N)	3-peak (N)	5-peak (N)
Transformer random weights	**80.45** $\pm$ 1.08	333.12	196.67	155.83
Transformer random weights (our loss, T=2)	83.95 $\pm$ 3.80	**282.92**	**174.95**	**143.83**
Transformer random weights (our loss, T=3)	89.68 $\pm$ 4.80	296.72	177.99	147.04
Transformer MTL	**77.95** $\pm$ 0.36	321.19	190.14	154.03
Transformer MTL (our loss, T=2)	84.95 $\pm$ 4.35	285.10	170.52	140.83
Transformer MTL (our loss, T=3)	86.23 $\pm$ 1.11	**277.25**	**170.10**	**139.57**
Transformer pre-trained	**80.51** $\pm$ 0.61	320.41	189.99	151.40
Transformer pre-trained (our loss, T=2)	83.20 $\pm$ 0.39	**285.72**	171.71	139.23
Transformer pre-trained (our loss, T=3)	84.58 $\pm$ 1.87	293.20	**170.15**	**134.57**

Table C.3: Sweep threshold, T, in gated-MSE loss

BIBLIOGRAPHY

[1] *NFL Announces Fifth Annual Big Data Bowl Competition — NFL Football Operations*, Oct. 2022. [Online]. Available: https://operations.nfl.com/updates/football-ops/nfl-announces-fifth-annual-big-data-bowl-competition/ (visited on 03/19/2023).

[2] *SafelyYou Launches SafelyYou Aware™ to Support and Transform Clinical Care at Senior Living Communities with Remote, Hourly Nighttime Wellness Checks — Business Wire*. [Online]. Available: https://www.businesswire.com/news/home/20230112005299/en/ (visited on 03/19/2023).

[3] C. Liao and W. Ke, *PowerArena uses AI image ID tech to diagnose production inefficiency*, en, Dec. 2018. [Online]. Available: https://www.digitimes.com/news/a20181212PD207.html (visited on 03/19/2023).

[4] R. Whissell, S. Lyons, D. Wilkinson, and C. Whissell, "National Bias in Judgments of Olympic-Level Skating," *Perceptual and Motor Skills*, vol. 77, no. 2, pp. 355–358, Oct. 1993, Publisher: SAGE Publications Inc, ISSN: 0031-5125. DOI: 10.2466/pms.1993.77.2.355. [Online]. Available: https://doi.org/10.2466/pms.1993.77.2.355 (visited on 03/22/2023).

[5] S. A. D. Van Veen, "Ice dance reacts to the 2002 olympic judging scandal: A study of skaters' movement practices under the new isu judging system," Ph.D. dissertation, University of British Columbia, 2012.

[6] H. Pirsiavash, C. Vondrick, and A. Torralba, "Assessing the quality of actions," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*, Springer, 2014, pp. 556–571.

[7] P. Parmar and B. Tran Morris, "Learning to score olympic events," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 20–28.

[8] J.-H. Pan, J. Gao, and W.-S. Zheng, "Action assessment by joint relation graphs," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6331–6340.

[9] Y. Gao, S. S. Vedula, C. E. Reiley, *et al.*, "Jhu-isi gesture and skill assessment working set (jigsaws): A surgical activity dataset for human motion modeling," in *Miccai workshop: M2cai*, vol. 3, 2014, p. 3.

[10] A. Jin, S. Yeung, J. Jopling, *et al.*, "Tool detection and operative skill assessment in surgical videos using region-based convolutional neural networks," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, 2018, pp. 691–699.

[11] S. Khalid, M. Goldenberg, T. Grantcharov, B. Taati, and F. Rudzicz, "Evaluation of deep learning models for identifying surgical actions and measuring performance," *JAMA Network Open*, vol. 3, no. 3, e201664–e201664, 2020.

[12] D. Liu, Q. Li, T. Jiang, *et al.*, "Towards unified surgical skill assessment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9522–9531.

[13] Y. Sharma, V. Bettadapura, T. Plötz, *et al.*, "Video based assessment of osats using sequential motion textures," Georgia Institute of Technology, 2014.

[14] H. Doughty, D. Damen, and W. Mayol-Cuevas, "Who's better? who's best? pairwise deep ranking for skill determination," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6057–6066.

[15] H. Doughty, W. Mayol-Cuevas, and D. Damen, "The pros and cons: Rank-aware temporal attention for skill determination in long videos," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 7862–7871.

