UNIVERSITÄT BONN

Context-aware Human Motion Anticipation

DISSERTATION

zur Erlangung des Doktorgrades (*Dr. rer. nat.*)

der Mathematisch-Naturwissenschaftlichen Fakultät

der Rheinischen Friedrich–Wilhelms–Universität, Bonn

vorgelegt von

JULIAN TANKE
aus
Rotenburg, Deutschland

Bonn, 2024

Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Rheinischen Friedrich–Wilhelms–Universität Bonn

Gutachter / Betreuer / Supervisor: Prof. Dr. Juergen Gall
Gutachter / Advisor: Prof. Dr. Bodo Rosenhahn
Tag der Promotion / Day of Promotion: Sept. 6th 2024
Erscheinungsjahr / Year of Publication: 2024

Abstract

by Julian Tanke

for the degree of

Doctor rerum naturalium

This thesis addresses the challenges of human motion anticipation and evaluation in complex contexts such as social interactions and human intention, focusing on motions lasting beyond one second. Central to our approach is the understanding of human motion in relation to context, categorized into historical motion data, underlying intentions, social interactions, and scene constraints. We introduce novel methodologies and a comprehensive dataset to facilitate advancements in this field. Our contributions include two generative models: Intention RNN and Social Diffusion. Intention RNN is an adversarially trained recurrent neural network which first forecasts discrete intention signals and then forecasts human motion based on the intention signal. Social Diffusion adopts a diffusion-based approach to predicting social motion dynamics - the motion of multiple humans socially interacting - and introduces a simple yet effective summarization function to model an arbitrary number of persons. Both models account for the complexity and stochasticity inherent in human motion and allow for forecasting horizons way beyond one second. Furthermore, we present a multi-person human motion dataset, "Humans in Kitchens", featuring natural interactions in kitchen environments. This dataset is instrumental in providing accurate 3D human motion tracking and annotated scene geometry, offering a rich resource for understanding complex human activities. Additionally, we propose two novel evaluation metrics: Normalized Directional Motion Similarity (NDMS) for assessing the quality of individual human motions and Symbolic Social Cues Protocol (SSCP) for evaluating social interactions. These metrics address the limitations of existing evaluation methods and provide a more nuanced understanding of motion quality and realism.

Keywords: Human Motion, Human Motion Forecasting, Social Human Motion

Acknowledgements

First and foremost, I extend my deepest gratitude to Prof. Juergen Gall, my supervisor, for his invaluable advice and support throughout the past years. His guidance and the freedom he afforded me in approaching various research topics have been instrumental to my development as a researcher.

I am also immensely grateful to Prof. Bodo Rosenhahn, who kindly agreed to serve as the external reviewer of this thesis.

Special thanks are due to my colleagues at the Computer Vision Group at the University of Bonn. I particularly appreciate Andreas Doering, Yazan Abu Fahra, Mohsen Fayyaz, Yaser Souri, Shiji Li, and Olga Zatsarinnaya for the countless insightful discussions that have enriched my research experience. I owe a particular debt of gratitude to Oh-Hun Kwon, who assisted me in the painstaking and time-consuming task of calibrating the kitchens for our dataset. His help was crucial to the success of this project.

I am thankful to Prof. Tim Landgraf for sparking my interest in Computer Vision, which set me on the path to pursuing this PhD. Similarly, I appreciate Prof. Pascal Molli and Brice Nédelec for giving me my first taste of the computer science research world, opening my eyes to the possibilities within this exciting field.

I am also immensely grateful to Christoph Lassner, Linguang Zhang, and Rene Ranfl, who as incredible industry researchers, have provided me with an abundance of guidance. Their insights from the forefront of industry research have greatly enriched my perspective and enhanced my academic work.

My heartfelt thanks go to my family and friends, who have provided understanding and support during deadlines and stressful periods. I am particularly grateful to my parents Petra Tanke & Detlef Tanke, my brother Simon Tanke, Leda Maria Machado Ladeira & Marcelo Ladeira, Mariana Machado Ladeira & Jamal Kabbaj, Debora Arieta de Oliveira & Joao Vitor Martins, and Sabaa Kiwan for their endless encouragement and love.

Last but certainly not least, I must express my deepest appreciation to Maira, my wonderful wife. She not only nudged me onto the path of this PhD journey but also was my steadfast rock during the most challenging deadlines. Her unwavering belief in me and her comforting presence made all the difference when the road seemed too tough to travel alone.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

An alphabetically sorted list of abbreviations used in the thesis:

ADE	Average Displacement Error
APD	Average Pairwise Distance
CNN	Convolutional Neural Network
FDE	Final Displacement Error
GRU	Gated Recurrent Unit
IS	Inception Score
MAE	Mean Angular Error
MOTA	Multiple Object Tracking Accuracy
MPJPE	Mean Per Joint Positional Error
MSE	Mean Squared Error
NPSS	Normalized Power Spectrum Similarity
PCP	Percentage of Correct Parts
RNN	Recurrent Neural Network
TCN	Temporal Convolutional Neural Network

List of Publications

The thesis is based on the following publications:

- **Iterative Greedy Matching for 3D Human Pose Tracking from Multiple Views**
  Julian Tanke and Juergen Gall
  DAGM German Conference on Pattern Recognition (GCPR), 2019.
  DOI:10.1007/978-3-030-33676-9_38

- **Recursive Bayesian Filtering for Multiple Human Pose Tracking from Multiple Cameras**
  Oh-Hun Kwon, Julian Tanke and Juergen Gall
  IEEE Asian Conference on Computer Vision (ACCV), 2020.
  DOI:10.1007/978-3-030-69532-3_27

- **Intention-based Long-Term Human Motion Anticipation**
  Julian Tanke, Chintan Zaveri and Juergen Gall
  IEEE International Conference on 3D Vision (3DV), 2021.
  DOI:10.1109/3DV53792.2021.00069

- **Social Diffusion: Long-term Multiple Human Motion Anticipation**
  Julian Tanke, Linguang Zhang, Amy Zhao, Chengcheng Tang, Yujun Cai, Lezi Wang, Po-Chen Wu, Juergen Gall and Cem Keskin
  IEEE International Conference on Computer Vision (ICCV), 2023.
  DOI:10.1109/ICCV51070.2023.00880

- **Humans in Kitchens: A Dataset for Multi-Person Human Motion Forecasting with Scene Context**
  Julian Tanke, Oh-Hun Kwon, Felix Mueller, Andreas Doering and Juergen Gall
  Conference on Neural Information Processing Systems (NeurIPS), 2023.
  DOI:10.5555/3666122.3666567

CHAPTER 1

Introduction

Contents

1.1 Motivation

Human motion understanding and anticipation stand central to advancing multiple domains, including human-machine interactions [140, 100, 8, 14, 106, 145, 104], robotics [2], and sociology. It plays a pivotal role in developing systems and models that can interact, respond, and predict human activities and behaviors in various contexts. A critical aspect of natural human motion is its inherent linkage to context, which can be distilled into four modes:

1. the historical motion data of an individual, which often influences their subsequent movements,

2. the intentions behind the motion, i.e. if a person wants to make a coffee or sit on a chair,

3. the social aspect, i.e. the presence and activities of other individuals within the scene, and

4. the scene itself, which puts constraints on how people can move and what objects people can interact with.

A lot of progress has been made in the simplest form of human motion anticipation, where only the past motion of a single person informs the forecast motion. Due to the inherent temporal nature of human motion, various temporal-based methods, such as recurrent neural networks (RNNs) [61, 68, 93, 132, 152, 153, 185, 213], temporal convolutional networks [28, 79, 113], transformers [7, 30, 38] and graph neural networks (GNNs) [38, 129, 130] have been used for this task. As this mode is mostly constraint by physics, i.e. the swing of an arm continuous due to inertia, these works only forecast short time horizons of up to 1 second. However, the other modes, such as intention, social [102] or scene interactions, require much longer time horizons, usually 10s of seconds. As the true intention of a human is challenging to assess, action labels [185] or characteristic poses [47] can be used as intention stand-ins. Social interactions consist of interdependent motions [168, 111], which requires modeling the relationships among all individuals. For example, in conversational turn-taking, a person's turn to talk highly depends on the start/end of the others' speaking, which might take several seconds to complete. While some attempts have been made in addressing social interaction forecasting [72, 200, 5] those methods only produce reasonable motion for up to 4 seconds, much shorter than most social interactions last. Similarly, a persons interaction with an object often lasts longer than a few seconds. For example, a person washing a plate will take at least 10-30 seconds. Due to the high degree of stochasticity of human motion generative models play a pivotal role in addressing these challenging forecasting problems and have been already successfully utilized to forecast motion sequences of single persons of up to 4 seconds [10, 212, 156, 200]. In this thesis we present two generative models, Intention RNN [185] (Chapter 6), which utilizes an adversarial training scheme to forecast very long time sequences using discrete intention signals, and Social Diffusion [186] (Chapter 7), a diffusion-based approach for social motion forecasting.

Holistically addressing the four modes of natural human motion combined with today's data-driven modelling approaches necessitates a large dataset which (a) should encompass natural interactions between multiple individuals recorded in a real environment, (b) should have annotated scene geometry to account for interactions with the scene, and (c) should include annotated per-person action labels to represent intent. These action labels can also be used to balance the evaluation and to avoid strong bias towards simple activities like standing, walking, and sitting. In this thesis we present such a dataset [187] (Chapter 8) and describe how it was obtained (Chapters 4, 5 and 8).

1.2 Problem Formulation

The problem addressed by this thesis falls into two primary categories: (a) how to anticipate human motion beyond short-term predictions of approximately one second, where motion is largely governed by inertia (e.g., the continuation of an arm's swing), and (b) how to evaluate predictions that extend into longer time horizons. Evaluation poses significant challenges due to the highly stochastic nature of human motion. Comparing model outputs to a single ground truth becomes increasingly inadequate for sequences extending beyond one second. This issue is compounded in scenarios involving multiple human motion anticipation, where a forecasting model may generate individually plausible motions that do not cohere realistically when combined. For instance, in a situation involving conversational turn-taking, an effective model should account for the likelihood of individuals waiting for their turn rather than talking over each other, even though each person talking independently might represent a valid scenario.

Human motion forecasting can be defined as $\hat{\mathbf{x}}^{t+1:T} = f(\mathbf{x}^{1:t}), t < T$ where we denote a human motion sequence of T frames of a single person as $\mathbf{x}^{1:T} \in \mathbb{A}^{T \times J}$ where $\mathbb{A}$ is usually either Euclidean space or $\mathbb{SO}(3)$ and where J is the number of articulated joints. We can extend this representation to $P > 1$ persons where $\mathbf{X}^{1:T} \in \mathbb{A}^{P \times T \times J}$. The formulation of multiple human motion forecasting then becomes $\hat{\mathbf{X}}^{t+1:T} = f(\mathbf{X}^{1:t}), t < T$. More details on how we represent human motion and how we represent generative multi-sample outputs can be found in Section 3.1. When f is a generative model, it takes an additional latent signal z which follows a well-known probability distribution. The generative formulation thus becomes $\hat{\mathbf{x}}^{t+1:T} = f(\mathbf{x}^{1:t}, z)$ for a single person and $\hat{\mathbf{X}}^{t+1:T} = f(\mathbf{X}^{1:t}, z)$ in the multiple person case. In this thesis all latent spaces are Gaussian.

For evaluating the quality of a generated single human motion sequence $\hat{\mathbf{x}}^{1:T}$ two approaches can be utilized: Directly defining a quality score on the motion $\hat{\mathbf{x}}^{1:T}$, or comparing to a ground

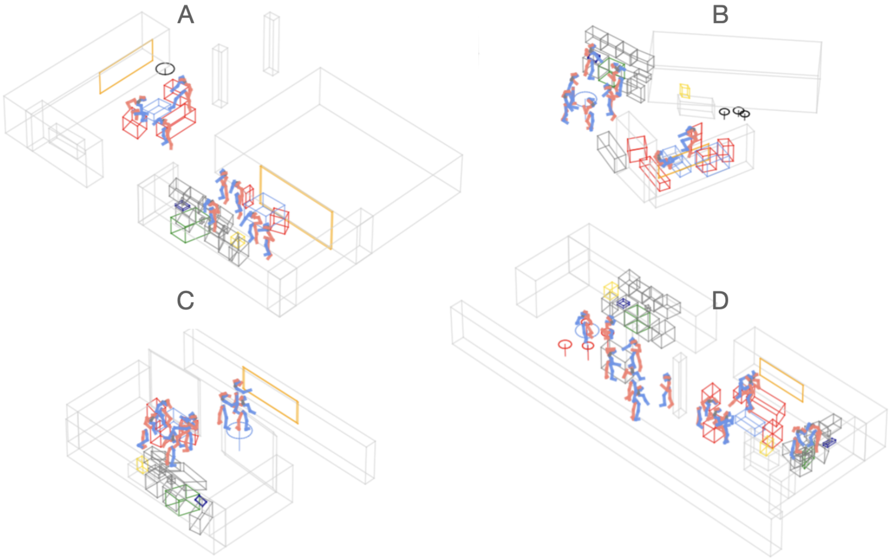


Figure 1.1: Our dataset Humans in Kitchens consists of 7.3h captured human poses of multiple persons in four different kitchen environments `A`, `B`, `C` and `D`.

truth sequence $\mathbf{x}^{1:T}$, i.e. $\mathrm{eval}(\hat{\mathbf{x}}^{1:T})$ or $\mathrm{eval}(\hat{\mathbf{x}}^{1:T}, \mathbf{x}^{1:T})$. This evaluation functions can either return a scalar value, summarizing the quality of the entire sequence for all T frames, or provide a per-frame score. The former provides a more easy to understand quality score while the latter provides more insight into a models performance. For multiple human pose sequences $\hat{\mathbf{X}}^{1:T}$ an evaluation function $\mathrm{multieval}(\hat{\mathbf{X}}^{1:T})$ must not just judge the quality of individual motion but also of the joint motion quality of all P persons in the sequence.

1.3 Contributions

In this thesis we make the following contributions towards human motion anticipation.

1.3.1 Humans in Kitchens

This thesis introduces the ambitious multi-person human motion dataset Humans in Kitchens [187], which contains multiple human motions acting naturally in four kitchen environments. Central to the dataset is accurate 3D human motion tracking [183, 109] of multiple humans from multiple calibrated cameras which will also be discussed. For the creation of the dataset it was crucial that the runtime of those methods scale well with the number of cameras and persons in the scene, as the number of continuous frames is unprecedented for a human motion datasets. Matching multiple humans across multiple calibrated cameras is NP-hard. In Chapter 4 we employ a greedy matching strategy which reduces the runtime to be quadratic in the number of persons. We build on this in Chapter 5 and reduce the runtime to be linear in the number of persons by employing an Bayesian tracking framework. Apart from 3D human motion we also annotate 3D kitchen geometry and per person per frame activity labels. A sample frame for each of our four kitchen environments can be found in Figure 1.1.

1.3.2 Intention-based Human Motion Forecasting

We introduce a novel generative sequence-to-sequence model Intention RNN [185] which takes as input a single human motion sequence and auto-regressively forecasts motion of arbitrary length. This method is adversarial trained utilizing a Gaussian latent space. A core insight of this work is that high-quality long-term human motion forecasting necessitates an additional control signal to remain sensible. In this work we utilize a discrete action representation as the control signal. This action representation is implicitly forecast given the motion sequence and no external signal has to be passed to the method. In this work we utilize a simple yet effective clustering algorithm to extract the action representation in an unsupervised manner. This method is described in detail in Chapter 6.

1.3.3 Social Human Motion Forecasting

Extending generative models to arbitrary multiple humans is not trivial: In Social Diffusion [186] we facilitate this by introducing a novel summarization function. This function must maintain two properties: it must reduce an arbitrary number of persons into a single summary signal which can be passed to all individuals, and, crucially, this function must be order-invariant, as there is no clearly defined order of humans in social settings. During training we follow the improved Diffusion model formulation by directly predicting the signal given a noised sample. We adjust the inference algorithm to explicitly facilitate forecasting by overwriting the denoised input motion signal with the input motion sequence. This method is described in detail in Chapter 7.

1.3.4 Evaluating Generative Human Motion Forecasting

Evaluating the output of generative models is an open problem and well-established metrics such as Mean Per Joint Positional Error (MPJPE) and Inception Score (IS) have well-known shortcomings, such as requiring a ground truth sequence or data distribution shifts. To address this we introduce two evaluation metrics, one to evaluate the quality of a single human motion of arbitrary lengths, and one to evaluate the quality of social interactions.

For single human motion quality evaluation, we introduce Normalized Directional Motion Similarity (NDMS) [185] which splits the motion sequence into small motion words of around $\frac{1}{3}$ seconds. Those motion words are then compared against a motion database, which is made up of the test set or a subset for more accurate action-conditioned evaluation, and then evaluation with a motion similarity score. This ensures that (a) the structure of the poses is similiar, thanks to the nearest neighbor search, and (b) that the motions are similar, due to the motion similarity score. This is discussed in detail in Chapter 6.

For social motion we introduce Symbolic Social Cues Protocol (SSCP) [186] which takes as input a motion sequence of multiple humans and predicts for each frame a discrete social signal, which has to be pre-defined for a given dataset. SSCP calculates the transition probabilities between the various states for the real test dataset and the generated motion which can then be compared using the squared Jensen Shannon distance. This is discussed in detail in Chapter 7.

1.4 Thesis Structure

The rest of this thesis is organized as follows: **Chapter 2** provides an overview of the related work while **Chapter 3** revises the preliminary concepts used in this thesis. **Chapter 4** to **Chapter 8** describe our contributions in detail. Finally, conclusions are given in **Chapter 9** along with suggested directions for future work.

CHAPTER 2

Related Work

This chapter explores the related work for the present thesis. Initially, we delve into the extraction of 3D human motion from sensor information, particularly from camera data. Subsequently, we provide a comprehensive account of how this extracted motion information can be effectively leveraged for future motion prediction, both for single individuals and scenarios involving multiple persons. Furthermore, this chapter also places significant emphasis on the discussion of datasets that play a pivotal role within the Human Motion anticipation community. These datasets serve as fundamental resources for developing robust anticipation models. In summary, this chapter systematically explores the relevant related work in the domain of 3D human motion. It begins with the extraction of 3D human motion from sensor information, proceeds to elucidate how this information can be utilized for motion prediction, and concludes by examining the datasets utilized within the Human Motion anticipation community to foster the development of robust anticipation models.

Contents

2.1 3D Human Pose Estimation

To forecast 3D human motion one first has to extract the motion from sensory information, such as video cameras. Estimating 3D human poses from cameras and videos often requires accurate 2D pose estimation and tracking, where significant progress has been made [32, 87, 143, 204, 39, 57, 71, 105, 165, 49, 50, 37] in recent years. For example, part affinity fields [32] are 2D vector fields that represent associations between body joints which form limbs. It utilizes a greedy bottom-up approach to detect 2D human poses and decouples the runtime complexity from the number of people in the image. Top-down approaches begin by identifying individuals in the image using a person detector, such as Faster R-CNN [55, 103, 148, 164], R-FCN [44, 204], Feature Pyramid Networks [39, 118], or YoloV3 [103, 163]. After detection, the region of the image encompassing the detected person is analyzed to estimate a heatmap for each joint type. These heatmaps are then integrated into a cohesive skeleton model [39, 204].

2.1.1 Monocular 3D Human Pose Estimation

Since 2D images lack depth information, detecting 3D human poses from 2D images is ill-posed [201]. Nonetheless modern algorithms are capable to recover 3D pose information. Library-based strategies use large databases to interpolate best 3D pose fits for predicted 2D poses [210, 36] but can be outperformed by neural-network-based approaches [133]. Utilizing temporal information using Recurrent Neural Networks (RNNs) [202], Euclidean Distance Matrices (EDM) [139], Graph Neural Networks (GCNs) [42, 219] or skip-connections over multiple frames [81, 153] further improves results. To obtain more plausible results, Generative Adversarial Networks (GAN) [198, 208] as well as bone length constraints [90, 153] have been used. Due to the ill-posed nature of the problem, these learning-based methods all suffer when confronted with novel poses.

2.1.2 Multi-View 3D Human Pose Estimation

Markerless motion capture extracts 3D human motion from multiple calibrated cameras in arbitrary scenes with humans in their natural clothing. While traditional motion capture setups are capable of producing higher-quality 3D motion, for example, at much higher framerates, they are constraint to studio scenes and only few people. On top of that, those recordings are slow and expensive, as markers have to be placed onto the actors - and usually require professional actors, as the clothing and the acting scene is unnatural. Markerless motion capture, on the other hand, has become increasingly relevant to extract 3D human motion, which can be utilized for downstream tasks, such as motion forecasting, or to produce novel 3D motion datasets. It can be used in-the-wild, and actors can wear and act in natural environments, making their behavior more realistic. Thus, many markerless motion capture methods [121, 122, 17, 18, 19, 54, 192, 41, 218, 216, 67] have been proposed. Multiple 3D human pose estimation from multiple views describes a matching problem where poses or joints have to be matched across views for accurate triangulation. Early works [17, 18, 20, 54] utilize 3D pictorial structure models, which are time-consuming to optimize, especially for multiple persons, due to their large state space. When many camera views are available, a voting mechanism [97] can be employed - assuming persons are visible in most camera views. Dong et al. [51] solve the correspondence problem of 2D poses per camera utilizing a top-down 2D pose estimator [39] for each view. They match 2D poses across views using geometric and appearance cues solving a convex optimization problem with cycle-consistency constrains. While this works well when persons are easy to differentiate, e.g. when full-body poses are visible, it can result in incorrect matches in more complex scenes. Cross-view matching and temporal tracking can also be formulated as a 4D association graph [218], similar to early works in 2D pose tracking [86]. Using 2D point locations as input, some approaches use triangulation directly [92] while others perform cross-view optimization before triangulation [126]. Alternatively the probability maps predicted by 2D human pose estimation approaches are used as input and projected into a probability voxel grid [192, 162] which are further improved using Voxel-to-Voxel prediction networks [138]. Other approaches learn 3D poses directly from 2D pose [191] or images [51, 207]. MetaPose [193] uses monocular epipolar pose estimates to generate 3D poses from multiple cameras individually, average the resulting poses and subsequently use a neural bundle adjustment to correct camera locations and pose prediction. TransFusion [126] is a cross-view 2D pose refinement network which is capable of improving the 2D pose predictions using multiple camera angles. Their transformer network is capable of improving other viewing angles also without knowing the camera extrinsics. ElliPose [67] builds on existing 2D human pose estimation methods and optimizes human poses across cameras and time, by fitting Ellipsoids. Crucially, the camera extrinsics do not have to be accurate and the the prediction quality can be estimated by the Ellipsoidal distortion.

2.2 3D Human Motion Forecasting

Anticipating human motion from various input signals is highly relevant for many interactive activities such as sports, manufacturing, or navigation [147] and significant progress has been made in forecasting human motion [61, 93, 132, 113, 68, 152, 65, 199, 30, 200, 5, 127, 73]. In this section, we comprehensively discuss the state of the art of the various input signals that can be utilized for human motion anticipation. These signals encompass a range of modalities, such as single human motion, focusing on methodologies and techniques designed specifically for accurately forecasting the future movements of individuals, social motion, encompassing multiple humans, and scene geometry.

2.2.1 Single Person 3D Motion Forecasting

In recent years, deep neural networks [61, 80, 28, 179, 180] have been used to synthesize and anticipate human motion. Auto-regressive methods [61, 132, 222, 179, 180] model first-order motion derivatives using the sequence-to-sequence model [181] popularized in machine translation. QuaterNet [152] replaces the exponential map representation with quaternions, which do not suffer from common 3D rotational problems such as gimbal locks. The loss is calculated in 3D Euclidean space by using forward kinematics which weights joints according to their importance in the kinematic chain. Furthermore, the authors show that the model can generate cyclic motion for very long time horizons when frame-wise user control is provided, similar to [110, 115, 120, 179]. A similar approach is utilized in Hierarchical Motion Recurrent networks [123] and Structured Prediction [6] where novel RNN structures are proposed which better represent skeletal structures. Neural State Machines (NSM) [179, 180] are a collection of feed forward neural networks which predict various motion modes which are selected by a learned gating network. During inference NSM is applied auto-regressively.