[16] D.-S. Komaris, G. Tarfali, B. O'Flynn, and S. Tedesco, "Unsupervised IMU-based evaluation of at-home exercise programmes: A feasibility study," *BMC Sports Science, Medicine and Rehabilitation*, vol. 14, no. 1, p. 28, Feb. 2022, ISSN: 2052-1847. DOI: 10.1186/s13102-022-00417-1. [Online]. Available: https://doi.org/10.1186/s13102-022-00417-1 (visited on 03/19/2023).

[17] F. M. Clemente, Z. Akyildiz, J. Pino-Ortega, and M. Rico-González, "Validity and Reliability of the Inertial Measurement Unit for Barbell Velocity Assessments: A Systematic Review," *Sensors (Basel, Switzerland)*, vol. 21, no. 7, p. 2511, Apr. 2021, ISSN: 1424-8220. DOI: 10.3390/s21072511. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8038306/ (visited on 03/19/2023).

[18] C. Crema, A. Depari, A. Flammini, E. Sisinni, T. Haslwanter, and S. Salzmann, "IMU-based solution for automatic detection and classification of exercises in the fitness scenario," in *2017 IEEE Sensors Applications Symposium (SAS)*, Mar. 2017, pp. 1–6. DOI: 10.1109/SAS.2017.7894068.

[19] C. Glaeser, *A Buyer's Guide to IMU Sport Sensor Devices for Professionals*, en-US, Sep. 2018. [Online]. Available: https://simplifaster.com/articles/buyers-guide-imu-sensor-devices/ (visited on 03/19/2023).

[20] *How to Squat: Muscles Worked & Proper Form*, https://www.strengthlog.com/squat/, [Accessed 18-02-2024].

[21] A. Goldbraikh, A.-L. D'Angelo, C. M. Pugh, and S. Laufer, "Video-based fully automatic assessment of open surgery suturing skills," *International Journal of Computer Assisted Radiology and Surgery*, vol. 17, no. 3, pp. 437–448, 2022.

[22] D. P. Azari, B. L. Miller, B. V. Le, *et al.*, "A comparison of expert ratings and marker-less hand tracking along osats-derived motion scales," *IEEE Transactions on Human-Machine Systems*, vol. 51, no. 1, pp. 22–31, 2020.

[23] A. Goldbraikh, T. Volk, C. M. Pugh, and S. Laufer, "Using open surgery simulation kinematic data for tool and gesture recognition," *International Journal of Computer Assisted Radiology and Surgery*, vol. 17, no. 6, pp. 965–979, 2022.

[24] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, "Evaluating surgical skills from kinematic data using convolutional neural networks," in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part IV 11*, Springer, 2018, pp. 214–221.

[25] A. S. Mattingly, M. M. Chen, V. Divi, F. C. Holsinger, and A. Saraswathula, "Minimally invasive surgery in the united states, 2022: Understanding its value using new datasets," *Journal of Surgical Research*, vol. 281, pp. 33–36, 2023.

[26] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[27] Z. Wang, Z. Cao, Y. Hao, and D. Sadigh, "Weakly supervised correspondence learning," in *2022 International Conference on Robotics and Automation (ICRA)*, IEEE, 2022, pp. 469–476.

[28] *CMU-Perceptual-Computing-Lab/openpose*, original-date: 2017-04-24T14:06:31Z, Mar. 2023. [Online]. Available: https://github.com/CMU-Perceptual-Computing-Lab/openpose (visited on 03/21/2023).

[29] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD Birds-200-2011 Dataset," en,

[30] N. D. Reddy, M. Vo, and S. G. Narasimhan, "CarFusion: Combining Point Tracking and Part Detection for Dynamic 3D Reconstruction of Vehicles," en, in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA: IEEE, Jun. 2018, pp. 1906–1915, ISBN: 978-1-5386-6420-9. DOI: 10.1109/CVPR.2018.00204. [Online]. Available: https://ieeexplore.ieee.org/document/8578302/ (visited on 03/21/2023).

[31] M. Andriluka, S. Roth, and B. Schiele, "Pictorial structures revisited: People detection and articulated pose estimation," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, ISSN: 1063-6919, Jun. 2009, pp. 1014–1021. DOI: 10.1109/CVPR.2009.5206754.

[32] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, "Efficient object localization using Convolutional Networks," en, in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA: IEEE, Jun. 2015, pp. 648–656, ISBN: 978-1-4673-6964-0. DOI: 10.1109/CVPR.2015.7298664. [Online]. Available: http://ieeexplore.ieee.org/document/7298664/ (visited on 03/22/2023).

[33] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2016, pp. 4724–4732.

[34] C. Morris, M. Mundt, M. Goldacre, J. Weber, A. Mian, and J. Alderson, "Predicting 3d ground reaction force from 2d video via neural networks in sidestepping tasks," *ISBS Proceedings Archive*, vol. 39, no. 1, p. 300, 2021.

[35] M. Goldacre, A. El-Sallam, H. Wyatt, *et al.*, "Predicting ground reaction forces from 2d video: Bridging the lab to field nexus," *ISBS Proceedings Archive*, vol. 39, no. 1, p. 9, 2021.

[36] H. Jeong and S. Park, "Estimation of the ground reaction forces from a single video camera based on the spring-like center of mass dynamics of human walking," *Journal of Biomechanics*, vol. 113, p. 110 074, 2020.

[37] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 7, pp. 1325–1339, 2013.

[38] E. Coumans and Y. Bai, *Pybullet, a python module for physics simulation for games, robotics and machine learning*, http://pybullet.org, 2021.

[39] J. Martin, G. Regehr, R. Reznick, *et al.*, "Objective structured assessment of technical skill (osats) for surgical residents," *British journal of surgery*, vol. 84, no. 2, pp. 273–278, 1997.

[40] D. P. Azari, L. L. Frasier, S. R. P. Quamme, *et al.*, "Modeling surgical technical skill using expert assessment for automated computer rating," *Annals of surgery*, vol. 269, no. 3, p. 574, 2019.

[41] E. Bkheet, A.-L. D'Angelo, A. Goldbraikh, and S. Laufer, "Pose estimation for surgical training," *arXiv preprint arXiv:2211.07021*, 2022.

[42] E. D. Goodman, K. K. Patel, Y. Zhang, *et al.*, "Analyzing surgical technique in diverse open surgical videos with multitask machine learning," *JAMA surgery*, 2023.

[43] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, "Hand keypoint detection in single images using multiview bootstrapping," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 1145–1153.

[44] F. Gomez-Donoso, S. Orts-Escolano, and M. Cazorla, "Large-scale multiview 3d hand pose dataset," *Image and Vision Computing*, vol. 81, pp. 25–33, 2019.

[45] C. Zimmermann, D. Ceylan, J. Yang, B. Russell, M. Argus, and T. Brox, "Freihand: A dataset for markerless capture of hand pose and shape from single rgb images," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 813–822.

[46] J. Zhang, J. Jiao, M. Chen, L. Qu, X. Xu, and Q. Yang, "A hand pose tracking benchmark from stereo matching," in *2017 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2017, pp. 982–986.

[47] B. Xiao, H. Wu, and Y. Wei, "Simple baselines for human pose estimation and tracking," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 466–481.

[48] Z. Li, J. Sedlar, J. Carpentier, I. Laptev, N. Mansard, and J. Sivic, "Estimating 3d motion and forces of person-object interactions from monocular video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8640–8649.

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

[50] D. Rempe, L. J. Guibas, A. Hertzmann, B. Russell, R. Villegas, and J. Yang, "Contact and human dynamics from monocular video," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, Springer, 2020, pp. 71–87.

[51] Y. Yuan, S.-E. Wei, T. Simon, K. Kitani, and J. Saragih, "Simpoe: Simulated character control for 3d human pose estimation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 7159–7169.

[52] S. Shimada, V. Golyanik, W. Xu, and C. Theobalt, "Physcap: Physically plausible monocular 3d motion capture in real time," *ACM Transactions on Graphics (ToG)*, vol. 39, no. 6, pp. 1–16, 2020.

[53] G. Bertasius, C. Feichtenhofer, D. Tran, J. Shi, and L. Torresani, "Learning temporal pose estimation from sparsely-labeled videos," in *Advances in Neural Information Processing Systems*, 2019, pp. 3027–3038.

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

[55] G. Ning, J. Pei, and H. Huang, "Lighttrack: A generic framework for online top-down human pose tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 1034–1035.

[56] M. Wang, J. Tighe, and D. Modolo, "Combining detection and tracking for human pose estimation in videos," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 088–11 096.

[57] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7291–7299.

[58] Y. Raaj, H. Idrees, G. Hidalgo, and Y. Sheikh, "Efficient online multi-person 2d pose tracking with recurrent spatio-temporal affinity fields," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4620–4628.

[59] S. Jin, W. Liu, W. Ouyang, and C. Qian, "Multi-person articulated tracking with spatial and temporal embeddings," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5664–5673.

[60] U. Iqbal, A. Milan, and J. Gall, "Posetrack: Joint multi-person pose estimation and tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2011–2020.

[61] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *TPAMI*, vol. 39, no. 6, pp. 1137–1149, 2017.