Recently, non-autoregressive methods have been shown to produce the most competitive results in short-term single person 3D motion forecasting of 1s. Predicting motion in one pass rather than auto-regressively allows methods to exploit more efficient motion representations such as Discrete Cosine Transform (DCT). [129] demonstrate that the trajectory space representation produces highly competitive results, by applying Graph-convolutional neural networks [215]. They build on this and further [130] show that applying attention [194] to better observe past motion greatly increases the methods effectiveness. Separable-Sparse Graph Convolutional Networks (SeS-GCN) [170] utilize knowledge distillation [66] to learn sparse spatial, temporal and channelwise adjacency matrices, resulting in a highly compressed model which performs 4 times faster than state of the art methods. Ma et al. [127] make a simple yet effective discovery: they find that refining an initial prediction yields superior results. They alternate between a spatial dense graph convolutional networks and temporal dense graph convolutional networks to incorporate spatial and temporal information. Last but not least, siMLPe [73] shows that complex models such as GCN, RNN, or Transformers are not required for effective short-term motion prediction: instead they combine the trajectory space representation with a light-weight MLP architecture of only 0.14M parameters, resulting in a compact representation that yields state-of-the-art results on short-term motion prediction.

Up until now, the approaches discussed have primarily focused on short-term motion forecasting, typically within a timeframe of approximately 1 second, where the influence of physical effects is highly significant. Consequently, these previous methods have predominantly employed discriminative models trained to predict a point estimate representing the most probable future motion. However, anticipating motion beyond the short-term requires addressing the inherent high degree of uncertainty associated with human motion. Recognizing this challenge, an alternative research direction has emerged, which employs generative models for long-term human multi-modal motion prediction. Generative models are well-suited for capturing uncertainty and generating diverse future trajectories as these models aim to capture the underlying distribution of possible future motion trajectories, allowing for the generation of multiple plausible outcomes instead of a single point estimate. This approach acknowledges the intrinsic variability and uncertainty in human motion, accounting for factors such as human intent, environmental interactions, and stochasticity. The utilization of generative models for long-term human motion prediction represents an orthogonal research direction compared to the previous discriminative models. By embracing uncertainty and employing generative frameworks, these methods have the potential to provide more realistic and diverse predictions, enabling a broader range of applications and addressing the challenges associated with long-term anticipation of human motion.

Early works on multi-modal human motion anticipation utilize stochastic conditional variational autoencoders (CVAE) [177, 197, 206, 29] or Generative Adversarial Networks (GAN) [16]. HP-GAN (Human Pose Generative Adversarial Network) [16] is a generative adversarial network specifically designed for modeling the probability density function of future human poses conditioned on given poses. During inference multiple samples can be generated given a random latent code. Similarly, MT-VAE (Motion Transformation Variational Autoencoder) [206] employs a Variational Autoencoder (VAE) to model the conditional distribution of data. Recently, novel sampling methods [212, 10, 11, 45] for conditional variational autoencoders were proposed for multi-modal human motion anticipation. While Mix-and-Match [10] randomly perturbs the hidden state to increase stochasticity, in their follow-up work they produce the random noise conditioned on the input poses. DLow [212] maps a random variable to a latent code, employing a two-stage approach by first learning a conditional variational autoencoder and then the mapping. By sampling multiple Gaussian latent codes, which can be thought of as belonging to different motion modes, DLow produces highly diverse human motion. This approach is further enhanced by Dang et al.[45], who generate multiple Gaussian distributions from an auxiliary space, thereby eliminating the need for a fixed-sized number of modes. The adversarial generative grammar model [156] learns stochastic production rules in conjunction with their corresponding latent non-terminal representations which can be applied to human motion prediction. By incorporating the selection of diverse production rules during the inference process, the model is capable of generating a multitude of distinct forecast outputs. However, experiments conducted in the context of human motion forecasting reveal certain limitations, specifically in generating long-term natural forecast outputs. Mao et al. [131] propose a unified deep generative network that leverages the observation that realistic human motions consist of smooth sequences of valid poses and focus on learning a pose prior, which is more tractable than learning a motion prior with limited data. The proposed generator predicts the motion of different body parts sequentially and incorporates a normalizing flow based pose prior along with a joint angle loss to ensure motion realism. Diller et al. [47] introduce the task of forecasting characteristic 3D poses, where the goal is to predict a future 3D pose of a person in a likely action-defining pose based on a short sequence observation. Unlike prior works that estimate future poses at fixed time intervals, this approach focuses on decoupling the predicted pose from time and capturing the intentional aspects of human action. Inspired by goal-directed behavior, a semantically meaningful pose prediction task is defined. To achieve characteristic pose prediction, a probabilistic approach is proposed to model the potential multi-modality in the distribution of likely characteristic poses. Autoregressive sampling is used to generate future pose hypotheses by considering the dependencies between joints. This enables the modeling of complex pose variations and allows for capturing different possible future poses.

2.2.2 Multiple Person 3D Motion Forecasting

Humans are social animals and thus a great interest exists in anticipating humans in social settings. Multiple person forecasting comes with two additional challenges when compared to single-person forecasting: (1) The number of persons in a scene might vary, so a model should be capable of addressing varying numbers of persons, and, (2) for single person forecasting the human motion can be normalized, which geometrically aligns 3D body joints into a canonical space which greatly reduces the required capacity of the model, freeing the model from having to learn global rotation and translation. Multi-Pose Extreme Motion Prediction [72] introduces a novel cross interaction attention mechanism, which leverages historical information from two persons and learns to predict cross dependencies between their pose sequences. Crucially, the method can only handle pairs of persons and normalizes the sequences according to the "leader" of the pair. The leader can be easily determined in their work as the utilized dataset contains only dancing moves. However, this approach does not scale to more persons and its normalization routine becomes more difficult when dealing with non-dancing data. Multi-Range Transformers (MRT) [200] consists of a local-range encoder to jointly capture individual motion and a global-range encoder to model social interactions. The Transformer decoder leverages both local and global-range encoder features to predict the trajectories for each person, treating the corresponding pose as a query. Thanks to the Transformer architecture, MRT can handle any number of persons but no normalization to the motion sequences is applied. Instead, the motion is represented as a combination of 3D global joints, which are not translation invariant and not rotation invariant, and velocities, which are translation invariant but not rotation invariant. This representation hinders generalization which the authors try to alleviate by utilizing an adversarial training scheme. However, the method cannot produce realistic motion for more than 3 seconds. TRiPOD [5] utilizes graph attentional networks to model human-human and human-object interactions in both the input and output spaces. Human-human interactions are learned in 3D space while objects are first detected in image space. Image features then represent the objects. Message passing is employed to fuse different levels of interactions. Additionally, the model incorporates an indicator to represent the visibility of estimated body joints at each frame, considering real-world challenges like occlusion or being outside the sensor field of view. TRiPOD can easily handle multiple persons thanks to its graph attention network. Similar to MRT it does not utilize a normalization function and instead utilizes global positions and velocities, which results in the same generaliztion problems. Dual-level generative modeling framework for Multi-person Motion Forecasting (DuMMF) [205] proposes a novel task of stochastic multi-person 3D motion forecasting and introduce a dual-level generative modeling framework. The framework separates the modeling of independent individual motion at the local level from social interactions at the global level. This dual-level mechanism is achieved within a shared generative model, utilizing learnable latent codes that represent future motion intents and switching the codes' modes of operation at different levels. Crucially, DuMMF predicts human motion locally through an articulated body model, making the motion prediction more effective while being capable of forecasting any number of persons.

2.2.3 Conditional Motion Synthesise

Although the primary focus of this thesis is on motion prediction, the methods presented in this research have strong connections to approaches that enable the generation of human motion from alternative input signals, such as text [116, 214, 188, 189], pre-defined motion categories [69, 154, 34], music [114] or 3D scenes [83].

TEMOS [155] incorporates a variational autoencoder (VAE) to learn a joint embedding of motion and natural language. However, the bottleneck structure of the VAE poses limitations on motion generation when the textual description becomes excessively detailed. The VAE's capacity to capture fine-grained nuances may be constrained, resulting in suboptimal or inaccurate motion generation. In contrast, MotionCLIP [188] takes advantage of the text-image latent space acquired by CLIP [160] to overcome the limitations imposed by data constraints. By leveraging the rich representation learned by CLIP, which enables effective cross-modal understanding between text and images, MotionCLIP expands the text-to-motion capabilities beyond the limitations imposed by the available data. Guo et al. [70] propose an auto-regressive approach which first encodes text into features and then auto-regressively synthesizes human motion conditioned on the features. MotionDiffuse [214] and Human Motion Diffusion Model (MDM) [189] utilize a transformer-based diffusion model to produce human motion from text. Indeed, the use of Transformers in the context of text-to-motion generation offers several advantages over VAEs or auto-regressive models. Transformers allow each frame in the motion sequence to have direct access to the full text description, eliminating the need for a potentially restrictive bottleneck structure. Having access to the complete text provides a richer context for motion generation and enables more accurate and detailed synthesis of human motion.

SceneDiffuser [83] leverages a Diffusion model for human pose and motion generation given a 3D scene while the inverse problem, generating a 3D scene from human motion, has been addressed in Pose2Room [144].

2.2.4 Scene Representation for 3D Human Motion

Encoding a 3D scene of arbitrary size and complexity is an open problem in human motion modelling. For this reason, TRiPOD [5] utilizes an object-centric approach by passing fixed-size image features for each object to the motion forecasting, avoiding any processing in 3D. However, this requires access to image features and appropriate detection methods and a way of handling multiple scales of objects. Evenly sampled 3D points on a sphere [179] centered at a person can be used to find and locate close-by objects via occupancy while encoding the local geometry into a fixed-size vector for neural network consumption. This has the advantage that it works on any number of objects with arbitrary shape. However, utilizing a fixed-sized grid introduces sampling artifacts and does not handle far-away but relevant objects.

A common representation of a 3D scene are point clouds where objects are represented as unordered sets of infinitesimal 3D points that usually represent the objects surface. PointNet [159] is designed to directly process point cloud data while preserving the permutation invariance of the input points. Unlike existing approaches that transform point clouds into voluminous voxel grids or collections of images, PointNet avoids unnecessary data expansion. The network offers a unified architecture applicable to various computer vision tasks, such as object classification, part segmentation, and scene semantic parsing. Despite its simplicity, PointNet demonstrates high efficiency and effectiveness, outperforming or achieving comparable results to state-of-the-art methods. Basis point sets (BPS) [158] are a simple yet effective representation of 3D point clouds of arbitrary size. They generate a fixed, ordered number of random basis points and produce feature vectors that represent the distance of the basis points to the closest point in the point cloud. By using BPS as input for a simple fully connected network, the authors achieve performance comparable to PointNet for shape classification while reducing floating-point operations by three orders of magnitude.

2.2.5 Human Motion Evaluation

Evaluating complex multi-variate time series with a high degree of stochasticity, such as human motion, remains a challenging research problem. The simplest approaches calculate the Euclidean distance [80, 93, 132] to a target sequence independently for each time step, which works well for very short time horizons ($< 0.5s$). However, frame-wise distances completely ignore motion dynamics and forecasting only the last pose results in competitive results [132]. To address these challenges, frequency-based metrics have been proposed. Frequency-based methods such as NPSS [65] incorporate motion information, but they accumulate it over the entire sequence. On top of that, distances in the frequency domain are difficult to interpret and make it hard to pinpoint when a motion can still be considered as realistic or not. In [10] the inception score [84] is adapted by training a model on skeleton data to evaluate the quality of the generated sequences. Complementary, a binary classifier is trained for quality assessment. However, both models are not publicly available, making comparisons difficult. While DLow [212] uses the average pairwise distance to measure diversity as [10], it only evaluates the best generated sequence using the quality metrics from [80, 93, 132]. We provide a detailed overview of these evaluation methods in Chapter 3.4.

2.3 Datasets for Multi-Person Motion

We present the most related datasets with one or multiple 3D human poses in Tab. 5.1 and briefly discuss them.

2.3.1 Single-Person Datasets

AMASS [128] unifies several 3D human motion datasets using SMPL [124]. In total, AMASS contains over 40 hours of motion capture recordings. While the underlying articulated model ensures high quality motion, AMASS only contains recordings of a single person and thus no human

Dataset	real data	real setting	SMPL	$\max(\#P)$	total time	activities	scene	framerate
AIST++ [114]	yes	no	yes	1	5.2h	no	no	60Hz
AMASS [128]	yes	no	yes	1	40h	no	no	60Hz
BEHAVE [23]	yes	no	yes	1	8.5min	no	yes	10Hz
CHAIR [95]	yes	no	yes	1	17.3h	no	yes	30Hz
CHICO [170]	yes	yes	no	1	3.77h	no	yes	25Hz
GIMO [221]	yes	yes	yes	1	1.2h	no	yes	30Hz
GTA-IM [33]	no	no	no	1	9.2h	no	yes	30Hz
Human3.6M [85]	yes	no	no	1	2.93h	no	no	50Hz
Humanise [203]	no	no	yes	1	5.55h	yes (language)	yes	60Hz
MoGaze [108]	yes	no	no	1	3h	no	yes	120Hz
PROX [76]	yes	yes	yes	1	55min	no	yes	30Hz
SAMP [77]	partial	no	yes	1	100min	no	yes	30Hz
3DPW [195]	yes	yes	yes	2	14min	no	no	60Hz
CHI3D [58]	yes	yes	no	2	40min	no	no	200Hz
CMU Mocap [1]	yes	no	no	2	9.75h	no	no	60Hz / 120Hz
CMU Panoptic [98]	yes	no	no	8	5.5h	no	no	29.97Hz
EgoBody [217]	yes	yes	yes	2	2h	no	yes	30Hz
ExPI [72]	yes	no	no	2	20min	no	no	25Hz
Haggling dataset [99]	yes	no	no	3	3h	yes (1)	no	29.97Hz
MuPoTS-3D [136]	yes	yes	no	3	$\leqslant$4.4min	no	no	30Hz / 60Hz
NTU-RGB+D 120 [119]	yes	no	no	2	63min	yes (120)	no	30Hz
RICH [82]	yes	yes	yes	2	2.7h	no	yes	60Hz

Table 2.1: Comparison of various datasets with 3D human poses. The *real data* column specifies if a dataset contains real, synthetic, or partially synthetic human motion. The *real setting* column specifies if the recording was done in a controlled studio environment or in a real-world scene. The *SMPL* column determines whether a dataset provides SMPL [124] poses. The column $\max(\#P)$ defines the maximal number of persons at the same time in a scene while columns *activities* and *scene* determine if the dataset contains per-frame annotations of activities or scene geometry. Numbers in *activities* indicate the number of different per-frame activities.

interactions. It also does not contain scene context information or per-frame action annotations. BEHAVE [23] presents fine-grained human-object interactions but is recorded at only 10Hz and very small (8.5 minutes). Another human-object interaction dataset is CHICO [170], which contains 3.77 hours of recording. In contrast to BEHAVE, the human poses are represented by 3D keypoints instead of SMPL [124] body poses. Human3.6M [85] contains around 900k high-quality human 3D poses. The motion sequences, however, are unrealistic since the actors pantomime activities without object except of sitting on a chair. Other objects or 3D geometry are missing. MoGaze [108] contains 3 hours of human-object interactions with annotated 3D geometry. PROX [76] provides very high-quality human-object interactions with static objects such as chairs, beds and sofas and utilizes SMPL as body model. The GTA Indoor Motion dataset (GTA-IM) [33] utilizes the GTA engine to produce human-scene interactions in indoor environments for human motion forecasting. The motion in this dataset, however, is synthesized and looks unrealistic and clumsy. SAMP [77] is a human-scene interaction dataset containing 7 real objects, such as sofas and armchairs, and various human interactions with those. An extensive augmentation pipeline is used to extend the dataset with a greater variety of human-object interactions. AIST++ [114] contains 30 subjects dancing to music sequences. GIMO [221] contains a single person interacting with a static, high-quality mesh, where data is provided in the form of an egocentric viewpoint. The dataset was explicitly designed for motion forecasting, where a person walks into a scene with the intent to interact with an object. Humanise [203] is a large-scale synthetic human-object interaction dataset that leverages existing datasets in human motion (AMASS) and 3D indoor scenes [43]. Similar to GIMO, the scenes are static. Furthermore, the person-object interactions are synthetic and do not include real person-object interactions. CHAIR [95] is a large-scale human-chair interaction dataset that contains a large variation of chairs.

2.3.2 Multi-Person Datasets

3DPW [195] contains 2 persons per scene, but the 3D joint locations were obtained from moving cameras, resulting in unrealistic sliding. In total, this dataset consists of 14 minutes of recording. CMU-Mocap [1] is a high-quality motion capture dataset of 1 to 2 persons. While some of the single-person sequences contain scene interaction, the scenes are not annotated. The dataset does not contain accurate action labels, but the sequences can be searched based on the video descriptions. CHI3D [58] consists of 40 minutes of two-person interactions, including close interactions such as touching. Human poses are represented as 3D skeletons and as SMPL-X models [151]. CMU Panoptic Studio [96, 98] is a large-scale 3D human motion dataset featuring 1 to 8 persons in various scenes. The human poses are represented using 3D COCO keypoints. The dataset size is 5.5h, but lacks labeled scene geometry and per person action annotations. Due to the studio recording, the range of motion is limited. MuPoTS-3D [136] is a test set for 3D human pose estimation. It contains up to 3 persons in a scene but it contains only 4.4 minutes of recording. NTU-RGB+D [172, 119] is an action recognition dataset containing more than 1 hour of recording of one or two persons. Human bodies are represented as 3D skeletons and per-frame activities are annotated. EgoBody [217] contains sequences of pairwise social interactions with real scene geometry. In contrast to our work, the scene geometry is only statically annotated and social interactions are less natural, as one actor wears a virtual headset. RICH [82] captures human-object interactions by defining pseudo-contact labels on the body mesh. Unlike PROX [76], RICH contains mostly outdoor scenes of around $60m^2$ and provides more accurate SMPL-X estimates. Some of the scenes contain two humans who interact with each other from a distance like throwing a ball.

CHAPTER 3

Background

In this chapter we formally define how human poses are represented for the tasks of 3D multiple human pose estimation and 3D human motion anticipation. We establish the necessary background in multiple-view geometry and in 3D human pose representation. As the task of 3D human motion anticipation requires handling sequential data, we briefly introduce neural network architectures specifically designed for sequence processing. This knowledge will form the foundation for our subsequent exploration of 3D human motion anticipation techniques. To address the challenges and complexities of the tasks at hand, we leverage both discriminative and generative models. We delve into these models, exploring their respective strengths, weaknesses, and applications within the context of 3D human motion anticipation. Lastly, we thoroughly discuss the evaluation metrics employed for assessing the performance and accuracy of human motion prediction methods.

Contents

3.1 Human Pose Representation

This thesis covers the topics of 3D human motion prediction as well as anticipation 3D human motion and both tasks require a way to represent a 3D human pose. Human poses can be represented in various ways, each having its own strengths and weaknesses. Since learning algorithms are generally independent of the pose space, we will denote a single human 3D pose with J joints as $x \in \mathbb{A}^J$, where $\mathbb{A}$ is an arbitrary space and usually either means 3D Euclidean space or 3D rotational space $\mathbb{SO}(3)$. When objects in this thesis are embedded in Euclidean space, we assume that the z-axis points upwards. As this thesis covers the topic of human motion we address a sequence of T human poses as

$$\mathbf{x}^{1:T} = (x_1, \ldots, x_T) \in \mathbb{A}^{T \times J} \tag{3.1}$$

where x_t is a single human pose at frame t. If a sequence consists of $P > 1$ persons we denote it as

$$\mathbf{X}^{1:T} = \left\{\mathbf{x}_p^{1:T}\right\}_p^P \in \mathbb{A}^{P \times T \times J} \tag{3.2}$$

where $\mathbf{x}_p^{1:T}$ represents human motion of length T for person p. In some of our works, we model the forecasting of human motion by generating multiple samples from an estimated distribution, thereby approximating the underlying probabilistic nature of human motion. Specifically, our model generates S samples for the same input motion, representing diverse potential outcomes as:

$$\mathcal{X}^{1:T} = \left\{\mathbf{x}_s^{1:T}\right\}_s^S \in \mathbb{A}^{S \times T \times J} \tag{3.3}$$

where $\mathbf{x}s^{1:T}$ represents the motion for sample s. For motion distributions involving multiple persons, we define the representation as:

$$\mathfrak{X}^{1:T} = \left\{ \left\{\mathbf{x}_{p,s}^{1:T}\right\}_s^S \right\}_p^P \in \mathbb{A}^{P \times S \times T \times J} \tag{3.4}$$

where $\mathbf{x}_{p,s}^{1:T}$ is the s-th sample motion sequence of person p. This section will outline the most commonly used 3D human pose representations.

3.1.1 3D Joint Positions

The most straightforward representation of 3D human pose, denoted as x, is an ordered list comprising J **3D joint locations**. Formally, we can express it as $x \in \mathbb{R}^{J \times 3}$, where each joint is represented by a 3D coordinate. This representation is appealing due to its simplicity and ease of acquisition through triangulation. One advantage of using 3D human joint representations in the context of machine learning is that they are embedded in Euclidean space. This Euclidean embedding aligns well with common loss functions, such as Mean Squared Error, which measure the discrepancy between predicted and ground truth joint positions. Moreover, this representation has been widely adopted in various 3D human pose datasets, such as Panoptic Studio [97] and related datasets such as Haggling [99] as well as in Kinect-based datasets such as NTU-RGB+D 120 [172]. While this representation offers simplicity and ease of use, it disregards the inherent interdependencies and constraints between joints, which can result in unrealistic body poses. The independent nature of joint locations in 3D pose representation can lead to situations where joints, such as the elbow and shoulder, may appear too close or too far from each other. These discrepancies can arise due to various factors, including occlusions or errors in the triangulation process, ultimately resulting in incorrect pose estimations. This issue is not only relevant to 3D pose estimation but also carries implications for tasks like motion forecasting. Small errors in joint locations can accumulate and lead to estimated poses that deviate from the valid pose manifold, generating poses with incorrect limb lengths or unrealistic body configurations.

Pose Normalization Poses represented as 3D joint positions are in global coordinates and can be factored into a canonical representation and a global rigid transformation where poses in canonical representation are translated and rotated so that the defined root joint(s) are located and oriented consistently. This is helpful for both learning and evaluation as global rotation and translation can be disregarded. For normalization, we select as root the hip of the person and translate and rotate the pose in such a way that the center of the hip lies at the origin and rotate along the z axis such that the hip aligns with the y axis. We rotate only around the z axis as it is the only axis which is invariant to gravitational forces. Rotating around either x or y axis would cause the 3D pose to be off-balance, removing potentially important geometric and physical cues. More formally, we apply the following transformation:

$$\text{norm}(x) = R_{\text{norm}}(x - x_{\text{root}}) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} (x - x_{\text{root}}) \tag{3.5}$$

where $x_{\text{root}} \in \mathbb{R}^3$ is the 3D root location, the center point between the left and right hip joint and where

$$\theta = \text{atan2}(\mathrm{y}_{\text{righthip}} - \mathrm{y}_{\text{lefthip}}, \mathrm{x}_{\text{righthip}} - \mathrm{x}_{\text{lefthip}}) \tag{3.6}$$

where $\mathrm{x}_{\text{righthip}}$ / $\mathrm{y}_{\text{righthip}}$ and $\mathrm{x}_{\text{lefthip}}$ / $\mathrm{y}_{\text{lefthip}}$ are the x / y coordinates of the left and right hip, respectively. Note that the normalization can be undone trivially

$$x = R_{\text{norm}}^T \text{norm}(x) + x_{\text{root}} \tag{3.7}$$

to recover the pose in global coordinates.