[62] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, *Detectron2*, `https://github.com/facebookresearch/detectron2`, 2019.

[63] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 1, pp. 172–186, 2019.

[64] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *International Conference on Learning Representations*, 2017.

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

[66] Y. Sun, Q. Bao, W. Liu, Y. Fu, B. Michael J., and T. Mei, "Monocular, One-stage, Regression of Multiple 3D People," in *ICCV*, 2021.

[67] D. Pavllo, C. Feichtenhofer, D. Grangier, and M. Auli, "3d human pose estimation in video with temporal convolutions and semi-supervised training," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[68] E. Remelli, S. Han, S. Honari, P. Fua, and R. Wang, "Lightweight multi-view 3d pose estimation through camera-disentangled representation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 6040–6049.

[69] G. Pavlakos, X. Zhou, K. G. Derpanis, and K. Daniilidis, "Harvesting multiple views for marker-less 3d human pose annotations," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 6988–6997.

[70] K. Iskakov, E. Burkov, V. Lempitsky, and Y. Malkov, "Learnable triangulation of human pose," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 7718–7727.

[71] H. Qiu, C. Wang, J. Wang, N. Wang, and W. Zeng, "Cross view fusion for 3d human pose estimation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 4342–4351.

[72] J. Zhang, Y. Cai, S. Yan, J. Feng, *et al.*, "Direct multi-view multi-person 3d pose estimation," *Advances in Neural Information Processing Systems*, vol. 34, pp. 13 153–13 164, 2021.

[73] R. Liu, J. Shen, H. Wang, C. Chen, S.-c. Cheung, and V. Asari, "Attention mechanism exploits temporal contexts: Real-time 3d human pose reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5064–5073.

[74] J. Wang, S. Yan, Y. Xiong, and D. Lin, "Motion guided 3d pose estimation from videos," in *European Conference on Computer Vision*, Springer, 2020, pp. 764–780.

[75] C. Zheng, S. Zhu, M. Mendieta, T. Yang, C. Chen, and Z. Ding, "3d human pose estimation with spatial and temporal transformers," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Los Alamitos, CA, USA: IEEE Computer Society, Oct. 2021, pp. 11 636–11 645. DOI: 10.1109/ICCV48922.2021.01145. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/ICCV48922.2021.01145.

[76] G. Moon and K. M. Lee, "I2l-meshnet: Image-to-lixel prediction network for accurate 3d human pose and mesh estimation from a single rgb image," in *European Conference on Computer Vision*, Springer, 2020, pp. 752–768.

[77] G. Pavlakos, X. Zhou, and K. Daniilidis, "Ordinal depth supervision for 3d human pose estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7307–7316.

[78] A. Zanfir, E. G. Bazavan, M. Zanfir, W. T. Freeman, R. Sukthankar, and C. Sminchisescu, "Neural descent for visual 3d human pose and shape," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 484–14 493.

[79] N. Kolotouros, G. Pavlakos, M. J. Black, and K. 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*, 2019, pp. 2252–2261.

[80] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "Smpl: A skinned multi-person linear model," *ACM transactions on graphics (TOG)*, vol. 34, no. 6, pp. 1–16, 2015.

[81] H. Xu, E. G. Bazavan, A. Zanfir, W. T. Freeman, R. Sukthankar, and C. Sminchisescu, "Ghum & ghuml: Generative 3d human shape and articulated pose models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6184–6193.

[82] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black, "Amass: Archive of motion capture as surface shapes," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 5442–5451.

[83] H. Tu, C. Wang, and W. Zeng, "Voxelpose: Towards multi-camera 3d human pose estimation in wild environment," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, Springer, 2020, pp. 197–212.

[84] S. Shimada, V. Golyanik, W. Xu, P. Pérez, and C. Theobalt, "Neural monocular 3d human motion capture with physical awareness," *ACM Transactions on Graphics (ToG)*, vol. 40, no. 4, pp. 1–15, 2021.

[85] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ international conference on intelligent robots and systems*, IEEE, 2012, pp. 5026–5033.

[86] E. Gärtner, M. Andriluka, E. Coumans, and C. Sminchisescu, "Differentiable dynamics for articulated 3d human motion reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 13 190–13 200.

[87] Z. Li, J. Sedlar, J. Carpentier, I. Laptev, N. Mansard, and J. Sivic, "Estimating 3d motion and forces of human-object interactions from internet videos," *International Journal of Computer Vision*, vol. 130, no. 2, pp. 363–383, 2022.

[88] K. Xie, T. Wang, U. Iqbal, Y. Guo, S. Fidler, and F. Shkurti, "Physics-based human motion estimation and synthesis from videos," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 532–11 541.

[89] E. Gärtner, M. Andriluka, H. Xu, and C. Sminchisescu, "Trajectory optimization for physics-based reconstruction of 3d human pose from monocular video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 13 106–13 115.

[90] X. B. Peng, P. Abbeel, S. Levine, and M. Van de Panne, "Deepmimic: Example-guided deep reinforcement learning of physics-based character skills," *ACM Transactions On Graphics (TOG)*, vol. 37, no. 4, pp. 1–14, 2018.

[91] Y. Yuan and K. Kitani, "Residual force control for agile human behavior imitation and extended motion synthesis," *Advances in Neural Information Processing Systems*, vol. 33, pp. 21 763–21 774, 2020.

[92] K. Werling, D. Omens, J. Lee, I. Exarchos, and C. K. Liu, "Fast and feature-complete differentiable physics for articulated rigid bodies with contact," *arXiv preprint arXiv:2103.16021*, 2021.

[93] S. Tripathi, L. Müller, C.-H. P. Huang, O. Taheri, M. J. Black, and D. Tzionas, "3d human pose estimation via intuitive physics," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4713–4725.

[94] A. L. Hof, M. Gazendam, and W. Sinke, "The condition for dynamic stability," *Journal of biomechanics*, vol. 38, no. 1, pp. 1–8, 2005.

[95] A. L. Hof, "The equations of motion for a standing human reveal three mechanisms for balance," *Journal of biomechanics*, vol. 40, no. 2, pp. 451–457, 2007.

[96] D. A. Winter, "Human balance and posture control during standing and walking," *Gait & posture*, vol. 3, no. 4, pp. 193–214, 1995.

[97] N. Santavas, I. Kansizoglou, L. Bampis, E. Karakasis, and A. Gasteratos, "Attention! a lightweight 2d hand pose estimation approach," *arXiv preprint arXiv:2001.08047*, 2020.

[98] G. Moon, S.-I. Yu, H. Wen, T. Shiratori, and K. M. Lee, "Interhand2.6m: A dataset and baseline for 3d interacting hand pose estimation from a single rgb image," in *European Conference on Computer Vision (ECCV)*, 2020.

[99] L. Tao, E. Elhamifar, S. Khudanpur, G. D. Hager, and R. Vidal, "Sparse hidden markov models for surgical gesture classification and skill evaluation," in *International conference on information processing in computer-assisted interventions*, Springer, 2012, pp. 167–177.

[100] L. Zappella, B. Béjar, G. Hager, and R. Vidal, "Surgical gesture classification from video and kinematic data," *Medical image analysis*, vol. 17, no. 7, pp. 732–745, 2013.

[101] G. Forestier, F. Petitjean, P. Senin, *et al.*, "Surgical motion analysis using discriminative interpretable patterns," *Artificial intelligence in medicine*, vol. 91, pp. 3–11, 2018.

[102] D. Sarikaya, J. J. Corso, and K. A. Guru, "Detection and localization of robotic tools in robot-assisted surgery videos using deep neural networks for region proposal and detection," *IEEE transactions on medical imaging*, vol. 36, no. 7, pp. 1542–1549, 2017.

[103] I. Laina, N. Rieke, C. Rupprecht, *et al.*, "Concurrent segmentation and localization for tracking of surgical instruments," in *International conference on medical image computing and computer-assisted intervention*, Springer, 2017, pp. 664–672.

[104] Z.-L. Ni, G.-B. Bian, X.-L. Xie, Z.-G. Hou, X.-H. Zhou, and Y.-J. Zhou, "Rasnet: Segmentation for tracking surgical instruments in surgical videos using refined attention segmentation network," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, 2019, pp. 5735–5738.

[105] X. Du, T. Kurmann, P.-L. Chang, *et al.*, "Articulated multi-instrument 2-d pose estimation using fully convolutional networks," *IEEE transactions on medical imaging*, vol. 37, no. 5, pp. 1276–1287, 2018.

[106] E. Colleoni, S. Moccia, X. Du, E. De Momi, and D. Stoyanov, "Deep learning based robotic tool detection and articulation estimation with spatio-temporal layers," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2714–2721, 2019.

[107] R. Richa, M. Balicki, E. Meisner, R. Sznitman, R. Taylor, and G. Hager, "Visual tracking of surgical tools for proximity detection in retinal surgery," in *International Conference on Information Processing in Computer-Assisted Interventions*, Springer, 2011, pp. 55–66.