In this work we often normalize a sequence of poses $\mathbf{x}_{1:T}$ at a given frame t where the factored global rotation R_{norm} and x_{root} are extracted from the pose at frame t and then applied to the entire sequence:

$$\text{norm}(\mathbf{x}_{1:T}, t) = R_{\text{norm},t}(\mathbf{x}_{1:T} - x_{\text{root},t}) \tag{3.8}$$

where $R_{\text{norm},t} \in \mathbb{SO}(3)$ and $x_{\text{root},t} \in \mathbb{R}^3$ are rotation and translation from global to canonical space for the t-th element in $\mathbf{x}_{1:T}$, x_t.

3.1.2 Pictorial Structure Model

The **Pictorial Structure Model** [59, 56, 13] represents the human body as a configuration $x = \{l_1, \ldots, l_J\}$ consisting of J rigid parts. Each part is characterized by its location, given by $x_i = (\mathrm{x}_i, \mathrm{y}_i, \theta_i)$, where $(\mathrm{x}_i, \mathrm{y}_i)$ represents the image position of the part, and θ_i represents its absolute orientation. Formulating the model as a conditional random field (CRF), we assume that the probability of the part configuration x given the image evidence can be factorized into a product of unary and pairwise terms. This factorization allows for tractable inference in practice. The assumption underlying the factorization is that the likelihood of the part configuration can be decomposed into the product of individual part likelihoods. This implies that the probability of each part's location is influenced by the image evidence independently, as well as by pairwise relationships with other parts in the configuration. By decomposing the part configuration likelihood in this way, the inference process becomes more manageable. It enables efficient estimation of the optimal part configuration given the image evidence by leveraging the unary and pairwise terms in the CRF. Multi-View Pictorial Structures [12] lift this representation from 2D to 3D by additionally introducing pairwise factors between corresponding parts in each camera. The 3D joint locations can be obtained by solving the pictorial structure model in all views and then triangulating each joint.

3.1.3 Articulated Body Models

Articulated body models [176, 175, 25, 46, 64, 124] are made up of pose parameters $x \in \mathbb{SO}(3)^J$, skeletal structure $\mathbb{S}$, shape parameter β and global translation $\mathrm{t}_{\text{global}} \in \mathbb{R}^3$. Pose parameters $x = (R_1, R_2, \ldots, R_J)$ represent an ordered list of J angular joint parameters, where each joint, such as an elbow, defines the rotation of that specific joint. Unlike the 3D joint position representation, this representation is local in nature. For example, when the elbow is fixed in rotation and the shoulder is moved, the angular values at the elbow remain the same, as only the angles of the shoulder have to change. To obtain 3D joint locations forward kinematics has to be applied to each joint, starting at a common root joint. Forward kinematics recursively walks up the kinematic chain of length n, starting at the root transformation $T_{0,1}$, and applies rigid transformation

$$[T] = T_{0,n} = \prod_{i=1}^{n} T_{i-1,i} \tag{3.9}$$

where transformation $T_{i-1,i}$, specifying the rigid transformation from chain element $i-1$ to i, is defined as:

$$T_{i-1,i} = \begin{bmatrix} & R_i & & \mathrm{t}_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.10}$$

where R_i is the joint rotation defined as i-th element in the ordered pose parameters x and where t_i a vector defining the bone attached to the i-th joint, where the length of the vector defines the bone length and the direction the bone orientation. Crucially, the bone orientation and length does not change for a person, meaning that t_i is fixed for each joint per person, and is thus part of the shape parameter β. Note that the rotation at the root joint expresses the rotation of the entire pose x in global space.

Importantly, the human body is structured as a kinematic tree. This tree begins at a single root joint, typically located at the pelvis, and branches out to various end-effectors such as the head, hands, and feet. This organization allows for an integrated movement across the skeletal structure, where movements in one part of the body can influence movements in another. The skeletal structure comprises these branches, each representing a pathway from the root to an extremity. The relationships and connections between joints are essential for defining the motion dynamics of the body, allowing us to model human motion more comprehensively. Various skeletal structure sets have been defined with the most common ones being the proprietary format of Vicon, used for example in simplified form in Human3.6M [85], Biovision Hierarchy, used for example in CMU Mocap [1], and Skinned Multi-Person Linear Model (SMPL) [124].

As alluded before, the shape parameter β is fixed for each person and does not change across time. The shape parameter, among others, contains the bones for each person, which define their physique. Depending on the type of articulated pose representation, β might contain more parameters, such as blend shapes, which help with producing realistic-looking 3D body meshes. However, we omit those details as this thesis solely focuses on 3D skeleton representations and not 3D body meshes.

Articulated 3D human poses have several advantages over 3D joint location representations: as each joint is embedded in a local frame, pose normalization is trivial, as the parameters are independent of global rotation and translation. Furthermore, the degree of freedom of the pose reduces as well, as many joints, such as elbows and knees, are hinge joints with only one degree of freedom. Furthermore, the skeletal structure of the body is fixed, ensuring valid bone lengths for any pose configuration. However, obtaining pose parameters x from sensors is difficult as it requires inverse kinematics, which requires numerical approaches for non-trivial kinematic chains found in human skeletons. Furthermore, rotational representations pose some challenges for usage in machine learning applications, due to discontinuities (Euler angles, axis-angles) and non-uniqueness (Quaternions, Euler angles, axis-angles).

3.2 Geometry

This section serves as the foundation for 3D pose estimation using multiple calibrated cameras, which is relevant for Chapters 4, 5 and 8. The field of modeling geometry from multiple camera views has been extensively studied and understood [75]. This chapter briefly introduces the pinhole camera model as well as epipolar geometry. The notation in this chapter is self-contained and thus might overload some previously used notations and variables. Most notably, we denote points in 3D world coordinates with capital letters and points in 2D image space with lowercase letter. Outside of this chapter we will explicitly indicate when making use of this chapters notation. To facilitate a better understanding of camera geometry, we will begin by introducing some fundamental concepts. In computer vision, the projective space is commonly used as a convenient representation for the 3D world. The projective space $\mathbb{P}^n$ extends the Euclidean space $\mathbb{R}^n$ by

$$\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \in \mathbb{R}^n \sim \begin{pmatrix} kx_1 \\ \vdots \\ kx_n \\ k \end{pmatrix} \in \mathbb{P}^n \tag{3.11}$$

where points in projective space form an equivalence class for all $k \in \mathbb{R}\backslash\{0\}$. We call points in this space *projective* or *homogeneous*. In projective space, two lines always intersect at a single point. This property holds true even for parallel lines, which meet at a point at infinity represented as $\big(x_1, \ldots, x_n, 0\big)^T$. In $\mathbb{P}^2$, all points at infinity of parallel lines form a line at infinity while in $\mathbb{P}^3$, they form a plane at infinity denoted as Π_∞. Affine transformations allow to translate, rotate, or scale objects linearly while keeping points at infinity unchanged. Projective geometry is homogeneous, treating all points equally, including points at infinity, which can be mapped to points on the projective plane using projective transformations.

3.2.1 Camera Models

A camera maps points from the 3D world onto points on a 2D image and can be approximated as a projective camera P which uses similar triangles to project points:

$$x = PX = K\big[R \mid \mathbf{t}\big]X \tag{3.12}$$

where $X \in \mathbb{P}^3$ represents a point in the 3D world in homogeneous coordinates, where $x \in \mathbb{P}^2$ is X projected onto the image plane, and where $R \in \mathbb{SO}(3)$ and $\mathbf{t} \in \mathbb{R}^3$ represent the cameras extrinsic parameters and where K represents the cameras intrinsic parameters:

$$K = \begin{bmatrix} f \cdot m_x & s & x_0 \\ 0 & f \cdot m_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.13}$$

where s defines the shear distortion, x_0 and y_0 are the principle point offsets in pixels, f is the focal length and where m_x and m_y are scaling factors to scale from world to pixel units. We can obtain the camera center C

$$C = -R^T\mathbf{t} \tag{3.14}$$

in world coordinates. We also call the 3×4 matrix P camera matrix.

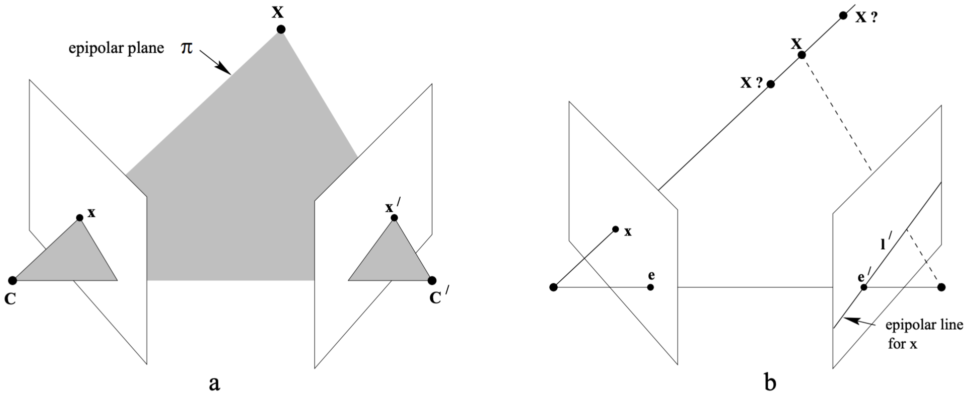


Figure 3.1: Point correspondence: (a) two cameras with their respective centres C, C' look at a point X in world coordinates which is projected to x and x' at the respective image planes. The camera centres and the 3D point form an epipolar plane Π. (b) A point x in one image projects as epiploar line l' in the other. The epipoles e, e' are points on the image that represent the other camera centers. This figure is adopted from [75].

3.2.2 Epipolar Geometry

Depth information is lost when projecting 3D world points onto a 2D image using the camera matrix. To recover depth, stereo vision is utilized, involving a second camera that observes the same 3D point, allowing for triangulation in 3D world coordinates. An overview is presented in Figure 3.1. Given two cameras with centers C and C' and a 3D point X in world coordinates, which is projected onto the respective image planes as x and x', the three points, X, C, and C', form a plane Π, referred to as the epipolar plane. The images of the other camera's center in each camera are the epipoles e and e'. In stereo correspondence, a significant challenge detailed in Figure 3.1 (b), each point such as x from one camera defines an epipolar line l' in the other camera's image plane. This line passes through the epipole e' and is the locus of all possible images of point x as seen by the second camera. To find the correspondence for x' on l', methods such as feature detection or cross-correlation are commonly employed around the predicted area of x. This linear relationship between x, x', and l' can be expressed as:

$$x'^T F x = x'^T l' = 0 \tag{3.15}$$

where F is the Fundamental matrix.

3.3 Forecasting with Neural Networks

In this section we briefly describe the fundamentals of supervised machine learning as well as of how to model sequences with neural networks. In our review of supervised machine learning we discuss both discriminative as well as generative models. This chapter's notation is based on [157]. We utilize Recurrent Neural Networks in Chapters 5 and 6, Temporal Convolutional Neural Networks and Attention in Chapter 7 and Transformers in Chapter 8. Many of our works use both Discriminative and Generative models. Discriminative models are used in Chapters 4, 5, 7 and 8 while Generative models are used in Chapters 5, 6, 7 and 8.

3.3.1 Supervised Machine Learning

In supervised machine learning we try to find a function, or model, that returns the posterior probability distribution $\mathrm{Pr}(\mathbf{y}|\mathbf{x})$ which infers the true world state $\mathbf{y}$ given a set of observations $\mathbf{x}$. Given the model parameters θ we parameterize the posterior as $\mathrm{Pr}(\mathbf{y}|\mathbf{x}, \theta)$. If $\mathbf{y}$ is continuous we call the process a regression, and if $\mathbf{y}$ is discrete we call it a classification. Inferring $\mathbf{y}$ from $\mathbf{x}$ is ambiguous in many cases. For example, imagine observing a single frame of a 3d human motion sequence where the arm of the human is mid-swing: without further information the arm might either swing forward, swing backward or stay still resulting in ambiguity when forecasting the next frame of the motion. Similarly, imagine we want to anticipate the future motion of an observed human motion sequence several tens of seconds into the future: due to the very high degree of stochasticity in human motion there is an infinite number of potential futures, i.e. a person may walk left, right or straight ahead, and, given the context, certain motions might be more or less likely. For complex distributions, such as human motion, computing the posterior is not tractable and instead we choose the mode of the posterior distribution, the maximum a posteriori solution (MAP) $\hat{\mathbf{y}}$, or, alternatively, we approximate the full distribution by drawing random samples from the posterior distribution. For supervised machine learning three components are essential: A model, parameterized by θ, that maps observations $\mathbf{x}$ to output states $\mathbf{y}$, a training algorithm that fits parameters θ using m known paired training samples $\{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^m$, and an inference algorithm.

We distinguish between two model types, **discriminative** and **generative** models: Discriminative models model $\mathrm{Pr}(\mathbf{y}|\mathbf{x})$ directly while generative models model $\mathrm{Pr}(\mathbf{x}, \mathbf{y}) = \mathrm{Pr}(\mathbf{x}|\mathbf{y}) \cdot \mathrm{Pr}(\mathbf{y})$. Generative models model how the observation pairs $\{\mathbf{x}, \mathbf{y}\}$ are actually generated while discriminative models make no such assumption. This allows for drawing samples from generative models that approximate the full distribution. Through Bayes' Theorem generative models have discriminative properties while the reverse is not true.

A training algorithm fits parameters θ using m known paired training samples $\{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^m$. We can fit the posterior distribution of discriminative models directly using Maximum Likelihood Estimation (MLE):

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} \left[\prod_{i=1}^{m} \Pr(\mathbf{y}_i|, \mathbf{x}_i, \theta) \right]. \tag{3.16}$$

Given a new observation $\mathbf{x}$, an inference algorithm either returns posterior distribution $\Pr(\mathbf{y}|\mathbf{x}, \theta)$ or the MAP solution. In the case of discriminative models, we can directly evaluate on the posterior distribution, while in the case of generative models, new samples can be drawn from the posterior to approximate the true data distribution.

Discriminative models are used in Chapters 4, 5, 7 and 8 while Generative models are used in Chapters 5, 6, 7 and 8.

3.3.2 Recurrent Neural Networks

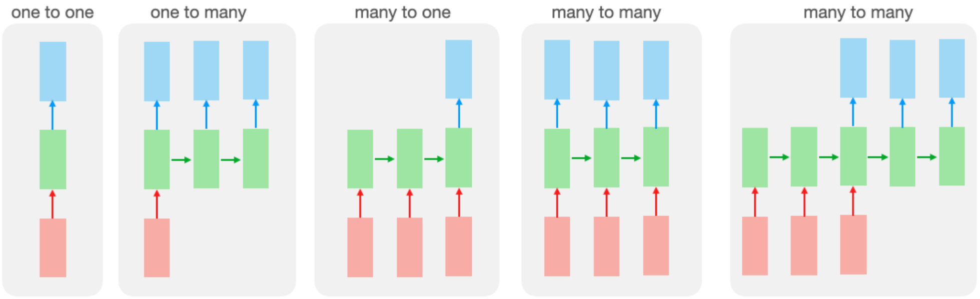


Figure 3.2: Recurrent Neural Networks (RNNs) allow for a flexible input-output mapping on sequential data, such as one-to-one, one-to-many, many-to-one, many-to-many and offset many-to-many. The red boxes represent input signals while the blue boxes represent the network outputs. Hidden states are represented as green boxes.

A Recurrent neural network (RNN) [134] allows to process sequential data by recursively applying a deep neural network. RNNs process inputs x_t in sequence one at a time and keep a hidden state h_t which recursively depends on the previous inputs $\mathbf{x}_{1:t-1}$ in the sequence:

$$y_t, h_t = \mathrm{rnn}(x_t, h_{t-1}) \tag{3.17}$$

where the first hidden state h_1 usually defaults to the zero vector. Crucially, for backpropagation, the recursion is unrolled at a given time step. RNNs allow for various complex and flexible input-output variants which we briefly overview in Figure 3.2. In their simplest form, RNNs can be used as feed forward neural networks (one-to-one) but they can also be used to produce an output sequence given a single input (one-to-many), they can be used to summarize a sequence (many-to-one), they can be used to process a sequence per frame (many-to-many) or they can be used as sequence-to-sequence (seq2seq, offset many-to-many) model. A commonly used variant of RNNs is the Gated Recurrent Unit [40] (GRU) which utilizes a gating mechanism for information flow control. Given input sequence $(x_1, \dots, x_T)$ and initial hidden state $h_0 = 0$

$$\begin{aligned} z_t &= \sigma(W_z x_t + U_z h_{t-1} + b_z) \\ r_t &= \sigma(W_r x_t + U_r h_{t-1} + b_r) \\ \hat{h}_t &= \tanh(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h) \\ y_t = h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot \hat{h}_t \end{aligned} \tag{3.18}$$

where W, U are parameter matrices and where b are parameter biases. In GRUs the output signal equals the output hidden state. In Chapter 5 we utilize a many-to-one GRU for the prediction part of the recursive Bayesian filtering. In Chapter 6 we use two many-to-many GRUs to first forecast a human intention and then their 3D motion. Last but not least, in Chapter 7 we use a many-to-one GRU to predict a persons role in a social scene and a many-to-many GRU to predict if a person is talking.

3.3.3 Temporal Convolutional Neural Networks

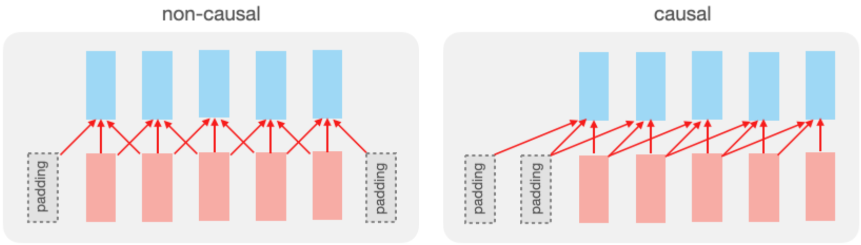


Figure 3.3: Temporal Convolutional Neural Networks (TCNs) apply a fixed-size convolution to a sequential input. Padding can be utilized to maintain the number of frames in the output sequence (left) or to causally constrain the output signal (right). Red boxes represent the input sequence, gray boxes represent padding, and blue boxes represent the output.

A Temporal Convolutional neural network (TCN) [146, 15] processes sequential data by applying a usually fixed-sized learned convolutional kernel to an input signal. A TCN takes as input a sequence $(x_1, \cdots, x_T)$ and produces an output sequence $(y_1, \cdots, y_T)$

$$y_1, \cdots, y_T = \text{tcn}(x_1, \cdots, x_T) \tag{3.19}$$

where tcn is made up of stacked layers of temporal convolutions. Given an element s of a sequence x the convolutional operation F is defined as:

$$F(s) = (x *_d f)(s) = \sum_{i=0}^{k-1} f(i) \cdot x_{s-d \cdot i} \tag{3.20}$$

with filter $f : \{0, \cdots, k-1\} \rightarrow \mathbb{R}$, filter size k and dilation factor d. Using dilation enables the network to look back at history with steps exponential in the network depth while no dilation results in just a linear look-back size. In contrast to RNNs, TCNs do not require the temporal sequence to be unrolled, allowing for parallel computation of the outputs, which greatly improves training and inference runtime performance. Furthermore, TCNs often demonstrate more stable gradient flow through layers, which counteracts common issues in RNNs such as vanishing or exploding gradients. Last but not least, TCNs do not require to carry a hidden state over time, allowing for more memory-efficient implementations. A graphical overview is presented in Figure 3.3.

3.3.4 Transformers

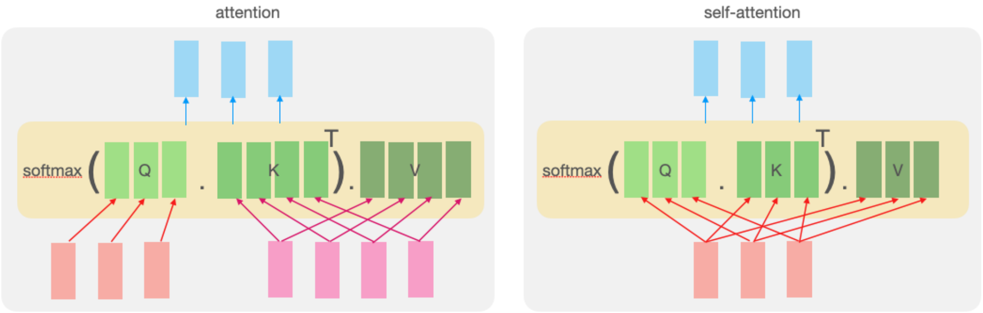


Figure 3.4: Overview of the dot-product cross-attention (left) and self-attention mechanism. In cross-attention (left), two sequences, a query sequence (red boxes) and context sequence (rose boxes), are processed. This two sequences do not have to have the same length and the output sequence (blue boxes) length depends on the query sequence. In self-attention (right), query and context sequence are the same. Note that Q, K, and V are linear transformations (green boxes) of the respective sequences.

Transformers [194] introduce the attention mechanism to process input sequences in parallel over very large time horizons. Unlike RNNs, Transformers process all sequence positions in parallel, allowing for more efficient computing and more model scalability. Unlike TCNs, Transformers are not limited by a fixed-size receptive field, allowing them to attend to all positions in the input sequence, regardless of distance. In detail, attention is a mapping of a query and a set of key-value pairs to an output. Attention outputs the weighted sum of the values, where the weights are defined by a *compatibility* function of the query and key. This form of attention mechanism is also called *Scaled Dot-Product Attention*:

$$\mathrm{attn}(Q, K, V) = \mathrm{softmax}(\frac{QK^T}{\sqrt{d_k}})V \tag{3.21}$$

where d_k is the dimension of the key vector and where Q, K, and V are linear transformations of the input sequence(s). Scaling of the dot product by $\sqrt{d_k}$ counteracts large magnitudes in the dot product, when d_k is large. This is harmful to the optimization as it pushes the softmax function into regions with very small gradients. We present an overview for attention (cross-attention) and self-attention in Figure 3.4. In cross-attention two sequences, a query sequence and context sequence, are processed. This two sequences do not have to have the same length and the output sequence length depends on the query sequence alone. In self-attention, query and context sequence are the same.

In practice, multi-head attention, the concatenation of multiple learned attention functions, yields superior results as it allows the model to attend to multiple elements in the sequence. Multi-head attention is defined as follows:

$$\begin{aligned}\text{MultiHead}(Q, K, V) &= \text{Concat}\big(\text{head}_1, \cdots, \text{head}_h\big)W^O \\ \text{where head}_i &= \text{attn}(QW_i^Q, KW_i^K, VW_i^V)\end{aligned} \tag{3.22}$$

and where W^O, W_i^Q, W_i^K and W_i^V are learned projected matrices.

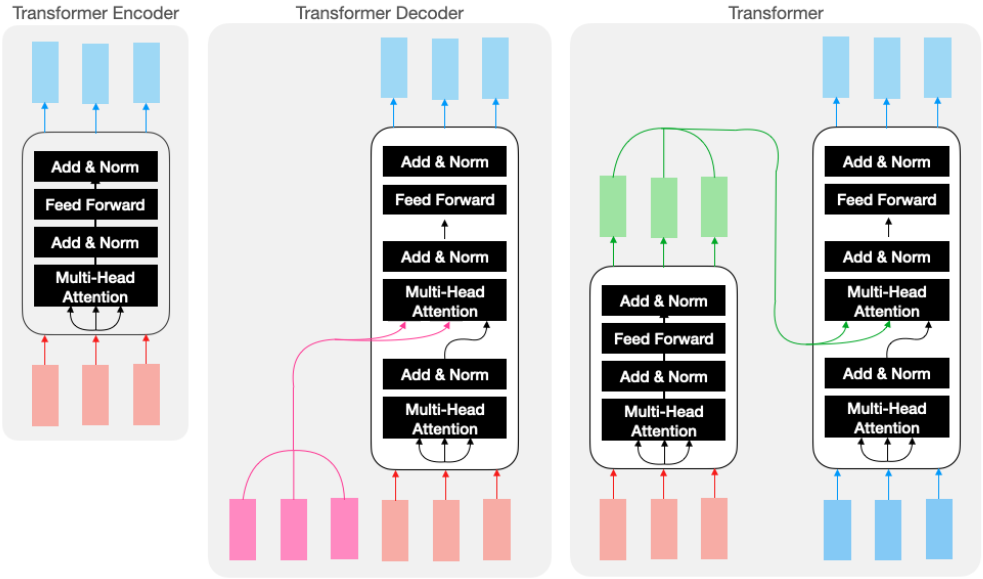


Figure 3.5: Overview of the Transformer Encoder, Transformer Decoder and Transformer architecture. The Transformer Encoder processes an input sequence (red) and maps it into a higher-dimensional space (blue), creating a set of representations that encapsulate the information about each part of the input while also considering the context provided by the rest of the input via self-attention. The Transformer Decoder processes an input sequence (red) via masked self-attention and cross-attends to a memory sequence (pink). The Transformer architecture introduced in [194] combines the Encoder and Decoder where the Decoder cross-attends to the Encoders output (green).