[108] R. Sznitman, R. Richa, R. H. Taylor, B. Jedynak, and G. D. Hager, "Unified detection and tracking of instruments during retinal microsurgery," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 5, pp. 1263–1273, 2012.

[109] C. I. Nwoye, D. Mutter, J. Marescaux, and N. Padoy, "Weakly supervised convolutional lstm approach for tool tracking in laparoscopic videos," *International journal of computer assisted radiology and surgery*, vol. 14, no. 6, pp. 1059–1067, 2019.

[110] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The clear mot metrics," *EURASIP Journal on Image and Video Processing*, vol. 2008, pp. 1–10, 2008.

[111] L. Ren, R. K. Jones, and D. Howard, "Whole body inverse dynamics over a complete gait cycle based only on measured kinematics," *Journal of biomechanics*, vol. 41, no. 12, pp. 2750–2759, 2008.

[112] W. R. Johnson, A. Mian, C. J. Donnelly, D. Lloyd, and J. Alderson, "Predicting athlete ground reaction forces and moments from motion capture," *Medical & biological engineering & computing*, vol. 56, no. 10, pp. 1781–1792, 2018.

[113] W. R. Johnson, J. Alderson, D. Lloyd, and A. Mian, "Predicting athlete ground reaction forces and moments from spatio-temporal driven cnn models," *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 3, pp. 689–694, 2018.

[114] R. S. Alcantara, W. B. Edwards, G. Y. Millet, and A. M. Grabowski, "Predicting continuous ground reaction forces from accelerometers during uphill and downhill running: A recurrent neural network solution," *bioRxiv*, 2021.

[115] A. Karatsidis, G. Bellusci, H. M. Schepers, M. De Zee, M. S. Andersen, and P. H. Veltink, "Estimation of ground reaction forces and moments during gait using only inertial motion capture," *Sensors*, vol. 17, no. 1, p. 75, 2017.

[116] D. Kim, H. Cho, H. Shin, S.-C. Lim, and W. Hwang, "An efficient three-dimensional convolutional neural network for inferring physical interaction force from video," *Sensors*, vol. 19, no. 16, p. 3579, 2019.

[117] J. L. McGinley, R. Baker, R. Wolfe, and M. E. Morris, "The reliability of three-dimensional kinematic gait measurements: A systematic review," *Gait & posture*, vol. 29, no. 3, pp. 360–369, 2009.

[118] G. Strutzenberger, R. Kanko, S. Selbie, H. Schwameder, and K. Deluzio, "Assessment of kinematic cmj data using a deep learning algorithm-based markerless motion capture system," *ISBS Proceedings Archive*, vol. 39, no. 1, p. 236, 2021.

[119] S. Wang, D. Yang, P. Zhai, C. Chen, and L. Zhang, "Tsa-net: Tube self-attention network for action quality assessment," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 4902–4910.

[120] Y. Tang, Z. Ni, J. Zhou, *et al.*, "Uncertainty-aware score distribution learning for action quality assessment," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9839–9848.

[121] X. Yu, Y. Rao, W. Zhao, J. Lu, and J. Zhou, "Group-aware contrastive regression for action quality assessment," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 7919–7928.

[122] A. Xu, L.-A. Zeng, and W.-S. Zheng, "Likert scoring with grade decoupling for long-term action assessment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 3232–3241.

[123] M. Fieraru, M. Zanfir, S. C. Pirlea, V. Olaru, and C. Sminchisescu, "Aifit: Automatic 3d human-interpretable feedback models for fitness training," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9919–9928.

[124] F. Sener, D. Chatterjee, D. Shelepov, *et al.*, "Assembly101: A large-scale multi-view video dataset for understanding procedural activities," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 21 096–21 106.

[125] D. P. Azari, B. L. Miller, B. V. Le, C. C. Greenberg, and R. G. Radwin, "Quantifying surgeon maneuevers across experience levels through marker-less hand motion kinematics of simulated surgical tasks," *Applied Ergonomics*, vol. 87, p. 103 136, 2020.

[126] D. P. Azari, Y. H. Hu, B. L. Miller, B. V. Le, and R. G. Radwin, "Using surgeon hand motions to predict surgical maneuvers," *Human factors*, vol. 61, no. 8, pp. 1326–1339, 2019.

[127] O. Chapelle, B. Scholkopf, and A. Zien, "Semi-supervised learning," 2010.

[128] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[129] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*, PMLR, 2020, pp. 1597–1607.

[130] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel, "Mixmatch: A holistic approach to semi-supervised learning," *Advances in neural information processing systems*, vol. 32, 2019.

[131] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[132] L. Zhou, N. Louis, and J. J. Corso, "Weakly-supervised video object grounding from text by loss weighting and object interaction," *arXiv preprint arXiv:1805.02834*, 2018.

[133] D. Lin, J. Dai, J. Jia, K. He, and J. Sun, "Scribblesup: Scribble-supervised convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3159–3167.

[134] Z. Kalal, K. Mikolajczyk, and J. Matas, "Forward-backward error: Automatic detection of tracking failures," in *2010 20th international conference on pattern recognition*, IEEE, 2010, pp. 2756–2759.

[135] P. Pan, F. Porikli, and D. Schonfeld, "Recurrent tracking using multifold consistency," in *Proceedings of the Eleventh IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, vol. 3, 2009.

[136] H. Wu, A. C. Sankaranarayanan, and R. Chellappa, "Online empirical evaluation of tracking algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 8, pp. 1443–1458, 2009.

[137] A. Bansal, S. Ma, D. Ramanan, and Y. Sheikh, "Recycle-gan: Unsupervised video retargeting," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 119–135.

[138] Q. Wang, B. Li, T. Xiao, *et al.*, "Learning deep transformer models for machine translation," *arXiv preprint arXiv:1906.01787*, 2019.

[139] T. Cheng, *Supervised, Semi-Supervised, Unsupervised, and Self-Supervised Learning — towardsdatascience.com*, `https://towardsdatascience.com/supervised-semi-supervised-unsupervised-and-self-supervised-learning-7fa79aa9247c`, [Accessed 19-02-2024].

[140] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[141] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[142] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4651–4659.

[143] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[144] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," in *NIPS*, 2017, pp. 6000–6010.

[145] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, "Videobert: A joint model for video and language representation learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7464–7473.

[146] J. Lu, D. Batra, D. Parikh, and S. Lee, "Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," *Advances in neural information processing systems*, vol. 32, 2019.

[147] A. Dosovitskiy, L. Beyer, A. Kolesnikov, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *ICLR*, 2021.

[148] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *European Conference on Computer Vision*, Springer, 2020, pp. 213–229.

[149] G. Bertasius, H. Wang, and L. Torresani, "Is space-time attention all you need for video understanding?" In *ICML*, vol. 2, 2021, p. 4.

[150] M. Snower, A. Kadav, F. Lai, and H. P. Graf, "15 keypoints is all you need," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6738–6748.

[151] K. Li, S. Wang, X. Zhang, Y. Xu, W. Xu, and Z. Tu, "Pose recognition with cascade transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 1944–1953.

[152] S. Yang, Z. Quan, M. Nie, and W. Yang, "Transpose: Keypoint localization via transformer," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 802–11 812.

[153] H. Ma, L. Chen, D. Kong, *et al.*, "Transfusion: Cross-view fusion with transformer for 3d human pose estimation," *arXiv preprint arXiv:2110.09554*, 2021.

[154] M. Al Borno, M. De Lasa, and A. Hertzmann, "Trajectory optimization for full-body movements with complex contacts," *IEEE transactions on visualization and computer graphics*, vol. 19, no. 8, pp. 1405–1414, 2012.

[155] M. Al Borno, L. Righetti, M. J. Black, S. L. Delp, E. Fiume, and J. Romero, "Robust physics-based motion retargeting with realistic body shapes," in *Computer Graphics Forum*, Wiley Online Library, vol. 37, 2018, pp. 81–92.

[156] C. D. Freeman, E. Frey, A. Raichuk, S. Girgin, I. Mordatch, and O. Bachem, "Brax–a differentiable physics engine for large scale rigid body simulation," *arXiv preprint arXiv:2106.13281*, 2021.

[157] M. Malathi and P. Sinthia, "Brain tumour segmentation using convolutional neural network with tensor flow," *Asian Pacific journal of cancer prevention: APJCP*, vol. 20, no. 7, p. 2095, 2019.

[158] R. D. Dias, A. Gupta, and S. J. Yule, "Using machine learning to assess physician competence: A systematic review," *Academic Medicine*, vol. 94, no. 3, pp. 427–439, 2019.

[159] S. Kumar, N. Ahmidi, G. Hager, P. Singhal, J. Corso, and V. Krovi, "Surgical performance assessment," *Mechanical Engineering*, vol. 137, no. 09, S7–S10, 2015.

[160] M. Andriluka, U. Iqbal, E. Insafutdinov, *et al.*, "Posetrack: A benchmark for human pose estimation and tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5167–5176.