In Figure 3.5 we present an overview of the Transformer Encoder, Transformer Decoder and Transformer architecture. The Transformer Encoder processes an input sequence (red) and maps it into a higher-dimensional space (blue), creating a set of representations that encapsulate the information about each part of the input while also considering the context provided by the rest of the input via self-attention. The Transformer Decoder processes an input sequence (red) via masked self-attention and cross-attends to a memory sequence (pink). The Transformer architecture introduced in [194] combines the Encoder and Decoder where the Decoder cross-attends to the Encoders output (green).

3.3.4.1 Positional Encoding

The Attention mechanism treats input sequences as sets and thus the Transformer does not inherently model the order or position of elements within a sequence. As for some use cases the order within the sequence is relevant a method is required which allows for the attention mechanism to understand how two elements are ordered. Positional encodings are added to sequence input tokens to provide some information about the position of the tokens in the sequence. In particular, positional encodings produce a unique signal for each frame in the sequence so that the model can easily distinguish between them. [194] proposes using sine and cosine functions of different frequencies for positional encoding:

$$\begin{aligned} \mathrm{PE}(k, 2i) &= \sin(\frac{k}{n^{\frac{2i}{d}}}) \\ \mathrm{PE}(k, 2i+1) &= \cos(\frac{k}{n^{\frac{2i}{d}}}) \end{aligned} \tag{3.23}$$

where d is the size of the feature dimension, $k \in \mathbb{N}^{\geqslant 0}$ is the index of the token in the sequence and $i \in (0, \frac{d}{2})$ is the index in feature space. The hyper-parameter n defines how gradual the frequencies progress. In the original Transformers paper $n = 10000$. As an alternative to the sinusoidal representation a fixed positional encoding can also be learned as linear code. One downside of such an absolute Positional Encoding is that it is limited to the length of the longest training sequence sample. To address this Relative Positional Encoding [174] consider the distance between sequence elements and thus allow to be translation-equivariant. This is achieved by adding the positional encoding to the attention score directly instead of the sequence token. Another benefit is that this relative positional encoding does not obscure the computed hidden states, allowing for better caching and hidden state reuse during training and inference. However, relative positional encoding introduces more parameters to the model increasing the risk of overfitting. On top of that, relative positional encoding further obscures the attention maps, making it harder to visualize and interpret a models attention mechanism.

3.4 Evaluation Metrics

Evaluating the quality of predicted human motion remains an open problem, due to the high degree of stochasticity. In this section we will review common evaluation metrics found in the literature.

Mean Angular Error (MAE): MAE assumes that the human pose is represented using a articulated model and that the pose parameters are Euler angles. MAE is the per-frame L1-distance from a ground-truth motion sequence $\mathbf{x}^{1:T}$ with a prediction motion sequence $\hat{\mathbf{x}}^{1:T}$ with T frames. More formally:

$$\mathrm{MAE}(\mathbf{x}^{1:T}, \hat{\mathbf{x}}^{1:T}) = \{ \sum_{j=1}^{J} |x_{t,j} - \hat{x}_{t,j}| \}_{t=1}^{T} \tag{3.24}$$

where $x_{t,j}$ is the j-th joint and frame t of the real motion and where $\hat{x}_{t,j}$ is the model-predicted equivalent. The main advantage of this metric is its simplicity as it can be quickly calculated. However, the non-continuous and ambiguous nature of Euler angles ensure that the evaluation results tend to be noisy which has been demonstrated in [152]. Furthermore, due to the high degree of stochasticity of human motion this metric is only meaningful for short time horizons of around $1s$. We utilize MAE in Chapters 6.

Mean Per Joint Positional Error: MPJPE assumes that human pose is represented in 3D coordinates and calculates the Euclidean distance between ground-truth and prediction joint locations. Similar to MAE, MPJPE is calculated per frame given a ground-truth sequence $\mathbf{x}^{1:T}$ and a predicted sequence $\hat{\mathbf{x}}^{1:T}$. MPJPE is defined as:

$$\mathrm{MPJPE}(\mathbf{x}^{1:T}, \hat{\mathbf{x}}^{1:T}) = \{ \sum_{j=1}^{J} |x_{t,j} - \hat{x}_{t,j}|_2 \}_{t=1}^{T} \tag{3.25}$$

Unlike MAE, MPJPE is evaluating in Euclidean space which is well-behaved for the Euclidean distance function. However, due to the high degree of stochasticity of human motion this metric is only meaningful for short time horizons of around $1s$. We utilize MPJPE in Chapters 6, 7 and 8.

Normalized Power Spectrum Similarity (NPSS): NPSS [65] evaluates sequences in the power spectrum to account for frequency shifts that cannot be captured by MAE. Intuitively, when comparing two sequences, ground-truth sequence $\mathbf{x}^{1:T}$ and predicted sequence $\hat{\mathbf{x}}^{1:T}$, a phase-shift, i.e a walking person shortly stopping before continuing to walk in the ground-truth motion, will result in a large per-frame error (MAE, MPJPE) while in reality both sequences might correctly contain walking motion. By comparing the sequences in frequency space, the similarity is calculated based on repeated motion patterns that may occur at different times in the two sequences. Periodic actions, such as walking, will have peaks in the signal at their respective frequencies while a-periodic actions will show a more uniform spread in frequency space. This way NPSS can be used to evaluate longer time horizons of around $4s$. In contrast to MAE and MPJPE, NPSS produces a scalar value given two sequences $\mathbf{x}^{1:T}$ and $\hat{\mathbf{x}}^{1:T}$:

$$\mathrm{NPSS}(\mathbf{x}^{1:T}, \hat{\mathbf{x}}^{1:T}) = \frac{\sum_{i=1}^{T} \sum_{j=1}^{J} p_{i,j} * \mathrm{emd}_{i,j}}{\sum_{i=1}^{T} \sum_{j=1}^{J} p_{i,j}} \tag{3.26}$$

where

$$p_{i,j} = \sum_{f} \chi_{i,j}^{\mathrm{norm}}[f] \tag{3.27}$$

is the total power of the i-th feature in the j-th sequence and with

$$\mathrm{emd}_{i,j} = \left| \hat{\chi}_{i,j}^{\mathrm{norm}}[f] - \chi_{i,j}^{\mathrm{norm}}[f] \right| \tag{3.28}$$

$$\chi_{i,j}^{\mathrm{norm}}[f] = \frac{\chi_{i,j}[f]}{\sum_{f'} \chi_{i,j}[f']} \tag{3.29}$$

where $\chi_{i,j}[f]$ is the squared magnitude spectrum of Discrete Fourier Transform coefficients per sequence i per feature dimension j of $\mathbf{x}_{i,j}[t]$. However, NPSS is uni-modal as it compares the motion to a single ground-truth sequence, which will not work in multi-modal situations where multiple vastly different future motions are plausible. We use NPSS for evaluation in Chapter 6.

Inception Score (IS): In [10] the well-known Inception Score [169] for images is extended to 3D human motion. The inception score measures the sample quality of a generative model and is defined as:

$$\mathrm{IS}(g, d) = \exp\Big[\mathbb{E}_{\mathbf{x}\sim g} \big[D_{\mathrm{KL}} \big(d(\cdot|\mathbf{x}) || \int d(\cdot|\mathbf{x}) g(\mathbf{x}) \mathrm{d}\mathbf{x} \big) \big] \Big] \tag{3.30}$$

where g is a model or probability distribution which we wish to evaluate, and where $d(\mathbf{y}|\mathbf{x})$ is a discriminator function which calculates the probability that a given sample $\mathbf{x}$ is of class $\mathbf{y}$, and where $d(\cdot|\mathbf{x})$ calculates the probability distribution over all labels conditioned on $\mathbf{x}$. In the original work [169] the inception network [182] is used as discriminator, hence the name *Inception Score*.

To extend IS to human motion, a skeleton-based action classifier [112] is trained. We evaluate with IS in Chapter 6.

Average Pairwise Distance (APD): When evaluting generative models, it is important to measure not just the motion quality but also the sample diversity to detect defects such as mode collapse. To measure the sample diversity of probabilistic or generative models, APD is used which calculates the mean L2 distance between all samples. Given a set of S sample human motion sequences $\mathcal{X}^{1:T} = \{\hat{\mathbf{x}}_s^{1:T}\}_s^S$ of length T, APD is calculated as:

$$\text{APD}(\mathcal{X}^{1:T}) = \frac{1}{S^2 \cdot T \cdot J} \sum_{a=1}^{S} \sum_{b=1}^{S} \sum_{t=1}^{T} \sum_{j=1}^{J} |\hat{\mathbf{x}}_a^{(t)}[j] - \hat{\mathbf{x}}_b^{(t)}[j]|_2 \tag{3.31}$$

where $\mathbf{x}_s^{(t)}[j]$ is the j-th joint of sample sequence s at frame t. We evaluate with APD in Chapters 6 and 7.

Average Displacement Error (ADE): As human motion is highly stochastic many samples produced by a generative model might be far from the ground-truth motion sequence. However, we assume that, given a large enough number of samples, at least one sequence will be produced that is close to the ground-truth sequence. ADE calculates the average L2 distance of the sample closest to the ground truth sequence. Given a set of S sample human motion sequences $\mathcal{X}^{1:T} = \{\hat{\mathbf{x}}_s^{1:T}\}_s^S$ and a ground-truth sequence $\mathbf{x}^{1:T}$ of length T, ADE is calculated as:

$$\text{ADE}(\mathcal{X}^{1:T}, \mathbf{x}^{1:T}) = \min_{\hat{\mathbf{x}}_s^{1:T} \in \mathcal{X}^{1:T}} \frac{1}{T} \sum_{t=1}^{T} |\mathbf{x}^{(t)} - \hat{\mathbf{x}}_s^{(t)}|_2 \tag{3.32}$$

Crucially, this does not provide a score of how well all the other samples are. We evaluate with ADE in Chapter 6.

Final Displacement Error (FDE): As human motion is highly stochastic many samples produced by a generative model might be far from the ground-truth motion sequence. However, we assume that, given a large enough number of samples, at least one sequence will be produced that is close to the ground-truth sequence. FDE calculates the L2 distance of the last frame of the sample closest to the ground truth sequence. Given a set of S sample human motion sequences $\mathcal{X}^{1:T} = \{\hat{\mathbf{x}}_s^{1:T}\}_s^S$ and a ground-truth sequence $\mathbf{x}^{1:T}$ of length T, FDE is calculated as:

$$\text{FDE}(\mathcal{X}^{1:T}, \mathbf{x}^{1:T}) = \min_{\hat{\mathbf{x}}_s^{1:T} \in \mathcal{X}^{1:T}} |\mathbf{x}^{(T)} - \hat{\mathbf{x}}_s^{(T)}|_2 \tag{3.33}$$

Crucially, this does not provide a score of how well all the other samples are. We evaluate with FDE in Chapter 6.

CHAPTER 4

Iterative Greedy Matching for 3D Human Pose Tracking from Multiple Views

In this chapter we introduce a method for multiple human pose tracking from multiple calibrated cameras. This problem can be formulated as an n-partite graph matching between camera views, persons in the scenes, and frames, which is NP-complete. We propose a greedy algorithm which solves this in quadratic time by utilizing a greedy iterative matching strategy. This approach is simple yet effective and yields stronger results than more complex state-of-the-art methods. Furthermore, this method builds on top of 2D multiple human pose estimation methods, meaning that it can perform even stronger, given a stronger 2D human pose estimation method.

Individual Contribution: The following chapter is based on the publication [183]:

Iterative Greedy Matching for 3D Human Pose Tracking from Multiple Views
Julian Tanke and Juergen Gall
DAGM German Conference on Pattern Recognition (GCPR), 2019.

This publication was done by Julian Tanke and Juergen Gall provided scientific guidance and supported this work with very valuable feedback and suggestions.

Contents

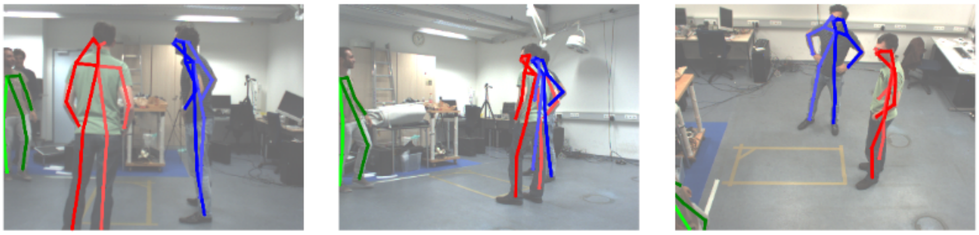


Figure 4.1: Qualitative results on the Shelf [17] dataset.

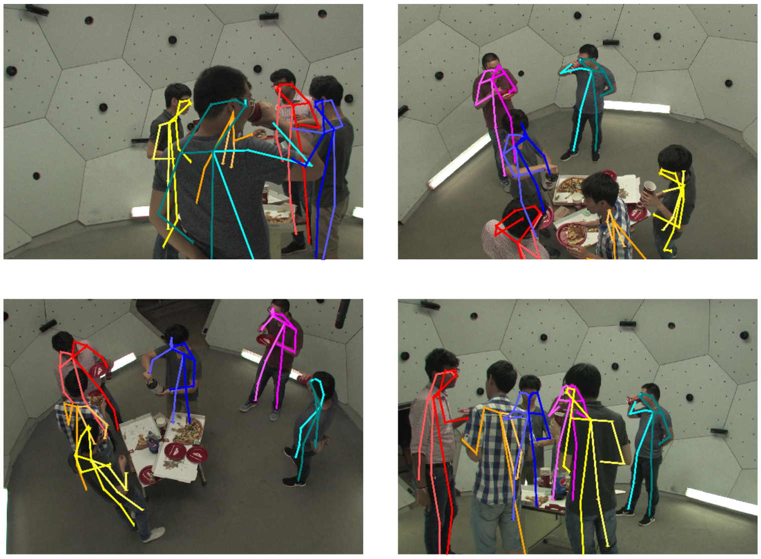


Figure 4.2: Challenging 3D reconstruction of 6 persons in the *CMU Panoptic Dataset* [96] with significant occlusion and partial visibility of persons.

4.1 Introduction

3D human pose tracking has applications in surveillance [220] and analysis of sport events [27, 101]. Most existing approaches [88, 89, 107, 121, 133, 150, 190, 135, 136] address 3D human pose estimation from single images while multi-view 3D human pose estimation [27, 101, 17, 20, 52] remains less explored, as obtaining and maintaining a configuration of calibrated cameras is difficult and costly. However, in sports or surveillance, calibrated multi-camera setups are available and can be leveraged for accurate human pose estimation and tracking. Utilizing multiple views has several obvious advantages over monocular 3D human pose estimation: ambiguities introduced by foreshortening as well as body joint occlusions or motion blurs can be resolved using other views. Furthermore, human poses are estimated within a global coordinate system when using calibrated cameras.

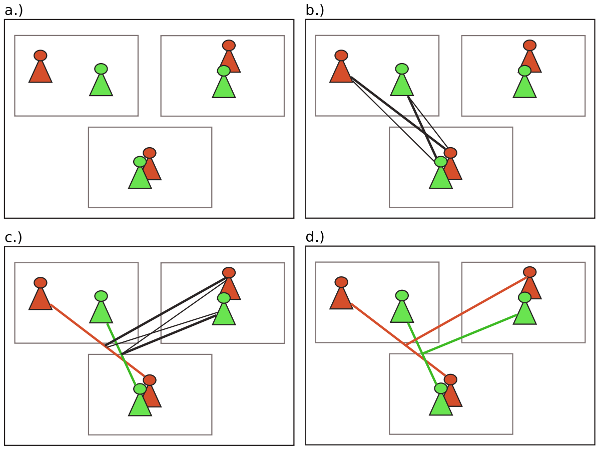


Figure 4.3: Estimating multiple people from multiple views can be formulated as k-partite graph partitioning where 2D human pose detections must be associated across multiple views. We employ a greedy approach to make the partitioning tractable. Given a set of 2D human pose detections on multiple views (a) we greedily match all detections on two images (b) where the weight between two detections is defined by the average epipolar distance of the two poses. Other views are then integrated iteratively where the weight is the average of the epipolar distance of the 2D detections in the new view and the already integrated 2D detections (c). 2D detections with the same color represent the same person.

In this work we propose an iterative greedy matching algorithm based on epipolar geometry to approximately solve the k-partite matching problem of multiple human detections in multiple cameras. To this end we utilize a real-time 2D pose estimation framework and achieve very strong results on challenging multi-camera datasets. The common 3D space proves to be very robust for greedy tracking, resulting in a very efficient and well-performing algorithm. In contrast to previous works [27, 101, 149, 54], our approach does not discretize the solution space but combines triangulation with an efficient pose association approach across camera views and time. Furthermore, our approach does not utilize individual shape models for each person [121].

We make the following contributions: (i) we present a greedy approach for 3D multi-person tracking from multiple calibrated cameras and show that our approach achieves state-of-the-art results. (ii) We provide extensive experiments on both 3D human pose estimation and on 3D human pose tracking on various multi-person multi-camera datasets.

4.2 Method

Our model consists of two parts: First, 3D human poses are estimated for each frame. Second, the estimated 3D human poses are greedily matched into tracks which is described in Section 4.2.2. To remove outliers and to fill-in missing joints in some frames, a simple yet effective smoothing scheme is applied, which is also discussed in Section 4.2.2.

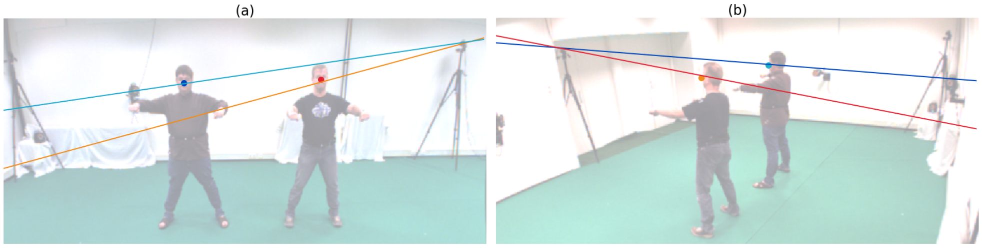


Figure 4.4: Epipolar lines for two camera views of the UMPM Benchmark [3]. The blue and the red dot in image (a) are projected as blue (red) epipolar lines in the second image (b) while the orange and light-blue dot from image (b) are projected onto image (a).

4.2.1 3D Human Pose Estimation

First, 2D human poses are extracted for each camera separately. Several strong 2D multi-person pose estimation [32, 39, 57, 71, 105, 143, 165, 204] models have been proposed but in our baseline we utilize OpenPose [32] as it is well established and offers real-time capabilities. We denote the 2D human pose estimations as

$$\{h_{i,k}\}_{i\in[1,N]}^{k\in[1,K_i]} \tag{4.1}$$

where N is the number of calibrated cameras and K_i the number of detected human poses for camera i.

In order to estimate the 3D human poses from multiple cameras, we first associate the detections across all views as illustrated in Figure 4.3. We denote the associated 2D human poses as $\mathcal{H}$ where $|\mathcal{H}|$ is the number of detected persons and $\mathcal{H}_m = \{h_{i,k}\}$ is the set of 2D human poses that are associated to person m. Once the poses are associated, we estimate the 3D human poses for all detected persons m with $|\mathcal{H}| > 1$ by triangulating the 2D joint positions.

For the association, we select camera $i = 1$ as starting point and choose all 2D human pose detections $h_{1,k}$ in this camera view as person candidates, i.e., $\mathcal{H} = \{\{h_{1,k}\}\}$. We then iterate over the other cameras and greedily match their 2D detections with the current list of person candidates $\mathcal{H}$ using bi-partite matching [141].

The cost for assigning a pose $h_{i,k}$ to an existing person candidate $\mathcal{H}_m$ is given by

$$\Phi(h_{i,k}, \mathcal{H}_m) = \frac{1}{|\mathcal{H}_m||J_{kl}|} \sum_{h_{j,l}\in\mathcal{H}_m} \sum_{\iota\in J_{kl}} \phi(h_{i,k}(\iota), h_{j,l}(\iota)) \tag{4.2}$$

where $h_{i,k}(\iota)$ denotes the 2D pixel location of joint ι of the 2D human pose $h_{i,k}$ and J_{kl} is the set of joints that are visible for both poses $h_{i,k}$ and $h_{j,l}$. Note that the 2D human pose detections might not contain all J joints due to occlusions or truncations. The distance between two joints in the respective cameras is defined by the distance between the epipolar lines and the joint locations:

$$\phi(p_i, p_j) = |p_j^T F^{i,j} p_i| + |p_i^T F^{j,i} p_j| \tag{4.3}$$

where $F^{i,j}$ is the fundamental matrix from camera i to camera j. Figure 4.4 shows the epipolar lines for two joints.

```
Result: Associated 2D poses $\mathcal{H}$
$\mathcal{H} := \{\{h_{1,k}\}\}$ ;
for camera $i \leftarrow 2$ to $N$ do
    for pose $k \leftarrow 1$ to $K_i$ do
        for hypothesis $m \leftarrow 1$ to $|\mathcal{H}|$ do
            $C_{k,m} = \Phi(h_{i,k}, \mathcal{H}_m)$ ;
        end
    end
    $X^* = \underset{X}{\operatorname{argmin}} \sum_{m=1}^{|\mathcal{H}|} \sum_{k=1}^{K_i} C_{k,m} X_{k,m}$ ;
    for $k, m$ where $X^*_{k,m} = 1$ do
        if $C_{k,m} < \theta$ then
            $\mathcal{H}_m = \mathcal{H}_m \bigcup \{h_{i,k}\}$ ;
        else
            $\mathcal{H} = \mathcal{H} \bigcup \{\{h_{i,k}\}\}$ ;
        end
    end
end
$\mathcal{H} = \mathcal{H} \backslash \mathcal{H}_m \ \forall m$ where $|\mathcal{H}_m| = 1$;
```

Algorithm 1: Solving the assignment problem for multiple 2D human pose detections in multiple cameras. $\Phi(h_{i,k}, \mathcal{H}_m)$ (4.2) is the assignment cost for assigning the 2D human pose $h_{i,k}$ to the person candidate $\mathcal{H}_m$. X^* is a binary matrix obtained by solving the bi-partite matching problem. The last line in the algorithm ensures that all hypotheses that cannot be triangulated are removed.

Using the cost function $\Phi(h_{i,k}, \mathcal{H}_m)$, we solve the bi-partite matching problem for each image i:

$$X^* = \underset{X}{\operatorname{argmin}} \sum_{m=1}^{|\mathcal{H}|} \sum_{k=1}^{K_i} \Phi(h_{i,k}, \mathcal{H}_m) X_{k,m} \tag{4.4}$$

where

$$\sum_k X_{k,m} = 1 \ \forall m \quad \text{and} \quad \sum_m X_{k,m} = 1 \ \forall k.$$

$X^*_{k,m} = 1$ if $h_{i,k}$ is associated to an existing person candidate $\mathcal{H}_m$ and it is zero otherwise. If $X^*_{k,m} = 1$ and $\Phi(h_{i,k}, \mathcal{H}_m) < \theta$, the 2D detection $h_{i,k}$ is added to $\mathcal{H}_m$. If $\Phi(h_{i,k}, \mathcal{H}_m) \geqslant \theta$, $\{h_{i,k}\}$ is added as hypothesis for a new person to $\mathcal{H}$. Algorithm 1 summarizes the greedy approach for associating the human poses across views.