[161] *Visipedia Annotation Toolkit*, original-date: 2017-10-02T22:21:41Z, Feb. 2023. [Online]. Available: `https://github.com/visipedia/annotation_tools` (visited on 03/22/2023).

[162] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, IEEE, vol. 2, 2006, pp. 1735–1742.

[163] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," *arXiv preprint arXiv:1801.07455*, 2018.

[164] D. Shan, J. Geng, M. Shu, and D. F. Fouhey, "Understanding human hands in contact at internet scale," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9869–9878.

[165] J. Verheul, W. Gregson, P. Lisboa, J. Vanrenterghem, and M. A. Robinson, "Whole-body biomechanical load in running-based sports: The validity of estimating ground reaction forces from segmental accelerations," *Journal of science and medicine in sport*, vol. 22, no. 6, pp. 716–722, 2019.

[166] F. E. Zajac and M. E. Gordon, "Determining muscle's force and action in multi-articular movement," *Exercise and sport sciences reviews*, vol. 17, no. 1, pp. 187–230, 1989.

[167] A. Baevski and M. Auli, "Adaptive input representations for neural language modeling," *arXiv preprint arXiv:1809.10853*, 2018.

[168] H.-S. Fang, J. Li, H. Tang, *et al.*, "Alphapose: Whole-body regional multi-person pose estimation and tracking in real-time," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[169] Z. Weng and S. Yeung, "Holistic 3d human and scene mesh estimation from single view images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 334–343.

[170] S. Tulsiani, S. Gupta, D. F. Fouhey, A. A. Efros, and J. Malik, "Factoring shape, pose, and layout from the 2d image of a 3d scene," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 302–310.

[171] M. A. Brubaker, L. Sigal, and D. J. Fleet, "Estimating contact dynamics," in *2009 IEEE 12th International Conference on Computer Vision*, IEEE, 2009, pp. 2389–2396.

[172] N. Louis, J. J. Corso, T. N. Templin, T. D. Eliason, and D. P. Nicolella, "Learning to estimate external forces of human motion in video," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 3540–3548.

[173] F. Patrona, A. Chatzitofis, D. Zarpalas, and P. Daras, "Motion analysis: Action detection, recognition and evaluation based on motion capture data," *Pattern Recognition*, vol. 76, pp. 612–622, 2018.

[174] J.-S. Monzani, P. Baerlocher, R. Boulic, and D. Thalmann, "Using an intermediate skeleton and inverse kinematics for motion retargeting," in *Computer Graphics Forum*, Wiley Online Library, vol. 19, 2000, pp. 11–19.

[175] 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.

[176] M. Habermann, W. Xu, M. Zollhoefer, G. Pons-Moll, and C. Theobalt, "Deepcap: Monocular human performance capture using weak supervision," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2020.

[177] P. S. Reitsma and N. S. Pollard, "Perceptual metrics for character animation: Sensitivity to errors in ballistic motion," in *ACM SIGGRAPH 2003 Papers*, 2003, pp. 537–542.

[178] L. Hoyet, R. McDonnell, and C. O'Sullivan, "Push it real: Perceiving causality in virtual interactions," *ACM Transactions on Graphics (TOG)*, vol. 31, no. 4, pp. 1–9, 2012.

[179] J. Tan, K. Liu, and G. Turk, "Stable proportional-derivative controllers," *IEEE Computer Graphics and Applications*, vol. 31, no. 4, pp. 34–44, 2011.

[180] N. Hansen, "The cma evolution strategy: A comparing review," *Towards a new evolutionary computation: Advances in the estimation of distribution algorithms*, pp. 75–102, 2006.

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

[182] T.-Y. Lin, M. Maire, S. Belongie, *et al.*, "Microsoft coco: Common objects in context," in *European conference on computer vision*, Springer, 2014, pp. 740–755.

[183] S. Jin, L. Xu, J. Xu, *et al.*, "Whole-body human pose estimation in the wild," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[184] P. Parmar and B. T. Morris, "What and how well you performed? a multitask learning approach to action quality assessment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 304–313.

[185] N. Louis, L. Zhou, S. J. Yule, *et al.*, "Temporally guided articulated hand pose tracking in surgical videos," *International Journal of Computer Assisted Radiology and Surgery*, pp. 1–9, 2022.

[186] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[187] D. F. Crouse, "On implementing 2d rectangular assignment algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 4, pp. 1679–1696, 2016.

[188] Q. Zhang, T. Xiao, A. A. Efros, L. Pinto, and X. Wang, "Learning cross-domain correspondence for control with dynamics cycle-consistency," *arXiv preprint arXiv:2012.09811*, 2020.