4.2.2 Tracking

For tracking, we use bipartite matching [141] similar to Section 4.2.1. Assuming that we have already tracked the 3D human poses until frame $t - 1$, we first estimate the 3D human poses for frame t as described in Section 4.2.1. The 3D human poses of frame t are then associated to the 3D human poses of frame $t - 1$ by bipartite matching. The assignment cost for two 3D human poses is in this case given by the average Euclidean distance between all joints that are present in both poses. In some cases, two poses do not have any overlapping valid joints due to noisy detections or truncations.

	[27]*	[101]*	[149]*	[17]	[20]	[54]	Ours	Ours$^+$
ua	.60	.89	1.0	.68	.98	.97	.99	1.0
la	.35	.68	1.0	.56	.72	.95	.99	1.0
ul	1.0	1.0	1.0	.78	.99	1.0	.98	.99
ll	.90	.99	1.0	.70	.92	.98	.93	.997
avg	.71	.89	1.0	.68	.90	.98	.97	**.997**

Table 4.1: Quantitative comparison of methods for single human 3D pose estimation from multiple views on the KTH Football II [101] dataset. The numbers are the PCP score in 3D with $\alpha = 0.5$. Methods annotated with * can only estimate single human poses, discretize the state space and rely on being provided with a tight 3D bounding box centered at the true 3D location of the person. *Ours*$^+$ and *Ours* describe our method with and without track smoothing (Section 4.2.2). *ul* and *la* show the scores for upper and lower arm, respectively, while *ul* and *ll* represent upper and lower legs.

The assignment cost is then calculated by projecting the mean of all valid joints of each pose onto the xy-plane, assuming that the z-axis is the normal of the ground plane, and taking the Euclidean distance between the projected points. As long as the distance between two matched poses is below a threshold τ, they will be integrated into a common track. Otherwise, a new track is created. In our experiments we set $\tau = 200mm$.

Due to noisy detections, occlusions or motion blur, some joints or even full poses might be missing in some frames or noisy. We fill in missing joints by temporal averaging and we smooth each joint trajectory by a Gaussian kernel with standard deviation σ. This simple approach significantly boosts the performance of our model as we will show in Section 4.3.

4.3 Experiments

We evaluate our approach on two human pose estimation tasks, single person 3D pose estimation and multi-person 3D pose estimation, and compare it to state-of-the-art methods. Percentage of correct parts (PCP) in 3D as described in [27] is used for evaluation. We evaluate on the limbs only as annotated head poses vary significantly throughout various datasets. In all experiments, the order in which the cameras are processed is given by the dataset. We then evaluate the tracking performance. The source code is made publicly available [1].

4.3.1 Single Person 3D Pose Estimation

Naturally, first works on 3D human pose estimation from multiple views cover only single humans. Typical methods [27, 101, 149] find a solution over the complete discretized state space which is intractable for multiple persons. However, we report their results for completeness. All models were evaluated on the complete first sequence of the second player of the KTH Football II [101] dataset. Our results are reported in Table 4.1. Our model outperforms all other multi-person approaches and gets close to the state-of-the-art for single human pose estimation [149] which makes strong assumptions and is much more constrained. Our model has the lowest accuracy for lower legs (*ll*)

[1] https://github.com/jutanke/mv3dpose

Campus dataset $(\alpha = 0.5)$

	[17]			[20]			[54]			Ours			Ours+		
Actor	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
ua	.83	.90	.78	.97	.97	.90	.97	.94	93	.86	.97	.91	.99	.98	.98
la	.78	.40	.62	.86	.43	.75	.87	.79	70	.74	.64	.68	.91	.70	.92
ul	.86	.74	.83	.93	.75	.92	.94	.99	88	1.0	.99	.99	1.0	.98	1.0
ll	.91	.89	.70	.97	.89	.76	.97	.95	81	1.0	.98	.99	1.0	.98	.99
avg	.85	.73	.73	.93	.76	.83	.94	.93	.85	.90	.90	.89	.98	.91	.98
avg*		.77			.84			.91			.90			**.96**	

Shelf dataset $(\alpha = 0.5)$

	[17]			[20]			[54]			Ours			Ours+		
Actor	1	2	3	1	2	3	1	2	3	1	2	3	1	2	3
ua	.72	.80	.91	.82	.83	.93	.93	.78	.94	.99	.93	.97	.1.0	.97	.97
la	.61	.44	.89	.82	.83	.93	.83	.33	.90	.97	.57	.95	.99	.64	.96
ul	.37	.46	.46	.43	.50	.57	.96	.95	.97	.998	1.0	1.0	1.0	1.0	1.0
ll	.71	.72	.95	.86	.79	.97	.97	.93	.96	.998	.99	1.0	1.0	1.0	1.0
avg	.60	.61	.80	.73	.74	.85	.92	.75	.94	.99	.87	.98	.998	.90	.98
avg*		.67			.77			.87			.95			**.96**	

Table 4.2: Quantitative comparison of multi-person 3D pose estimation from multiple views on the evaluation frames of the annotated Campus [60, 17] and Shelf dataset [17]. The numbers are the PCP score in 3D with $\alpha = 0.5$. *Ours*+ and *Ours* describe our method with and without track smoothing (Section 4.2.2). We show results for each of the three actors separately as well as averaged for each method (*average**).

which experience strong deformation and high movement speed. This can be mostly attributed to the 2D pose estimation framework which confuses left and right under motion blur, as can be seen in Figure 4.7. When smoothing the trajectory (Section 4.2.2) this kind of errors can be reduced.

4.3.2 Multi-Person 3D Pose Estimation

To evaluate our model on multi-person 3D pose estimation, we utilize the Campus [60, 17], Shelf [17], CMU Panoptic [96] and UMPM [3] dataset. The difficulty of the Campus dataset lies in its low resolution (360 × 288 pixel) which makes accurate joint detection hard. Furthermore, small errors in triangulation or detection will result in large PCP errors as the final score is calculated on the 3D joint locations. As in previous works [17, 20] we utilize frames $350 - 470$ and frames $650 - 750$ of the Campus dataset and frames $300 - 600$ for the Shelf dataset. Clutter and humans occluding each others make the Shelf dataset challenging. Nevertheless, our model achieves state-of-the-art results on both datasets by a large margin which can be seen in Table 4.2. Table 4.3 reports quantitative results on video *p2_chair_2* of the UMPM [3] benchmark. A sample frame from this benchmark can be seen in Figure 4.4. As the background is homogeneous and the human actors maintain a considerable distance to each other the results of our method are quite strong.

	Ours^{+}	
Actor	1	2
ua	.997	.98
la	.98	.996
ul	1.0	1.0
ll	.99	.997
avg	0.99	0.99

Table 4.3: Quantitative comparison of multi-person 3D pose estimation from multiple views on *p2_chair_2* of the UMPM benchmark [3].

	Ours	Ours^{+}
160422_ultimatum1 [96]	.89	.89
160224_haggling1 [96]	.92	.92
160906_pizza1 [96]	.92	.93

Table 4.4: Quantitative evaluation of multi-person 3D pose tracking on the CMU Panoptic dataset [96] using the MOTA [21] score. *Ours*$^{+}$ and *Ours* describe our method with and without track smoothing (Section 4.2.2).

4.3.3 Tracking

For evaluating the tracking accuracy, we utilize the MOTA [21] score which provides a scalar value for the rate of false positives, false negatives, and identity switches of a track. Our model is evaluated on the CMU Panoptic dataset [96] which provides multiple interacting people in close proximity. We use videos *160224_haggling1* with three persons, *160422_ultimatum1* with up to seven person, and *160906_pizza1* with six persons. For the videos *160422_ultimatum1* we use frames 300 to 3758, for *160906_pizza1* we use frames 1000 to 4458 and for *160224_haggling1* we use frames 4209 to 5315 and 6440 to 8200. The first five HD cameras are used. Our results are reported in Table 4.4 which shows that our approach yields strong tracking capabilities.

4.3.4 Effects of Smoothing

4.3.5 Early Commitment

As can be seen in Table 4.1 and Table 4.2 the effects of smoothing can be significant, especially when detection and calibration are noisy as is the case with the Campus and the KTH Football II dataset. In both datasets 2D human pose detection is challenging due to low resolution (Campus) or strong motion blur (KTH Football II). Datasets with higher resolution and less motion blur like the Shelf dataset do not suffer from this problems as much and as such do not benefit the same way from track smoothing. However, a small gain can still be noted as smoothing also fills in joint detections that could not be triangulated. Figure 4.5 explores different σ values for smoothing on the KTH Football II, Campus, and Shelf dataset. It can be seen that smoothing improves the performance regardless of the dataset but that too much smoothing obviously reduces the accuracy. We chose $\sigma = 2$ for all our

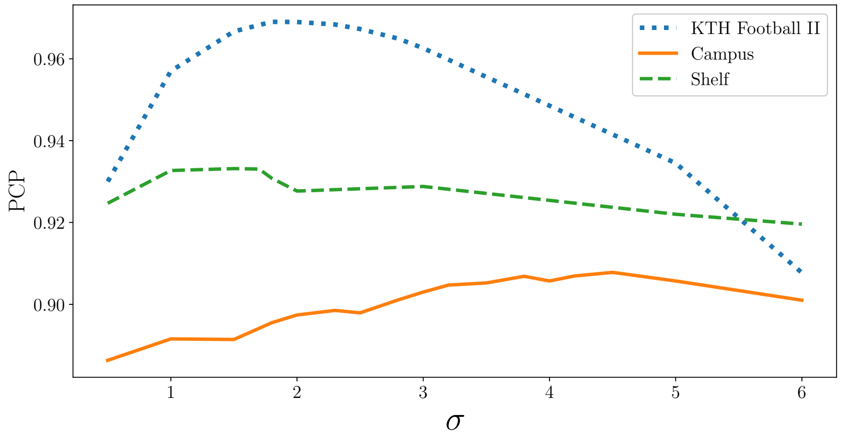


Figure 4.5: PCP score for different smoothing values σ for tracking on KTH Football II, Campus, and Shelf. If σ is too small, the smoothing has little effect and coincides with the un-smoothed results. When the joint trajectories are smoothed too much, the PCP score drops as well as the trajectories do not follow the original path anymore. (Larger PCP scores are better)

experiments except for the Campus dataset where we set $\sigma = 4.2$. The reason for the higher value of σ for the Campus dataset is due to the very low resolution of the images compared to the other datasets, which increases the noise of the estimated 3D joint position by triangulation.

4.3.6 Effects of camera order

So far we used the given camera order for each dataset, but the order in which views are greedily matched matters and different results might happen with different orderings. To investigate the impact of the camera order, we evaluated our approach using all 120 permutations of the 5 cameras of the Shelf dataset. The results shown in Figure 4.6 show that the approach is very robust to the order of the camera views.

A failure case happens due to the early commitment of our algorithm with regards to the 2D pose estimation, as can be seen in Figure 4.7. When the pose estimation is unsure about a pose, it still fully commits to its output and disregards uncertainty. This problem occurs due to motion blur as the network has difficulties to decide between left and right in this case. As our pose estimation model has mostly seen forward-facing persons it will be more inclined towards predicting a forward-facing person in case of uncertainty. When left and right of a 2D prediction are incorrectly flipped in at least one of the views, the merged 3D prediction will collapse to the vertical line of the person resulting in a poor 3D pose estimation.

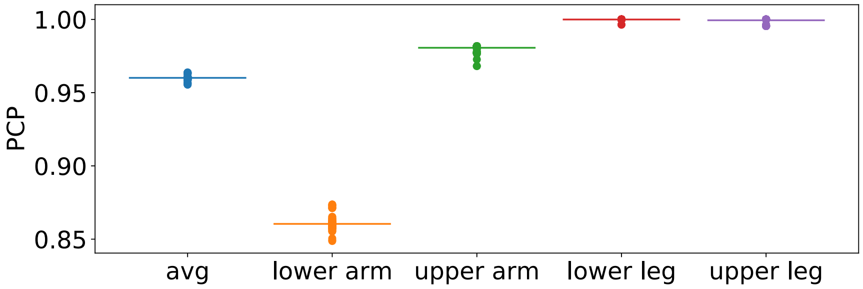


Figure 4.6: PCP score averaged over all subjects for all 120 camera permutations of the Shelf dataset. The vertical line represents the mean value over all permutations while the dots represent each camera permutation.

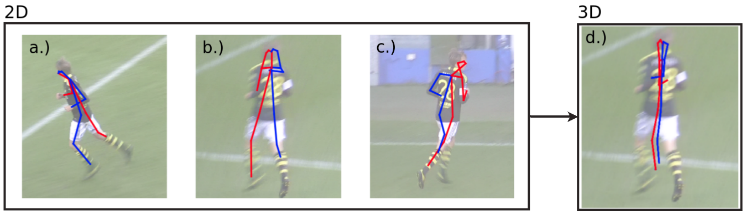


Figure 4.7: Issues with early commitment. As we utilize the 2D pose estimations directly, our method suffers when the predictions yield poor results. In this example the pose estimation model correctly estimates (a) and (c) but confuses left and right on (b) due to motion blur. The resulting 3D pose estimation (d) collapses into the centre of the person. The red limbs represent the right body side while blue limbs represent the left body side.

4.3.7 Conclusion

In this work we presented a simple baseline approach for 3D human pose estimation and tracking from multiple calibrated cameras and evaluate it extensively on several 3D multi-camera datasets. Our approach achieves state-of-the-art results in multi-person 3D pose estimation while remaining sufficiently efficient for fast processing. Due to the models simplicity some common failure cases can be noted which can be build upon in future work. For example, confidence maps provided by the 2D pose estimation model could be utilized to prevent left-right flips. Our approach may serve as a baseline for future work.

CHAPTER 5

Recursive Bayesian Filtering for Multiple Human Pose Tracking from Multiple Cameras

In this chapter we introduce a method for multiple human pose tracking from multiple calibrated cameras utilizing the Recursive Bayesian Filtering technique. We build on top of the previous work to reduce the algorithmic runtime from quadratic in the number of persons in the scene to only linear, so that that this method can be applied to very large datasets with large groups of persons.

Individual Contribution: The following chapter is based on the publication [109]:

Recursive Bayesian Filtering for Multiple Human Pose Tracking from Multiple Cameras
Oh-Hun Kwon, Julian Tanke and Juergen Gall
IEEE Asian Conference on Computer Vision (ACCV), 2020.

This publication was done in very close collaboration between Oh-Hun Kwon and Julian Tanke. This publication is based on the Master thesis of Oh-Hun Kwon who was advised and supported by Julian Tanke. Juergen Gall provided scientific guidance and supported this work with very valuable feedback and suggestions. The idea to utilize Recursive Bayesian Filtering was propose by Oh-Hun Kwon, the forecasting scheme was proposed by Julian Tanke. The method implementation was done by Oh-Hun Kwon and the forecasting method as well as the pose normalization was supplied by Julian Tanke. The paper experiments closely follow the experiments of the previous work in Chapter 4. The paper was written by Julian Tanke.

Contents

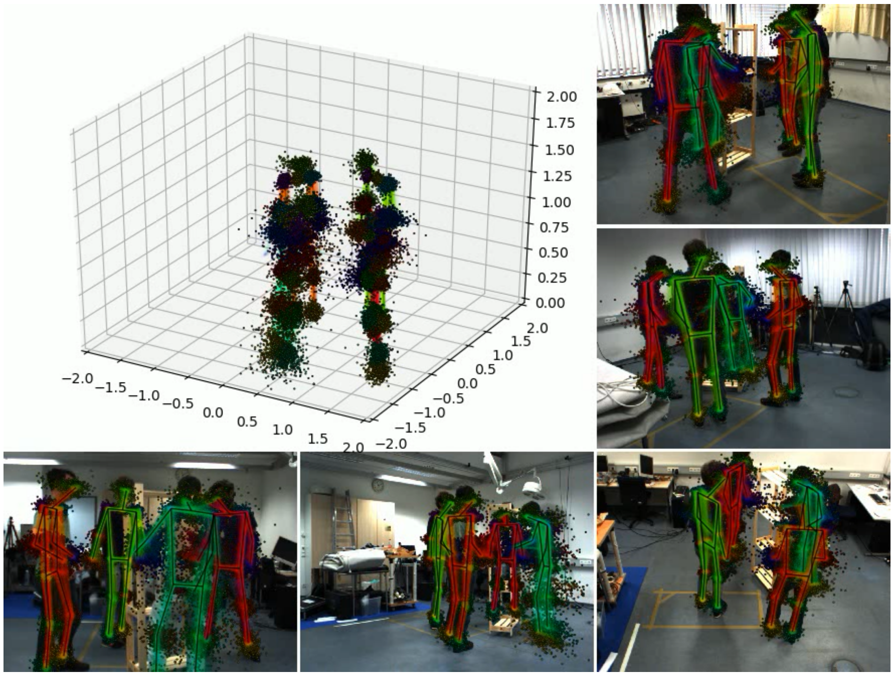


Figure 5.1: Probabilistic representation for 3D pose tracking. The black points represent 3D pose predictions from the prediction step while the colored skeletons represent the pose samples after the update step. Notice that both representations model uncertainty. The final pose is the black skeleton at the center of each person.

5.1 Introduction

Markerless motion capture [121, 122, 54, 218, 51, 17, 18, 20, 183] has many applications in sports [27, 101] and surveillance [220]. Utilizing multiple calibrated cameras extends the field of view, allows to resolve ambiguities such as foreshortening and occlusions, and provides accurate 3D estimates. However, challenges still remain: large crowds and close interactions result in heavy occlusions which severely degrade the 3D tracking performance. Furthermore, most recent works [54, 218, 51, 183] cast multiple 3D human pose estimation from multiple camera views as an association problem where extracted 2D pose features have to be matched across views and across time. This way, the time complexity grows quadratic [183] or even exponential [218, 51] with the number of tracked individuals, making tracking of large numbers of persons impractical.

In this work we cast the problem of estimating multiple persons from multiple calibrated cameras as a tracking problem where each person is individually tracked using the well-known recursive Bayesian filtering method [171]. Individually tracking each person results in a linear relationship between time complexity and the number of persons in the scene. Furthermore, utilizing a tracking framework enables us to retain plausible poses even under temporary heavy occlusion. Last but not least, the Bayesian framework allows us to quantify uncertainty.

Recursive Bayesian filtering naturally lends itself for human pose tracking from multiple cameras. It models an underlying process $z_{1:T}$, which we are interested in but which we cannot directly observe. Instead, at each time step t we receive observations o_t which are related to z_t. Bayesian filtering provides us with tools to form our best guess about z_t given the observations $o_{1:t}$. For 3D human pose tracking, the unobserved hidden state z_t represents the 3D pose at time t while the observation o_t represents the camera input at time t for all cameras. Bayesian filtering utilizes a prediction step, which forecasts the current estimate in time, and an update step, which incorporates current observations into the prediction. To model uncertainty we utilize a sample-based approach for z_t. For the prediction step we build on recent advancements in 3D human motion anticipation [132] and utilize a sequence-to-sequence model. During the update step we process all samples in z_t and make use of importance sampling, similar to the particle filter [171]. In order to reduce the number of required particles, we combine it with an optimization step to find good 3D poses. Our method achieves state-of-the-art results on common multiple person multi-camera pose estimation and tracking benchmarks.

5.2 Method

In this work we formulate the problem of estimating and tracking multiple 3D human poses from multiple calibrated cameras as a recursive Bayesian filter where the hidden states z_t represent the 3D human poses and where the camera images are the observations o_t at time step t. More precisely, each person in the scene has a 3D pose state z which is tracked independently through time, as described in Figure 5.2. This means that we have a Bayesian filter for each person. This has the advantage that we can easily deal with appearing and disappearing persons.

A Bayesian filter recursively cycles through prediction and update steps. The prediction step utilizes a prediction model $p(z_t|z_{t-1})$ which evolves the hidden state in time while the update step utilizes an observation model $p(o_t|z_t)$ which integrates measurements into the prediction. We build on recent advancements in 3D human motion anticipation [132] and model $p(z_t|z_{t-1})$ as a recurrent neural network (RNN) where uncertainty is represented by Dropout as Bayesian approximation [63]. The observation model $p(o_t|z_t)$ measures how well a 3D pose sample matches the extracted 2D joint confidence maps and part affinity fields [32, 31] for each camera view. For each tracked person, a set of 3D sample poses is used to represent the posterior $p(z_t|o_{1\dots t})$. A sample-based representation of the distribution [209, 64, 142] allows for a highly non-linear state space, which is required for complex human poses, while being simple to implement. In Section 5.2.1 we detail the prediction model while in Section 5.2.2 we discuss the observation model. The initialization procedure for $p(z_1|o_1)$ of each person is explained in Section 5.2.3. Finally, Section 5.2.4 explains how point samples can be obtained for each frame t from the estimated posterior $p(z_t|o_1 \dots o_t)$.

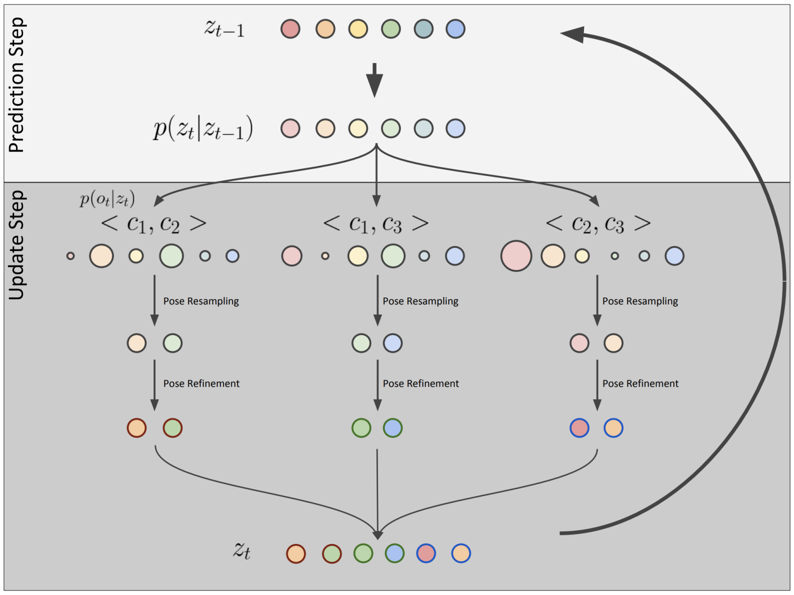


Figure 5.2: Tracking procedure for tracking a single person z with a set of three cameras c_1, c_2 and c_3. The prediction step forecast z_{t-1} from time $t-1$ to t. In the update step each pose sample gets assigned an importance weight independently for each camera pair. The importance weights are calculated using the observations o_t at time t. We then resample for each camera pair relative to the total number of samples and refine the poses using pose refinement. Finally, we concatenate the sub-samples for each camera pair and obtain our prediction for z_t.

5.2.1 Prediction Step

The prediction model $p(z_t|z_{t-1})$ evolves the pose state of a single tracked person in time - without taking any observation into account. The pose state z_t of a person encompasses possible 3D poses which we make tractable by representing them as a fixed set of 3D pose samples. A sample is made up of 14 3D joints.

We represent $p(z_t|z_{t-1})$ as GRU [132] and we inject uncertainty by utilizing Dropout during training and inference at the final linear layer that extracts z_t. Dropout is crucial to generate a diverse set of forecast poses which we will discuss in our ablation studies. As z_{t-1} is represented as a list of 3D pose samples, we apply the forecast for each sample independently with an independent hidden layer for the GRU for each sample. This way, z_t and z_{t-1} will be represented by the same number of samples while samples will be sufficiently varied due to the independent forecasting. For brevity, we define this as:

$$z_t, h_t = \text{GRU}(z_{t-1}, h_{t-1}) \tag{5.1}$$

where z_t, h_t, z_{t-1} and h_{t-1} are 3D pose samples and GRU hidden states, respectively.

The 3D poses in z are in a global coordinate frame which is defined by the calibrated cameras. We transform the 3D poses into a standardized coordinate frame before forecasting. Here, the center hip joint of the poses in z_{t-1} are set as the origin and the poses are rotated along the z axis[1] such that the left and right hip joints align to the y axis and such that the 3D pose faces forward along the x axis. More formally, we apply the following transformation to each 3D pose

$$\hat{\mathbf{x}}_j = R^{(t-1)} \left(\mathbf{x}_j - \mathbf{x}_{\text{hiproot}}^{(t-1)}\right) \forall j \in \mathrm{J} \tag{5.2}$$

where J represents all joints that make up a 3D pose and where $\mathbf{x}_j$ represents the j-th joint as 3D point in global coordinate space and where $\hat{\mathbf{x}}_j$ represents the same joint in normalized coordinates. The hip root joint of the pose at time $t-1$ is defined as $\mathbf{x}_{\text{hiproot}}^{(t-1)}$ and the rotation to forward-face the pose at $t-1$ is defined as

$$R^{(t-1)} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{5.3}$$

$$\theta = \operatorname{atan2}\big(y_{\text{righthip}}^{(t-1)} - y_{\text{lefthip}}^{(t-1)}, x_{\text{righthip}}^{(t-1)} - x_{\text{lefthip}}^{(t-1)}\big) \tag{5.4}$$

where $x_{\text{righthip}}^{(t-1)}$, $x_{\text{lefthip}}^{(t-1)}$, $y_{\text{righthip}}^{(t-1)}$ and $y_{\text{lefthip}}^{(t-1)}$ represent the x and y coordinate of the right hip and left hip, respectively. After forecasting a pose, the original position and orientation in global coordinates can be recovered by applying the transformation

$$\mathbf{x}_j = {R^{(t-1)}}^T \hat{\mathbf{x}}_j + \mathbf{x}_{\text{hiproot}}^{(t-1)} \forall j \in \mathrm{J}. \tag{5.5}$$

The prediction model is trained with motion capture data from the Human3.6M [85] and the CMU mocap database [1] where we select 14 joints that the two datasets have in common. We utilize Adam with learning rate 0.001 and optimize over the Huber loss. The number of hidden units for the GRU is set to 2048. The dropout rate is set to 50% and a weight decay of 10^{-8} is added. We set the framerate to 25Hz and 30Hz, respectively, which is similar to the framerate used in the evaluation datasets.

5.2.2 Update Step

To obtain the posterior $p(z_t|o_{1,\ldots,t})$ for a single tracked person we need to incorporate the observations o_t into the predictions z_t obtained from the prediction model. For each camera we utilize Openpose [32] to extract part confidence maps and part affinity fields, similar to other multi-person multi-camera 3D pose estimation methods [183, 218]. We then calculate importance weights for each sample pose in z_t and then re-sample z_t based on the weights. To prevent poses that are visible in many camera views to be over-represented over poses that are visible in less cameras and to tackle false-positive detections caused by occlusion, we sample the importance weight for each camera pair independently - for all samples. The weight is calculated as follows:

$$w_{v,s} = \frac{\Phi(v,s)}{\sum_{\hat{s}}^{z_t} \Phi(v,\hat{s})} \tag{5.6}$$

where v represents a camera pair and where s represents a single 3D pose sample from z_t. We normalize by the scores of all samples $\hat{s}$ in z_t. The unnormalized weight $\Phi(\cdot,\cdot)$ is calculated as follows:

$$\Phi(v,s) = \prod_{l \in \mathrm{L}} \sqrt{\sum_{c \in v} \phi(c,l,s)^2} + \varepsilon \tag{5.7}$$

[1] assuming z axis points upwards

where L represents all limbs of a pose, as described in Openpose [32]. Each camera pair v consists of two different camera views c. The score $\phi(\cdot,\cdot,\cdot)$ is calculated using part affinity fields paf_c and confidence maps conf_c, which are obtained from Openpose [32], for a given camera c:

$$\phi(c,l,s) = \left(\int_{u=0}^{u=1} \max\big(0, \mathrm{paf}_c(s,l,u)\big)\, \mathrm{d}u \right) \prod_{j \in l} \mathrm{conf}_c(s,j) \tag{5.8}$$

where $\mathrm{paf}_c(s,l,u)$ calculates the dot product between the part affinity field for limb l and the projected limb from s, linearly interpolated by u. $\mathrm{conf}_c(s,j)$ calculates the confidence score for the joint j of sample s for camera c. Finally, we resample z_t for each camera pair a subset of particles to obtain the same number of initial samples as shown in Figure 5.2. As sampling procedure we use stochastic universal sampling.

In practice, the state space of a 3D pose is prohibitively large for a sample-based representation. However, we utilize a simple yet effective heuristic optimization called joint refinement to keep the number of samples low while obtaining accurate results. For each joint of a sample, we sample additional joint positions from a normal distribution centered at the joint. We then take the joint position with the highest confidence map score. In our ablation study we show that this significantly improves the results while it allows the numbers of samples for each person to be low.

5.2.3 Initialization Step

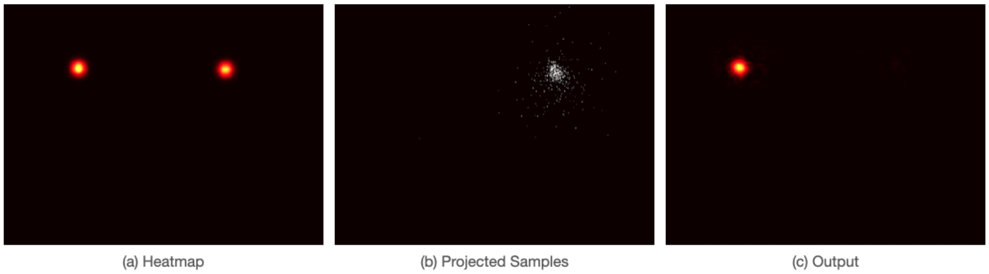

(a) Heatmap (b) Projected Samples (c) Output

Figure 5.3: Input and output of the confidence subtraction network. The input is composed of a confidence map (a) extracted by [32] for a specific joint and projected points (b) of that joint for the tracked person. The network removes the part of the heatmap that corresponds to the tracked person.

To facilitate multi-person 3D pose tracking, a set of currently tracked persons is kept which are all independently tracked using the prediction (Section 5.2.1) and update (Section 5.2.2) step. However, at each time step we have to check whether one or more untracked persons have entered the 3D recording volume and generate new tracks accordingly. To do so, we first remove currently tracked persons from the confidence maps for each camera using a confidence subtraction network. To remove a tracked person from a confidence map, we project the joints of all samples of that person for the given frame to that camera view (see Figure 5.3 (b)). We then pass the projected points as well as the confidence map to the confidence subtraction network which will return an updated confidence map without the peak of the tracked person. Figure 5.3 shows an example while Figure 5.4 details the network structure. We repeat this procedure for all tracked persons and for all camera views.

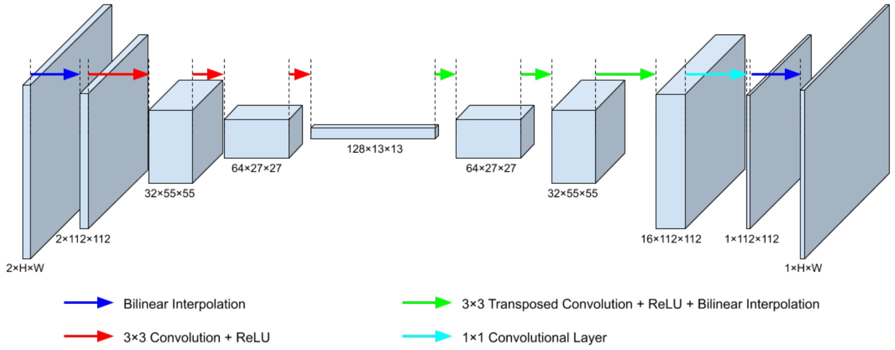


Figure 5.4: Fully convolutional architecture for the confidence map subtraction network.

Once all tracked persons are removed from the confidence maps, we find the remaining local maxima and triangulate them pairwise if both points are close to their respective epipolar line as in [183]. To reduce the number of redundant points, we apply agglomerative hierarchical clustering with threshold ε_j and use the mean point of the clusters. We then build a set of 3D pose candidates by greedily matching joints based on the part affinity fields [32]. We also drop limbs that have unreasonable length. Each pose candidate is then scored using Equation (5.7), where v contains all camera views, and the 3D pose with the highest score is selected for the new track.

As a person track is represented as a list of 3D pose samples, we utilize a stochastic generation function $z \leftarrow g(\cdot)$ which takes as input the previously selected best 3D pose candidates and generates a set of 3D pose samples z that represent the distribution of the newly generated track. Once the pose samples are generated the person can be tracked using the the prediction and update steps. The new pose is removed from the confidence maps and the initialization procedure is repeated until no further person tracks are found.

We model g as a three-layer feed-forward Bayesian neural network [63] which takes as input a pose vector and outputs a pose vector. As a person might only be partially visible g also fills in missing joints. This is facilitated by adding a binary vector to the input pose vector which indicates if a joint is missing. As dropout is utilized during inference, g generates a diverse set of 3D pose samples. The network is trained with motion capture data from Human3.6M [85] and from the CMU mocap database [1], similar to Section 5.2.1. During training, random joints are removed from the pose to encourage the model to fill in missing joints. The model has three layers with 2048 hidden units each and is optimized over the Huber loss using SGD with a learning rate of 0.001, weight decay of 10^{-6} and dropout of 75%.

5.2.4 Inference

Using multiple samples to represent a 3D pose allows for robust tracking. However, when extracting 3D poses a final single pose is required. To obtain a final pose from z_t for a tracked person at frame t, we calculate the weighted average for all samples using Equation (5.6) where v contains all cameras.

	Campus					Shelf				
	A1(48)	A2(188)	A3(136)	aAvg	gAvg	A1(279)	A2(37)	A3(161)	aAvg	gAvg
Belagiannis et al. [17]	82.01	72.43	73.72	76.05	74.14	66.05	64.97	83.16	71.39	71.75
+Belagiannis et al. [18]	83.00	73.00	78.00	78.00	76.12	75.00	67.00	86.00	76.00	78.09
Belagiannis et al. [19]	93.45	75.65	84.37	84.49	81.14	75.26	69.68	87.59	77.51	79.00
Ershadi-Nasab et al. [54]	94.18	92.89	84.62	90.56	90.03	93.29	75.85	94.83	87.99	92.46
Dong et al. [51]	97.40	90.10	89.40	92.30	90.79	97.20	79.50	96.50	91.07	95.59
*Dong et al. [51]	97.60	93.30	**98.00**	**96.30**	**95.57**	98.80	94.10	97.80	96.90	98.10
+Chaper 4	**98.00**	91.00	**98.00**	95.67	94.46	99.21	93.51	97.14	96.62	98.07
+Zhang et al. [218]	-	-	-	-	-	99.00	**96.20**	97.60	97.60	98.31
+Ours	97.35 ± 0.40	**93.44** ± 0.04	97.43 ± 0.18	96.07 ± 0.13	95.40 ± 0.07	**99.49** ± 0.06	95.81 ± 0.37	**97.83** ± 0.00	**97.71** ± 0.13	**98.64** ± 0.05

Table 5.1: Quantitative comparison with state-of-the-art methods using percentage of correctly estimated parts (PCP) on the Campus and Shelf datasets. *A1* to *A3* represent the three actors while the number in parentheses represents the number of ground-truth frames. We report both actor-wise (*aAvg*) as well as global average (*gAvg*) PCP. Models utilizing temporal information are marked with + while appearance information is marked with *. As our method is probabilistic, we report results as mean ± standard deviation, which is calculated over 10 runs using different random seeds.

5.3 Experiments

5.3.1 Quantitative Comparison

We provide a quantitative comparison to recent state-of-the-art methods using the Campus [17, 60] as well as the Shelf [17] dataset. Qualitative results on this datasets can be seen in Figure 5.5. As metric we use percentage of correct parts (PCP) in 3D [27] and we adopt the head position alignment utilized in [51] as well as the temporal Gaussian smoothing described in [183]. Furthermore, we report PCP averaged over the actors (aAvg) and PCP averaged over the actors weighted by the number of visible frames (gAvg), which was first discussed in [19]. Weighting by the number of visible frames (gAvg) provides a more accurate measure as it does not overemphasize actors which appear only in very few frames. Table 5.1 presents our results. For the the Shelf dataset we achieve state-of-the-art results while we achieve highly competitive results on the Campus dataset. We argue that the top-down pose estimation model and the appearance model of [51] are beneficial when the full bodies are visible and the scenes are relatively uncluttered, as it is the case with the Campus dataset (Figure 5.5 top row). However, in more complex scenes where bodies are only partially visible and with large background clutter and occlusions, such as Shelf, the appearance model does not help as much. Here, temporal information is crucial to recover from occlusions.

5.3.2 Tracking

For evaluating the tracking performance of our method, we utilize the MOTA [21] score as well as precision and recall. We cannot evaluate tracking on the Shelf or Campus dataset as some of the ground-truth annotations are missing, which results in a large number of false positives. Instead we evaluate on the CMU Panoptic studio [96], which utilizes the same human pose keypoints [117] as our method and which provides unique identifiers for each person in the scene. We use the sequence 160422_ultimatum1 from frames 300 to 1300 as in Chapter 4 since it contains different interacting

Method	MOTA	Precision	Recall	MOTA	Precision	Recall
	Average			Nose		
Chapter 4	0.82	91.0	91.1	0.84	91.7	91.8
Ours	**0.87**	**93.3**	**94.1**	**0.94**	**96.6**	**97.5**
	Left Wrist			Right Wrist		
Chapter 4	0.82	**91.2**	91.3	0.86	**93.0**	93.1
Ours	**0.83**	91.1	**91.9**	0.86	92.6	**93.4**
	Left Foot			Right Foot		
Chapter 4	0.81	90.5	90.6	0.77	88.6	88.7
Ours	**0.90**	**94.6**	**95.5**	**0.84**	**91.5**	**92.3**

Table 5.2: Tracking scores MOTA [21], precision and recall for sequence 160422_ultimatum1 of the CMU Panoptic Studio [96].

persons that enter and leave the scene. A sample scene can be seen in Figure 5.6. To ensure occlusions, we utilize only the first three hd-cameras and we consider a track as correct if its prediction is within 10cm of the ground-truth. For measuring the tracking accuracy, we utilize the nose, left/right wrist and left/right foot. Our results are presented in Table 5.2. We observe that our model significantly outperforms Chapter 4 for feet and nose since these keypoints are for some frames only visible in one camera as shown in Figure 5.6. Our method can recover these cases.

5.3.3 Ablation

Our ablation results are presented in Table 5.3. Removing tracking and only using the pose initialization algorithm described in Section 5.2.3 at each frame results in very strong results for the Shelf dataset while the performance drops significantly for the Campus dataset. The reason for this is that the pose initialization works better when multiple views are present (5 for Shelf, 3 for Campus) while tracking helps when a person is temporally visible in only one or two views. Removing pose resampling during the update step and instead using a fixed set of samples for each camera pair results in a significant performance drop. One of the biggest factors for the strong performance of our method is the joint refinement as the sample-based representation of 3D poses does not permit enough samples to accurately represent such high dimensional data. Removing the pose prediction model and just utilizing a zero velocity model also results in a significant performance loss. Replacing the zero-velocity model with a normal distribution for pose prediction does not significantly improve the results. Replacing the heuristic joint refinement algorithm described in Section 5.2.2 with a gradient ascent based algorithm results in a slight performance drop. We argue that the local optimization gets stuck in local optima while the heuristic can jump over them and find even better pose configurations.

5.3.4 Parameters

The effects of the hyperparameters are shown in Figure 5.7. The Dropout rates of both the prediction model and the initialization model g are determined to obtain a reasonable approximation of uncer-

	Campus					Shelf				
	A1(48)	A2(188)	A3(136)	aAvg	pAvg	A1(279)	A2(37)	A3(161)	aAvg	pAvg
only Pose Initialization	91.85 ± 1.33	92.94 ± 0.22	69.96 ± 0.95	84.92 ± 0.68	84.40 ± 0.46	99.51 ± 0.14	94.03 ± 0.83	97.69 ± 0.05	97.07 ± 0.31	98.47 ± 0.13
w/o Pose Resampling	87.29 ± 7.86	90.57 ± 0.33	88.27 ± 5.43	88.71 ± 3.74	89.31 ± 2.62	97.47 ± 1.10	88.95 ± 1.40	97.83 ± 0.09	94.75 ± 0.70	96.93 ± 0.70
w/o Joint Refinement	47.33 ± 24.53	74.78 ± 14.91	57.52 ± 5.16	59.88 ± 11.89	64.93 ± 9.81	90.96 ± 1.95	73.11 ± 5.39	91.89 ± 1.92	85.32 ± 2.24	89.89 ± 1.27
w/o Pose Prediction	70.56 ± 12.91	82.93 ± 6.40	73.19 ± 5.22	75.56 ± 4.23	77.77 ± 4.01	99.16 ± 0.04	65.22 ± 11.09	97.73 ± 0.03	87.37 ± 3.70	96.04 ± 0.86
Pose Prediction : $\mathcal{N}(0, 0.01^2)$	75.83 ± 27.52	80.84 ± 6.47	70.49 ± 10.29	75.72 ± 11.99	76.41 ± 8.38	99.12 ± 0.04	69.24 ± 12.09	97.71 ± 0.04	88.69 ± 4.03	96.33 ± 0.94
Pose Prediction : w/o dropout	90.42 ± 0.73	92.12 ± 0.12	97.28 ± 0.15	93.27 ± 0.27	93.79 ± 0.15	99.29 ± 0.04	94.78 ± 1.09	97.76 ± 0.00	97.28 ± 0.37	98.43 ± 0.09
Joint Refinement : Gradient Ascent	96.15 ± 0.10	92.34 ± 0.11	97.13 ± 0.22	95.21 ± 0.07	94.58 ± 0.12	99.40 ± 0.05	93.92 ± 0.68	97.80 ± 0.03	97.04 ± 0.23	98.43 ± 0.07
Proposed	97.35 ± 0.40	93.44 ± 0.04	97.43 ± 0.18	96.07 ± 0.13	95.40 ± 0.07	99.49 ± 0.06	95.81 ± 0.37	97.83 ± 0.00	97.71 ± 0.13	98.64 ± 0.05

Table 5.3: Ablation study using percentage of correctly estimated parts (PCP) on the Campus and Shelf datasets. *A1* to *A3* represent the three actors while the number in parentheses represents the number of ground-truth frames. We report both actor-wise (*aAvg*) as well as global average (*gAvg*) PCP.

	2 Actors	3 Actors	4 Actors
Chapter 4	0.023s	0.045s	0.104s
Ours	0.062s	0.088s	0.114s

Table 5.4: Time analysis for the Shelf dataset with respect to the number of actors. The time for the prediction and update steps of our method are measured with 300 sampled 3D poses per person and 50 sampled points for joint refinement.

tainty. If it is too small, the uncertainty is underestimated. For large values, the generated samples are too diverse, making the approach inefficient. The number of pose samples is important to ensure a sufficient representation of the pose distribution. However, a too high number of pose samples impedes sometimes the discovery of newly appearing persons and thus degrades the overall quality of the results. The distance threshold ε_j of the hierarchical clustering for merging joints influences the quality of the triangulated 3D joint positions to initialize poses. While a high value ε_j merges 3D joints of different persons, more redundant 3D joints would remain with a lower threshold. Using many samples for joint refinement encourages that each joint is located in regions with high part confidences. When the number is too large, it reduces the variety of the samples which weakens the tracking quality. Similarly, a high standard deviation for the joint refinement allows to search a large 3D space for each joint. If it is too large, the joints might move to the wrong position.

5.3.5 Runtime Analysis

In Table 5.4 we compare the runtime of the approach in Chapter 4 with our approach on the same machine, using an Intel Core i7-7700 3.60GHz and a Nvidia GeForce 1080ti. We evaluate the run-

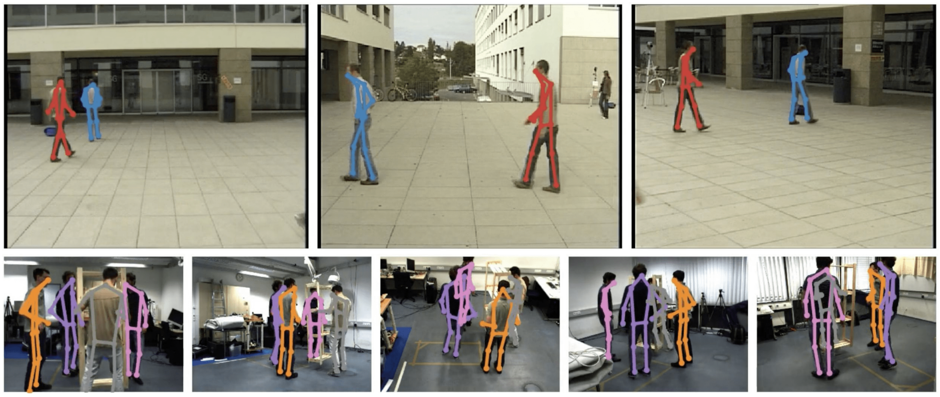


Figure 5.5: Qualitative results from the Campus [17, 60] (top row) and Shelf [17] (bottom row) dataset.

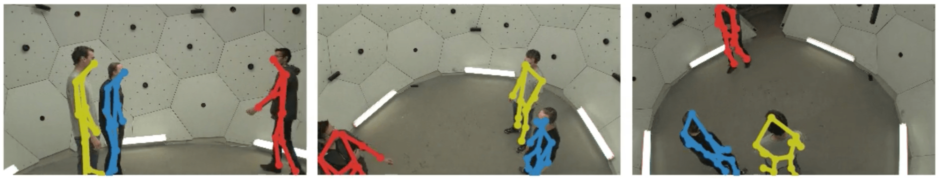


Figure 5.6: Qualitative results showing the first three hd-cameras of the CMU Panoptic studio [96].

time on the Shelf dataset, which uses five cameras and which has 2, 3 or 4 persons in the scene. Both Chapter 4 and our method are implemented in Python, utilizing the output of the official OpenPose [32] implementation which processes an image in around $35\mathrm{ms}$. While our approach needs more time than Chapter 4 for two actors, the runtime scales better as the number of actors increases. While the runtime of Chapter 4 increases quadratically as the number of actors increases, our approach requires $26\mathrm{ms}$ for each additional actor, i.e. the runtime increases linearly as the number of actors increases.

5.4 Conclusion

In this paper we have presented a novel tracking algorithm based on the well-known recursive Bayesian filtering framework and on recent advancements in 2D human pose estimation and 3D human motion anticipation. Our approach tracks multiple persons, initializes newly appearing persons, and recovers occluded joints. Our approach achieves state-of-the-art results for 3D human pose estimation as well as for 3D human pose tracking. In the future our approach could be extended using an appearance model similar to [51]. Furthermore, we could include a smoothing step which would improve 3D pose predictions backwards through time, utilizing the model uncertainty.

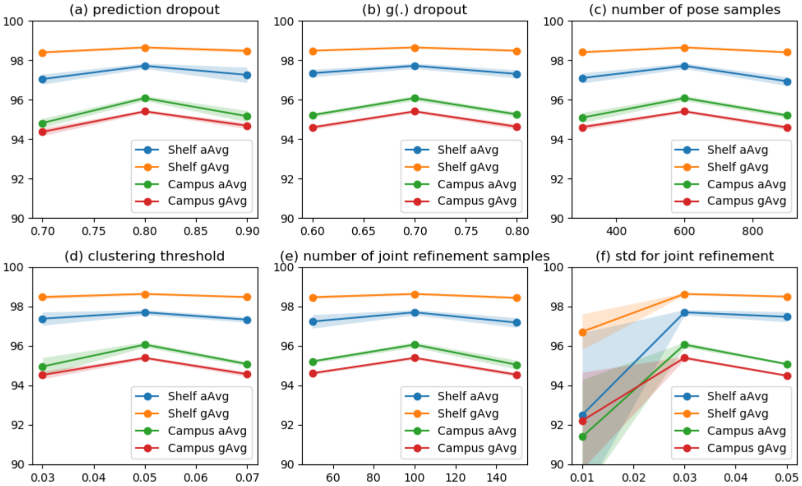


Figure 5.7: Evaluation of hyperparameters. PCP is evaluated while varying the hyperparameters. With each setting, the experiments are performed 10 times. The solid line indicates the mean value of the PCP and the colored area is the coverage determined by the standard deviation. (a) The dropout rate of the prediction model. (b) The dropout rate of the model $g(\cdot)$. (c) The number of pose samples. (d) Distance parameter to merge joints using the hierarchical clustering. (e) The number of samples for joint refinement. (f) Standard deviation for joint refinement.

CHAPTER 6

Intention-based Long-Term Human Motion Anticipation

The previous two chapters introduce human pose estimation from multiple calibrated cameras which can then be used for downstream tasks. One such task is human motion forecasting where given an input of past human motion the future poses are predicted, which is important for human-robot interaction. In this chapter we introduce Intention RNN, a recurrent neural network that first forecasts a discrete per-frame latent code, which we term *intention*, and then forecasts the 3D human motion based on the humans forecast intent. This allows us to forecast diverse but sensible human motion for much longer time horizons than previous works were capable of. We further introduce a novel human motion evaluation score which closely follows human supervision and which can judge the motion quality of sequences of arbitrary length.

Individual Contribution: The following chapter is based on the publication [185]:

Intention-based Long-Term Human Motion Anticipation
Julian Tanke, Chintan Zaveri and Juergen Gall
IEEE International Conference on 3D Vision (3DV), 2021.

This publication was done in very close collaboration between Chintan Zaveri and Julian Tanke. This publication is based on the Master thesis of Chintan Zaveri who was advised and supported by Julian Tanke. Juergen Gall provided scientific guidance and supported this work with very valuable feedback and suggestions. The idea to utilize discrete forecast labels was proposed by Juergen Gall and the network architecture, experiment setup and novel *Normalized Directional Motion Similarity* (NDMS) score were proposed by Julian Tanke. The Intention RNN implementation was done by Chitan Zaveri with supervision from Julian Tanke while the implementation of NDMS was done by Julian Tanke. The paper was written by Julian Tanke.

Contents

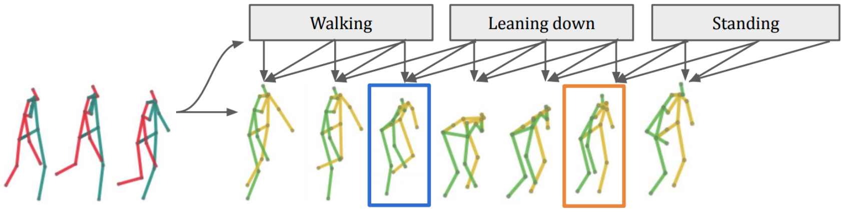


Figure 6.1: Intention-based human motion anticipation. Given a human motion input sequence (red-blue skeletons), our method forecasts the intention of the person ahead of time (top row) and the human motion (green-yellow skeletons) conditioned on the previous motion and the future intention. This allows not only long-term forecasting but also realistic transitions between different actions. For example, the blue and orange boxes show how the motion already prepares for the next action *leaning down* or *standing*, respectively.

6.1 Introduction

Anticipating human motion is highly relevant for many interactive activities such as sports, manufacturing, or navigation [147] and significant progress has been made in forecasting human motion [30, 61, 65, 68, 93, 113, 132, 153, 199]. Most progress has been made in anticipating motion over a short time horizon of around half a second. However, these methods fail when anticipating longer time horizons as they either produce unrealistic poses or the motion freezes. Another issue that occurs when the time horizon gets larger is the fact that there are more than one future sequence that are plausible for a single observed sequence of human motion as it is shown in Figure 6.2. Going from a short time horizon of less than one second to a larger time horizon of a few seconds therefore imposes the following challenges: (a) How can we model the uncertainty of the future? (b) How can we ensure that the motion remains plausible? (c) How can we measure the quality of methods that generate more than one sequence?

Handling the uncertainty of the future has been so far only addressed in very few recent works [10, 156, 212] for human motion anticipation. These approaches are able to forecast diverse sequences from the same observation, but the quality of the sequences decreases for longer time horizons beyond 1 second. In this work, we also propose a network that generates multiple sequences as shown in Figure 6.2, but our goal is to generate more plausible sequences for time horizons of

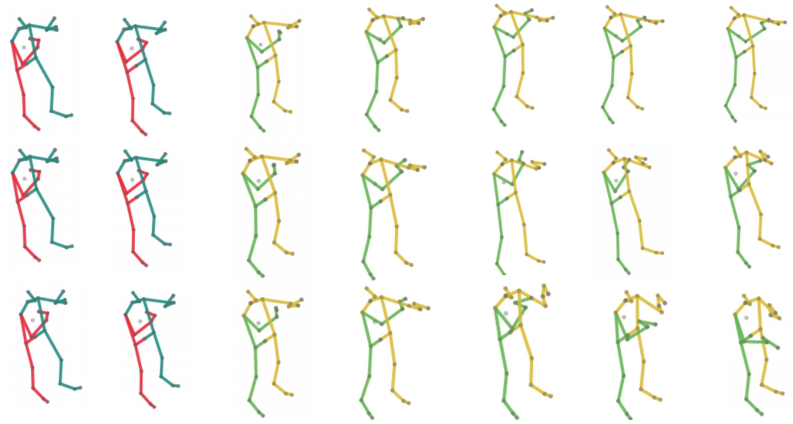


Figure 6.2: Our approach forecasts multiple sequences of plausible future human motion for long time horizons. Each row shows a different prediction of three seconds made by our model, given the same input sequence (*Discussion* from Human3.6M [85]). The red-blue skeletons represent the ground-truth input while the green-yellow skeletons are model predictions. During the first second, the model generates fairly consistent human poses but it starts to generate diverse but realistic human motion after 1 second. The qualitative results are best viewed in the supplementary video.

multiple seconds. In order to achieve this goal, we not only model the human motion but also the intention of the person as illustrated in Figure 6.1. In fact, human motion anticipation depends on two factors, namely the past motion and the intention. The latter, which is ignored by existing works, is very important for longer sequences since a motion without a goal is perceived as random and unrealistic. We therefore model the intention as discrete actions and propose to forecast the intention as well as the human motion. The key aspect is that our model forecasts the intention ahead of time and that the forecast human motion is conditioned on the past motion and on the forecast intention as shown in Figure 6.1.

It, however, remains an open issue how methods that generate multiple sequences are best compared. Recent works suggest to evaluate both the quality of the generated motion as well as the sample diversity. While diversity is commonly measured by using the average pairwise distance between multiple generated predictions [10, 212], measuring the quality is still an open problem. In [212], for instance, multiple sequences are forecast but only the error of the sequence with the lowest error is reported. Such measures are misleading since they evaluate only one forecast sequence while the other sequences can be implausible. In fact, we show in Section 6.4.4.3 that this measure can be easily fooled by a simple but unrealistic baseline approach, yielding competitive results on clearly unrealistic motion. In [10], pre-trained skeleton-based action classifiers are used to compute the inception score and a quality score over all generated sequences. While the inception score is an indicator for plausibility it is highly depended on the model. The authors did not make the models publicly available, making an evaluation very difficult. Normalized Power Spectrum Similarity [65] (NPSS) evaluates sequences in the power spectrum to account for frequency shifts that cannot be captured by MSE. However, NPSS is uni-modal as it compares the motion to a single ground-truth sequence. We therefore propose a new complementary similarity score that measures

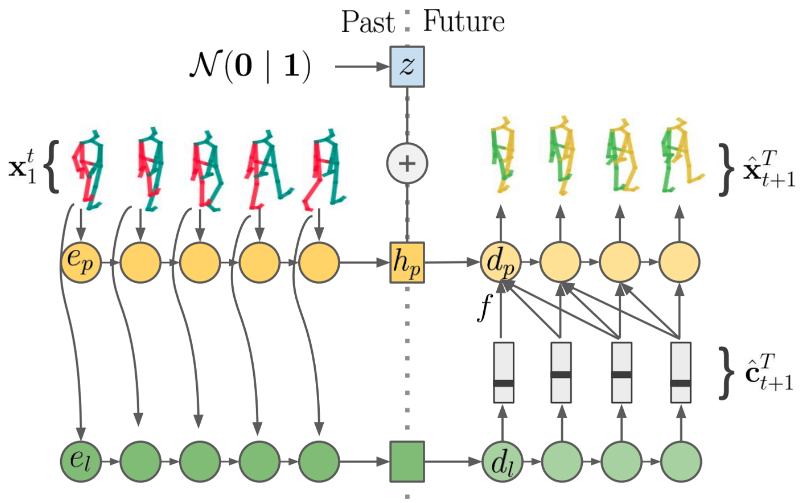

Past
Future
$\mathcal{N}(\mathbf{0} \mid \mathbf{1})$
z
$+$
$\mathbf{x}_1^t$
e_p
h_p
d_p
$\hat{\mathbf{x}}_{t+1}^T$
f
$\hat{\mathbf{c}}_{t+1}^T$
e_l
d_l

Figure 6.3: Overview of our method. The blue-red skeletons are the observed human poses while the yellow-green skeletons are forecast future human poses. The network forecasts the human poses at two levels: at the pose level (yellow) and at an intention level (green). During inference, the network forecasts the intention labels ahead in time which guide then the generation of the future poses. By conditioning the pose decoder d_p in addition on z, multiple plausible sequences can be generated for a single sequence of observed human poses.

the normalized directional motion similarity between motion snippets of forecast and real motions that have the same semantic meaning. The measure has the advantage that it takes the multi-modality of human motion into account and that it correlates much better with human perception than NPSS.

Our contribution is therefore two-fold:

- We propose a novel quality score for long-term human motion anticipation that measures the plausibility of multiple generated sequences and that correlates better with human perception than other metrics.
- We propose a novel approach for human motion forecasting that forecasts the intention of a person ahead of time and that is capable of generating multiple plausible future sequences for long time horizons.

6.2 Stochastic Human Motion Anticipation from Intention

In this work, we address the task of forecasting human motion. This means that we observe 3d human skeletons for t frames, which are denoted by $\mathbf{x}_1^t = (x_1, \ldots, x_t) \in \mathbb{R}^{t \times \mathfrak{d}}$ and where $\mathfrak{d}$ is the feature dimension that represents the human pose, and our goal is to forecast plausible future pose sequences $\hat{\mathbf{x}}_{t+1}^T \sim p(\mathbf{x}_{t+1}^T | \mathbf{x}_1^t)$ where $\hat{\mathbf{x}}_{t+1}^T = (\hat{x}_{t+1}, \ldots, \hat{x}_T) \in \mathbb{R}^{(T-t) \times \mathfrak{d}}$ and $p(\mathbf{x}_{t+1}^T | \mathbf{x}_1^t)$ is the distribution of all plausible future sequences given the observed human motion.

As it is illustrated in Figure 6.2, our approach does not predict a single sequence but aims to learn the distribution $p(\mathbf{x}_{t+1}^{T}|\mathbf{x}_{1}^{t})$ such that we can generate multiple plausible future sequences

$$\hat{\mathcal{X}}_{t+1}^{T} = \{\hat{\mathbf{x}}_{t+1}^{T} : \hat{\mathbf{x}}_{t+1}^{T} \sim p(\mathbf{x}_{t+1}^{T}|\mathbf{x}_{1}^{t})\}. \tag{6.1}$$

While we introduce in Section 6.4.4 a new quality score that evaluates the plausibility of the set $\hat{\mathcal{X}}_{t+1}^{T}$ and that correlates very well with human perception, we first discuss the novel approach that forecasts (6.1).

Although the recent works [10, 156, 212] are able to forecast diverse sequences, the quality of the sequences decreases for longer time horizons beyond 1 second as we show in the user study reported in Table 7.6. This is expected since the methods model human motion but not the intention of the person. The latter, however, is very important for longer sequences since a motion without a goal is perceived as random and unrealistic.

We therefore propose an approach that generates multiple future sequences that remain plausible even for longer time horizons of 4 seconds. In order to achieve this goal, our network not only forecasts human poses, but also the intention as shown in Figure 6.1 and 6.3. An important aspect of our network is that it forecasts the intention $\hat{\mathbf{c}}_{t+1}^{T}$ ahead in time, which then guides the generated poses

$$\hat{\mathbf{x}}_{t+1}^{T} \sim p(\mathbf{x}_{t+1}^{T}|\mathbf{x}_{1}^{t}, \hat{\mathbf{c}}_{t+1}^{T}) \tag{6.2}$$

and ensures plausible motion transitions when the intention changes. We describe the module of the network that forecasts the intention in Section 6.2.1 and the module that forecasts the human motion conditioned on the intention in Section 6.2.2.

6.2.1 Intention Anticipation

We model the intention by a categorical representation $c_t \in C$ where C is set of possible intention classes. While we forecast the intention ahead in time as shown in Figure 6.1, we estimate it for each future frame $\hat{\mathbf{c}}_{t+1}^{T}$. In Section 8.2.1.2, we describe how C can be obtained in an unsupervised way.

To anticipate future intent, we use a recurrent encoder-decoder where the recurrent encoder e_l takes as input a sequence of observed human motion $\mathbf{x}_1^t$ and the recurrent decoder d_l forecasts the future intentions $\hat{\mathbf{c}}_{t+1}^{T}$:

$$\hat{\mathbf{c}}_{t+1}^{T} = d_l(e_l(\mathbf{x}_1^t)). \tag{6.3}$$

With decoder d_l being auto-regressive, we are not constrained to a fixed time horizon and as such T can be as large as needed. We represent both e_l and d_l with single layer GRUs.

For training, we utilize the categorical cross-entropy as loss function:

$$\mathcal{L}_{\text{sym}} = \frac{1}{T-t}\sum_{\tau=t+1}^{T}\sum_{j=1}^{|C|} c_\tau \log(\hat{c}_{\tau j}) \tag{6.4}$$

where $|C|$ is the total number of discrete intention labels, c_τ denotes the reference label at time step τ, and $\hat{c}_{\tau j}$ denotes the predicted probability of the j-th class at time step τ. We will discuss in Section 8.2.1.2 how the reference labels c_τ are computed for the training set.

In order to generate plausible sequences of future intentions, we furthermore add an adversarial loss:

$$\begin{aligned}\mathcal{L}_{\text{sym}}^{\text{adv}} = \min_{d_l}\max_{D_{\text{label}}} &\, \mathbb{E}_{\mathbf{c}}\big[\log D_{\text{label}}(\mathbf{c})\big] \\ &+ \mathbb{E}_{\mathbf{x}}\big[1 - \log D_{\text{label}}(d_l(\mathbf{x}))\big]\end{aligned} \tag{6.5}$$

where $\mathrm{D}_{\text{label}}$ is a one-hidden-layer feed forward network.

6.2.2 Human Motion Anticipation

In order to sample sequences of future human poses from $p(\mathbf{x}_{t+1}^{T}|\mathbf{x}_1^t, \hat{\mathbf{c}}_{t+1}^{T})$, we utilize a conditional GAN [137] with normal distributed noise vector $z \sim \mathcal{N}(\mathbf{0}, \mathbf{1})$ as shown in Figure 6.3. It is conditioned on the past human motion sequence $\mathbf{x}_1^t$ and the forecast intent $\hat{\mathbf{c}}_{t+1}^{T}$.

Specifically, we first encode $\mathbf{x}_1^t$ into the vector h_p using the recurrent pose encoder e_p, *i.e.*, $h_p = e_p(\mathbf{x}_1^t)$. We then concatenate h_p and z and auto-regressively generate future poses for $t < \tau \leqslant T$ using the pose decoder d_p:

$$(\hat{x}_\tau, h_\tau) = d_p(\hat{x}_{\tau-1} \oplus f(\hat{\mathbf{c}}_\tau^{\tau-1+\gamma}) \mid h_{\tau-1}) \tag{6.6}$$

where $h_t = h_p \oplus z$, $\hat{x}_t = x_t$, and $\oplus$ denotes the concatenation of two vectors. The pose encoder e_p consists of a single layer GRU while the pose decoder d_p consists of a three layer GRU.

The pose decoder d_p, however, not only depends for each frame τ on the previous generated pose $\hat{x}_{\tau-1}$ and the previous hidden state $h_{\tau-1}$, but also on $f(\hat{\mathbf{c}}_\tau^{\tau-1+\gamma})$, *i.e.*, on the intention which is forecast already γ frames ahead. If $\gamma = 1$, the decoder does not look ahead and it takes only the estimated intention labels until the current frame into account. We will show in the experiments that this results in less plausible sequences since the decoder cannot prepare the transition between two types of motions if they change, *e.g.*, between leaning down and standing as shown in Figure 6.1. If we allow the decoder to look ahead, the transitions are more smooth and plausible. We found that $\gamma = 10$ $(0.4s)$ is sufficient to obtain good results. Before adding the probabilities $\hat{\mathbf{c}}_\tau^{\tau-1+\gamma}$ to the decoder, we aggregate them by f, which is a temporal convolutional layer with kernel size γ.

During training, we optimize the adversarial loss

$$\begin{aligned}\mathcal{L}_{adv} = \min_{d_p} \max_{D_{\text{pose}}} \ & \mathbb{E}_{\mathbf{x}}\big[\log D_{\text{pose}}(\mathbf{x})\big] \\ & + \mathbb{E}_{\mathbf{x}|\mathbf{c}|z}\big[1 - \log D_{\text{pose}}(d_p(\mathbf{x}, \mathbf{c}, z)\big]\end{aligned} \tag{6.7}$$

where D_{pose} is a two-hidden-layer feed forward network. While there is usually not a high variability of the plausible human motion directly after the last observed frame but the diversity increases the longer the time horizon gets as shown in Figure 6.2, we additionally utilize a reconstruction loss with decreasing impact as τ increases:

$$\mathcal{L}_{\text{rec}} = \frac{1}{J \cdot (T-t)} \sum_{\tau=t+1}^{T} \sum_{j=1}^{J} \lambda(\tau) ||x_{\tau j} - \hat{x}_{\tau j}||_2 \tag{6.8}$$

where J is the number of joints in the pose, and $x_{\tau j}$ and $\hat{x}_{\tau j}$ denote the ground truth and model prediction of joint j at time frame τ, respectively. The weight $\lambda(\tau)$ decreases linearly over time with $\lambda(t) = 1$ and $\lambda(t + \tau_{rec}) = 0$. In our experiments, we show that $\tau_{rec} = 15$ $(0.6s)$ is sufficient.

For training the network, we use all four loss terms where the loss terms $\mathcal{L}_{\text{sym}}$ (6.4) and $\mathcal{L}_{\text{sym}}^{\text{adv}}$ (6.5) supervise the intention forecasting (green) and the loss terms $\mathcal{L}_{adv}$ (6.7) and $\mathcal{L}_{\text{rec}}$ (6.8) supervise the human motion forecasting (yellow) as shown in Figure 6.3.

6.2.3 Intention Labels

In order to obtain the intention labels $c_t \in C$ for training, we cluster the training sequences. We first cluster the poses of all training sequences using k-means and assign each frame to a cluster. Since these clusters only consider poses but not motion, we sequentially generate intention labels by detecting cycles of cluster ids in the training sequences. For all datasets, we use 8 intention labels. More details are provided in the supplementary material where we also evaluate the impact of the size of C.

6.2.4 Implementation Details

The encoder e_l and decoder d_l in Figure 6.3 are both represented as single-layer GRUs with 16 hidden units. The decoder has an additional softmax layer to generate class probabilities. D_{label} is a feed-forward neural network with a single hidden layer with 32 units, a ReLU non-linearity, and a single unit sigmoid output. The input to D_{label} are the predicted output labels of d_l, stacked for 25 frames.

The pose encoder e_p and decoder d_p are represented as GRUs where e_p is a single-layer GRU with 512 hidden units and where d_p is a three-layer GRU with 512 hidden units and dropout rate 0.3. The encoder output h_p is passed as hidden state to the first decoder layer, while the remaining two layers are initialized with a zero hidden state. The noise state z is concatenated to all three hidden states before passing to the decoder. The pose discriminator D_{pose} is represented as one hidden layer feed-forward neural network with 512 hidden units, ReLU non-linearity and a single unit sigmoid output. The input to D_{pose} are the predicted output labels of d_p stacked for 25 frames. Each pose has dimension 54. The implementation of the NDMS metric and the source code for the approach are available at `https://github.com/jutanke/human_motion_ndms`.

6.3 Long-term Human Motion Quality Score

As discussed in Section 6.2, we need for evaluation a score that measures the plausibility of forecast human motion for longer time horizons beyond one second. Furthermore, the measure needs to measure the quality of a set of forecast sequences $\hat{\mathcal{X}}_{t+1}^{T}$ instead of a single sequence.

We therefore propose a novel quality measure that correlates better with human perception. The main idea is that a plausible sequence of poses should be close to a real sequence. For long-time horizons, however, the sequences are too long to compare them directly. Instead, we divide all sequences that have the same semantic meaning but that are not part of the training data into overlapping short motion sequences of fixed length κ. We call the short motion sequences *motion words* and we use $\kappa = 8$ for sequences with 25Hz. This results in a very large motion database $\mathcal{D}$.

When evaluating a sequence $\hat{\mathbf{x}}_{t+1}^{T} \in \hat{\mathcal{X}}_{t+1}^{T}$ for observation $\mathbf{x}_1^t$, we split the sequence into overlapping motion words as well, where we include the last $\kappa - 1$ observed frames, *i.e.*, $\hat{\mathbf{x}}_{t+2-\kappa}^{t+1}, \hat{\mathbf{x}}_{t+3-\kappa}^{t+2}, \ldots, \hat{\mathbf{x}}_{T+1-\kappa}^{T}$. We include the last observed frames such that the transition between observed and forecast motion is also taken into account. This is important since discontinuities between observed and forecast frames are perceived by humans as highly unrealistic. Using the motion words of all sequences of $\hat{\mathcal{X}}_{t+1}^{T}$, we can then compute the plausibility score by measuring the similarity of the motion words of $\hat{\mathcal{X}}_{t+1}^{T}$ with the motion words in $\mathcal{D}$:

$$f_{sim}\left(\hat{\mathcal{X}}_{t+1}^{T}\right) = \frac{1}{Z}\sum_{\hat{\mathbf{x}}_{t+1}^{T}\in\hat{\mathcal{X}}_{t+1}^{T}}\sum_{\tau=t+2-\kappa}^{T+1-\kappa} g\left(\hat{\mathbf{x}}_{\tau}^{\tau+\kappa},\mathcal{D}\right), \tag{6.9}$$

where $Z = (T-t)|\hat{\mathcal{X}}_{t+1}^{T}|$ is the normalization factor.

For computing the plausibility of a motion word, we find the closest motion word in $\mathcal{D}$ using nearest neighbor search (NN) and compute the normalized directional motion similarity (NDMS), which is discussed in Section 6.3.1:

$$g\left(\hat{\mathbf{x}}_{\tau}^{\tau+\kappa},\mathcal{D}\right) = \mathrm{NDMS}(\hat{\mathbf{x}}_{\tau}^{\tau+\kappa},\mathrm{NN}\big(\hat{\mathbf{x}}_{\tau}^{\tau+\kappa},\mathcal{D}\big)\big). \tag{6.10}$$

The function $g\left(\hat{\mathbf{x}}_{\tau}^{\tau+\kappa},\mathcal{D}\right)$ is 1 when $\mathcal{D}$ contains the exact motion word $\hat{\mathbf{x}}_{\tau}^{\tau+\kappa}$ and it is $0 \leqslant g\left(\hat{\mathbf{x}}_{\tau}^{\tau+\kappa},\mathcal{D}\right) < 1$ otherwise. Using motion words and not single poses ensures that the score evaluates motion quality and consistency and not just pose quality while the nearest neighbor approach ensures that the multi-modality of human motion is addressed. Due to the normalization factor Z, f_{sim} (6.9) provides a plausibility score between 0 and 1 for a set of forecast human motions.

6.3.1 Normalized Directional Motion Similarity

In order to compare two motion words x and y, we need to define a similarity measure. The Euclidean distance of the poses is insufficient as this favours sequences that remain close to the mean pose. Similarly, using the mean square error of the velocities favours small motion over larger motion, as we discuss in the supplementary material. Instead, we measure the similarity of the motion direction and the ratio of motion magnitudes.

Specifically, the proposed Normalized Directional Motion Similarity (NDMS) compares two motion words $\mathbf{x}, \mathbf{y}$ of length κ by

$$\mathrm{NDMS}(\mathbf{x},\mathbf{y}) = \frac{\sum_{t=1}^{\kappa-1}\frac{1}{J}\sum_{j=1}^{J}\Psi_t^j(\mathbf{x},\mathbf{y})}{\kappa-1} \tag{6.11}$$

$$\begin{aligned}\Psi_t^j(\mathbf{x},\mathbf{y}) = \frac{1}{2}&\left(1+\frac{\dot{\mathbf{x}}_{t,j}\dot{\mathbf{y}}_{t,j}^{T}}{||\dot{\mathbf{x}}_{t,j}||\cdot||\dot{\mathbf{y}}_{t,j}||+\varepsilon}\right)\\ &\cdot\frac{\min\left(||\dot{\mathbf{x}}_{t,j}||,||\dot{\mathbf{y}}_{t,j}||\right)}{\max(||\dot{\mathbf{x}}_{t,j}||,||\dot{\mathbf{y}}_{t,j}||)+\varepsilon}\end{aligned} \tag{6.12}$$

where J represents the numbers of joints of the human pose and $\dot{\mathbf{x}}_{t,j}$ is the 3D velocity of joint j at time t. The first part in (6.12) yields large values when the j-th joint of x and y move in the same direction while the second part yields large values when the magnitudes of the vectors are similar. To prevent division by zero, we add a small $\varepsilon > 0$. This way, $\Psi_t^j(\mathbf{x},\mathbf{y})$ produces values close to 1 when the motions of $\mathbf{x}$ and $\mathbf{y}$ are similar and values close to 0 when they are very dissimilar.

It is important to note that the proposed quality measure (6.10) has several advantages compared to existing measures: a) the measure penalizes discontinuities in motion; b) it penalizes unrealistic motion at a fine-grained level; c) it can be used to measure the quality of deterministic as well as stochastic approaches; d) it measures the plausibility of all forecast sequences even if they deviate from the observed future sequence; e) it correlates better than other measures with human perception.

Long-Term					
method	walking	eating	smoking	discussion	average
Seq2Seq [132]	0.549	0.754	1.403	1.245	0.987
[65]	**0.359**	**0.288**	0.577	1.001	0.556
Trajectory [129]	0.841	0.909	0.824	1.733	1.077
History [130]	0.590	0.821	0.491	1.616	0.879
Grammar [156]	0.467	0.301	0.751	0.945	0.616
Ours	0.367	0.621	**0.363**	**0.795**.	**0.536**

Table 6.1: NPSS measure from [65] for long-term motion anticipation.

6.3.2 Implementation Details

For the nearest neighbor search, we use the joint positions of *wrists*, *elbows*, *shoulders*, *hips*, *knees*, and *ankles*. For evaluation, we populate $\mathcal{D}$ with all relevant test sequences, *e.g.*, all *basketball* test sequences for evaluating *basketball*. This way, models have to produce sequences that have the same semantic meaning as the current test set (*e.g.walking*, *eating*) and not just produce common motion patterns observed in all sequences.

6.4 Experiments

We evaluate our method on the two standard large scale motion capture datasets: Human3.6M [85] and CMU Mocap [1]. We first analyze the quality of the forecast sequences using different measures including a user study for long-term forecasting. We evaluate both long- and short-term motion forecasting.

6.4.1 Comparison to State-of-the-Art

Long-Term Forecasting: For evaluating long-term human motion forecasting, we first report NPSS as described in [65], utilizing the publicly available implementation. The results of the long-term time scale of $2-4$ seconds can be seen in Table 6.1 where our method slightly outperforms current state-of-the art methods. Grammar [156] achieves competitive results. We will, however, later show that the sequences that are generated by Grammar are less realistic than the sequences of other state-of-the-art methods. This indicates that NPSS is not a very reliable measure for the plausibility of the forecast human motion.

We therefore compare the methods using the proposed NDMS metric (see Section 6.4.4) with motion word size $\kappa = 8$ on Human3.6M. For each of the 15 actions in Human3.6M, we calculate the scores independently where we populate the database $\mathcal{D}$ with the test sequences of the given action only - to ensure that the forecast sequences are semantically meaningful and consistent with the action. The results for up to 4 seconds are reported in Tables 6.2 and 6.3.

The results in Table 6.2 show that our approach outperforms stochastic and deterministic methods in terms of quality. On cyclic motion such as walking, [130] produces very strong results over long time periods. However, the motion freezes on non-periodic motion such as *discussion* and *posing*. As expected, other approaches including deterministic approaches [132, 129] perform fairly well for short sequences up to 1.2 seconds. For such short time horizon, the results are quite similar to our

seconds:	0.4	0.8	1.2	1.6	2.0	2.4	2.8	3.2	3.6	4.0	0.4	0.8	1.2	1.6	2.0	2.4	2.8	3.2	3.6	4.0
	walking										eating									
Seq2Seq [132]	0.891	0.660	**0.673**	**0.670**	**0.647**	0.617	0.562	0.537	0.515	0.418	0.794	0.517	0.504	0.514	0.483	0.493	0.428	0.428	0.395	0.409
Trajectory [129]	0.899	0.657	0.605	0.597	0.580	0.463	0.494	0.519	0.461	0.453	0.877	0.622	**0.567**	**0.560**	**0.549**	**0.543**	0.505	**0.565**	**0.555**	0.530
History [130]	**0.929**	**0.694**	0.647	0.668	0.641	**0.657**	**0.638**	**0.646**	**0.646**	**0.630**	**0.884**	0.460	0.418	0.426	0.405	0.406	0.392	0.415	0.394	0.395
Grammer [156]	0.839	0.429	0.413	0.415	0.315	0.291	0.268	0.186	0.172	0.131	0.826	0.421	0.347	0.315	0.227	0.210	0.192	0.167	0.166	0.171
Mix&Match [10]	0.847	0.645	0.607	0.625	0.573	0.564	0.541	-	-	-	0.830	0.534	0.524	0.534	0.506	0.494	0.474	-	-	-
Ours	0.902	0.647	0.619	0.630	0.586	0.592	0.574	0.604	0.577	0.603	0.878	**0.625**	0.530	0.560	0.520	0.521	**0.531**	0.525	0.546	**0.534**
	smoking										discussion									
Seq2Seq [132]	0.685	**0.491**	0.446	**0.422**	0.426	0.432	0.420	0.380	0.365	0.367	0.833	0.503	0.484	**0.472**	0.404	0.415	0.430	0.399	0.340	0.300
Trajectory [129]	0.824	0.476	**0.468**	0.411	0.406	0.381	0.415	0.402	0.390	0.390	0.852	0.417	0.361	0.314	0.302	0.293	0.272	0.290	0.267	0.273
History [130]	**0.874**	0.455	0.389	0.404	0.404	0.394	0.388	0.399	0.397	0.367	**0.893**	0.418	0.318	0.317	0.289	0.290	0.267	0.279	0.262	0.276
Grammer [156]	0.659	0.229	0.221	0.201	0.182	0.179	0.171	0.179	0.183	0.185	0.758	0.187	0.162	0.155	0.162	0.198	0.171	0.191	0.189	0.164
Mix&Match [10]	0.646	0.420	0.429	0.419	0.408	0.404	0.411	-	-	-	0.799	0.481	0.449	0.428	0.411	0.395	0.380	-	-	-
Ours	0.800	0.474	0.456	0.412	**0.446**	**0.477**	**0.452**	**0.433**	**0.445**	**0.417**	0.838	**0.503**	**0.497**	0.466	**0.474**	**0.467**	**0.492**	**0.498**	**0.463**	**0.449**
	posing										average									
Seq2Seq [132]	0.819	0.497	0.443	0.403	0.381	0.367	0.335	0.313	0.296	0.285	0.806	0.530	**0.508**	0.478	0.445	0.438	0.414	0.383	0.361	0.339
Trajectory [129]	0.827	0.476	0.454	0.374	0.406	0.328	0.303	0.317	0.281	0.307	0.840	0.485	0.452	0.391	0.389	0.364	0.344	0.367	0.334	0.338
History [130]	**0.896**	0.419	0.343	0.317	0.246	0.224	0.218	0.211	0.204	0.198	**0.884**	0.434	0.359	0.350	0.337	0.326	0.320	0.318	0.315	0.309
Grammer [156]	0.782	0.267	0.246	0.213	0.224	0.220	0.208	0.202	0.209	0.160	0.746	0.262	0.245	0.236	0.212	0.205	0.202	0.190	0.191	0.181
Mix&Match [10]	0.777	**0.528**	0.486	0.453	0.421	0.402	0.378	-	-	-	0.770	0.500	0.480	0.465	0.444	0.430	0.419	-	-	-
Ours	0.726	0.509	**0.539**	**0.471**	**0.478**	**0.430**	**0.421**	**0.439**	**0.407**	**0.393**	0.826	**0.531**	0.507	**0.491**	**0.484**	**0.478**	**0.474**	**0.477**	**0.467**	**0.465**

Table 6.2: NDMS scores on Human3.6M [85] for actions *walking*, *eating*, *smoking*, *discussion* and *posing* as well as averaged over all 15 actions. For Mix-and-Match and our approaches we report the mean score over 50 samples for a given input sequence.

seconds:	0.4	0.8	1.2	1.6	2.0	2.4	2.8	3.2	3.6	4.0	0.4	0.8	1.2	1.6	2.0	2.4	2.8	3.2	3.6	4.0
	walking										eating									
VAE [212]	0.589	0.413	0.354	0.329	0.322	0.321	0.320	0.320	0.313	0.299	0.521	0.302	0.322	0.326	0.314	0.310	0.310	0.308	0.298	0.291
DLow [212]	0.586	0.422	0.376	0.345	0.346	0.344	0.331	0.325	0.312	0.296	0.519	0.288	0.315	0.318	0.320	0.320	0.320	0.320	0.305	0.292
Ours	**0.906**	**0.653**	**0.636**	**0.584**	**0.608**	**0.577**	**0.558**	**0.570**	**0.556**	**0.553**	**0.818**	**0.536**	**0.457**	**0.454**	**0.406**	**0.388**	**0.356**	**0.376**	**0.368**	**0.352**
	smoking										discussion									
VAE [212]	0.455	0.295	0.324	**0.334**	**0.330**	0.321	0.310	**0.303**	**0.295**	0.288	0.536	0.334	0.333	0.306	0.288	0.281	0.274	0.276	**0.264**	0.245
DLow [212]	0.454	0.280	0.324	0.322	0.315	0.308	0.297	0.294	0.288	0.280	0.536	0.331	**0.345**	**0.334**	0.315	0.307	**0.291**	0.270	0.257	0.241
Ours	**0.754**	**0.348**	**0.356**	0.319	0.327	**0.345**	**0.310**	0.294	0.286	**0.322**	**0.859**	**0.470**	0.340	0.331	**0.336**	**0.320**	0.283	**0.293**	0.262	**0.274**
	posing										average									
VAE [212]	0.519	0.355	0.334	0.280	0.260	0.264	0.265	0.261	0.246	0.234	0.542	0.342	0.331	0.309	0.294	0.290	0.288	0.286	0.277	0.265
DLow [212]	0.521	0.367	**0.360**	0.332	0.315	0.291	0.271	0.261	0.260	0.241	0.541	0.342	0.341	0.325	0.311	0.298	0.290	0.282	0.274	0.263
Ours	**0.724**	**0.398**	0.301	**0.353**	**0.366**	**0.320**	**0.324**	**0.312**	**0.309**	**0.297**	**0.818**	**0.444**	**0.379**	**0.380**	**0.366**	**0.349**	**0.330**	**0.328**	**0.320**	**0.320**

Table 6.3: NDMS scores on Human3.6 [85] using the 17 3D joint representation from DLow [212]. We report the mean score over 50 samples for a given input sequence.

approach. However, for longer time horizons the benefit of forecasting the intention becomes evident and our approach outperforms the other methods by a large margin.

Since DLow [212] uses a different skeleton representation than the other methods, we also report the NDMS score for the skeleton from [212] in Table 6.3. On average, our approach outperforms DLow and a variational autoencoder (VAE). It needs to be noted that both DLow and VAE suffer from a motion discontinuity between the observed frames and the forecast frames. The NDMS score is therefore relatively low for the shortest time horizon (0.4s).

To validate our results, we conducted a user study with 28 individuals who were given random sequences of length of 4 seconds. The users had then to rate each sequence whether it was realistic or not. Our results can be seen in Table 7.6. When we compare, for instance, the results for *walking* to the results reported in Tables 6.2 and 6.3, we observe a high similarity between the human perception and the NDMS metric. For a cyclic motion like *walking*, our approach performs best followed by [129] and [130]. The generative grammars [156] start showing unrealistic artifacts after around 2 seconds of walking, which is captured both by our user study as well as by our evaluation score. DLow [212] performs better than generative grammars, but by far worse than the other methods. This is due to the discontinuity at the beginning, which is more prominent for walking than for the other activities where the person often stands at the beginning, but also due to the very high diversity

	walking	eating	smoking	discussion
Seq2Seq [132]	0.750	0.353	0.312	0.188
Trajectory [129]	0.903	0.625	0.114	0.227
History [130]	0.902	0.221	0.356	0.279
Grammer [156]	0.161	0.324	0.167	0.171
Mix&Match [10]⋆	0.875	0.617	0.445	0.523
DLow [212]	0.190	0.578	0.449	0.428
Ours	**0.938**	**0.792**	**0.633**	**0.714**

Table 6.4: User study for the results on Human3.6M [85]. 28 users were randomly asked to judge 4 seconds of forecast human motion. The users could only choose between *realistic* or *not realistic* where we count realistic as 1 and not realistic as 0. In the table we report the mean values and sequences valued close to 1 are deemed highly realistic. ⋆ indicates sequence length of 3.2 seconds.

Diversity (Human3.6M)					
[206]	[197]	[16]	Mix&Match [10]	DLow [212]	Ours
0.26	1.70	0.48	3.52	**4.71**	3.07
Diversity (CMU)					
0.41	**3.00**	0.43	2.63	2.90	2.40

Table 6.5: Average pairwise distance (APD) of recent state-of-the-art methods on Human3.6M [85] and CMU [1]. Results for DLow [212] are taken from [9].

of the generated sequences. The forecast sequences quickly generate motions that are very unlikely to occur after a walking motion. Indeed, Table 6.5 shows that [10, 212] have a higher diversity among the forecast sequences, but more of the generated sequences are perceived as unrealistic as shown in Table 7.6. The results in Tables 6.2, 6.3, and 7.6 show that our approach achieves a higher forecast quality than the state-of-the-art for long time horizons, both for cyclic as well as non-cyclic motions.

As already mentioned, the generative grammars [156] achieve competitive NPSS scores, as can be seen in Table 6.1. This, however, is not supported by our user study and the qualitative results. This shows the weakness of NPSS, which does not occur for the proposed NDMS score. The Pearson Correlation Coefficient of NPSS to the ground truth is -0.238 while our motion similarity scores a correlation of 0.901.

Short-Term Forecasting: To evaluate short-term motion prediction, we follow the same protocol as described in [10]. Table 6.6 and Table 6.7 detail our short-term forecasting results on Human3.6M [85] and on CMU Mocap [1], respectively. Even though our main objective is long-term and not short-term human motion anticipation, our approach achieves competitive results.

6.4.2 Ablation Study

Impact of Loss Functions: In Figure 6.4, we show the impact of the loss functions. While the blue curve corresponds to the proposed method, the red curve ($\mathcal{L}_{\text{adv}} + \mathcal{L}_{\text{rec}}$) is a special case in which we do not forecast the intention. The plots show that the NDMS scores are much lower when we do not forecast the intent. This is in particular visible for walking and eating. Without intent the model always converges to a mean motion, which, in Human3.6M, is standing and gesticulating with hands. Because of this, using no intent performs competitively only on Discussion, which is mostly made

	walking					eating					smoking					discussion				
milliseconds:	80	160	320	400	560	80	160	320	400	560	80	160	320	400	560	80	160	320	400	560
Zero Velocity [132]	0.39	0.86	0.99	1.15	1.35	0.27	0.48	0.73	0.86	1.04	0.26	0.48	0.97	0.95	1.02	0.31	0.67	0.94	1.04	1.41
Seq2Seq [132]	0.28	0.49	0.72	0.81	0.93	0.23	0.39	0.62	0.76	0.95	0.33	0.61	1.05	1.15	1.25	0.31	0.68	1.01	1.09	1.43
AGED [68]	0.22	0.36	0.55	0.67	0.78	0.17	**0.28**	0.51	0.64	0.86	0.27	0.43	0.82	0.84	1.06	0.27	0.56	**0.76**	**0.83**	1.25
Imitation [199]	0.21	0.34	0.53	0.59	0.67	0.17	0.30	0.52	0.65	0.79	0.23	0.44	0.86	0.85	0.95	0.27	0.56	0.82	0.91	1.34
ConvSeq2Seq [113]	0.33	0.54	0.68	0.73	-	0.22	0.36	0.58	0.71	-	0.26	0.49	0.96	0.92	-	0.32	0.67	0.94	1.01	-
Trajectory [129]	**0.18**	**0.31**	**0.49**	**0.56**	0.65	**0.16**	0.29	0.50	0.62	0.76	0.22	**0.41**	0.86	0.80	0.87	**0.20**	0.51	0.77	0.85	1.33
Grammar [156]	0.26	0.44	0.67	0.77	0.84	0.20	0.34	0.54	0.68	0.85	0.27	0.50	0.92	0.90	1.00	0.30	0.65	0.92	1.00	1.37
Mix&Match [10]	0.33	0.48	0.56	0.58	**0.64**	0.23	0.34	**0.41**	**0.50**	**0.61**	0.23	0.42	0.79	**0.77**	**0.82**	0.25	0.60	0.83	0.89	**1.12**
DLow [212]*	0.31	0.42	0.53	0.75	0.83	0.24	0.32	0.44	0.55	0.77	**0.21**	0.43	0.80	0.79	0.97	0.31	0.55	0.80	0.88	1.15
Ours	0.23	0.42	0.73	0.83	0.89	0.17	0.33	0.66	0.85	1.02	0.29	0.50	**0.72**	0.78	0.83	0.27	**0.47**	0.82	1.07	1.31

Table 6.6: Mean Angular Error on Human3.6M [85]. *from [9]

	Basketball					Basketball Signal					Directing Traffic					Jumping				
milliseconds	80	160	320	400	1000	80	160	320	400	1000	80	160	320	400	1000	80	160	320	400	1000
Seq2Seq [132]	0.50	0.80	1.27	1.45	1.78	0.41	0.76	1.32	1.54	2.15	0.33	0.59	0.93	1.10	2.05	0.56	0.88	1.77	2.02	2.4
convSeq2Seq [113]	0.37	0.62	1.07	1.18	1.95	0.32	0.59	1.04	1.24	1.96	0.25	0.56	0.89	1.00	2.04	0.39	0.6	1.36	1.56	2.01
Trajectory [129]	**0.33**	**0.52**	**0.89**	**1.06**	**1.71**	**0.11**	**0.20**	**0.41**	**0.53**	**1.00**	**0.15**	**0.32**	**0.52**	**0.60**	2.00	**0.31**	**0.49**	**1.23**	**1.39**	**1.80**
Ours	0.41	0.66	1.15	1.38	2.05	0.30	0.56	0.97	1.12	1.56	0.27	0.48	0.78	0.91	**1.50**	0.84	0.87	1.43	1.65	2.04

	Soccer					Walking					Washwindow					Average				
milliseconds	80	160	320	400	1000	80	160	320	400	1000	80	160	320	400	1000	80	160	320	400	1000
Seq2Seq [132]	0.29	0.51	0.88	0.99	1.72	0.35	0.47	0.60	0.65	0.88	0.30	0.46	0.72	0.91	1.36	0.38	0.62	1.02	1.18	1.67
convSeq2Seq [113]	0.26	0.44	0.75	0.87	1.56	0.35	0.44	**0.45**	**0.50**	0.78	0.30	0.47	0.80	1.01	1.39	0.32	0.52	0.86	0.99	1.55
Trajectory [129]	**0.18**	**0.29**	0.61	**0.71**	1.40	0.33	0.45	0.49	0.53	**0.61**	0.22	**0.33**	**0.57**	0.75	1.20	**0.25**	**0.39**	**0.68**	**0.79**	**1.33**
Ours	0.25	0.42	**0.60**	0.79	**1.06**	**0.26**	**0.41**	0.49	0.53	0.71	**0.21**	**0.33**	0.61	**0.74**	**1.18**	0.35	0.52	0.83	0.96	**1.33**

Table 6.7: Mean Angular Error on CMU Mocap [1].

up of a person standing and gesticulating.

If we remove only one of the loss terms $\mathcal{L}_{\text{sym}}$ (orange), $\mathcal{L}_{\text{sym}}^{\text{adv}}$ (green), or $\mathcal{L}_{\text{adv}}$ (purple), the NDMS score decreases. Without $\mathcal{L}_{\text{adv}}$ (purple), our method furthermore loses the ability to generate multiple samples for a given input sequence. Removing L_{rec} (not plotted) results in unrealistic motion and poses. Without $\mathcal{L}_{\text{sym}}^{\text{adv}}$ we observed that the network sometimes predicts the same intention label for an unrealistic long time.

Impact of $\lambda(\tau)$: For the reconstruction loss L_{rec} (6.8), we use the weighting function $\lambda(\tau)$ which linearly decreases from 1 to 0 until τ_{rec}. In Figure 6.5 (left), we compare three settings: τ_{rec}=15, τ_{rec}=30, and no weighting at all. In the latter case, $\lambda(\tau)$=1. We observe that all settings produce similar early results but that decaying the reconstruction loss to 0 yields the best results over long time horizons. The reason for this is that the adversarial learning scheme has a greater influence with a more aggressive reconstruction decay, which allows more realistic motion over longer time horizons. However, early motion smoothness is not impeded by this. For this reason we set τ_{rec}=15.

Effect of Clustering: In Section 8.2.1.2, we describe our method to obtain frame-wise symbolic labels in an unsupervised way. For clustering, we merge cycles to avoid high frequent changes of labels. In Figure 6.5 (middle), we compare our clustering approach with a naive clustering where we only apply k-means to the training data. We observe that the naive clustering results in more volatile predictions as the human motion generator tries to catch up to the fast-changing symbolic labels. When using our clustering, on the other hand, we observe more stable predictions with higher quality results.

Impact of γ: The parameter γ in (7.10) defines for how many frames ahead the intention is forecast. If $\gamma = 1$, the decoder d_p does not look ahead and takes only the estimated intention labels until the current frame into account. In Figure 6.5 (right), we evaluate four different values for γ: no look-ahead, 5 frames look-ahead, 10 frames look-ahead, and 20 frames look-ahead. We observe that small

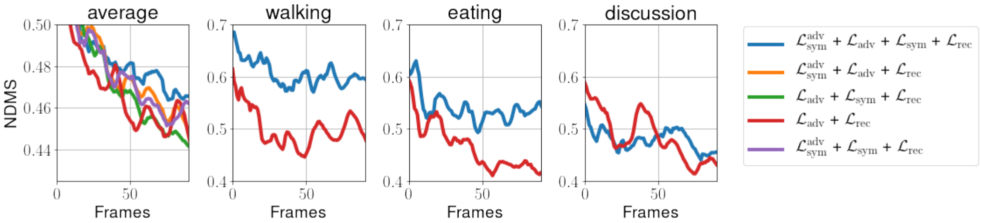


Figure 6.4: Impact of loss functions. The NDMS score is averaged over 50 samples per input sequence. *average*: NDMS scores averaged over all actions of Human3.6M [85]. *walking, eating, discussion*: Comparison of motion forecasting without (red) and with (blue) intention forecasting for the corresponding action.

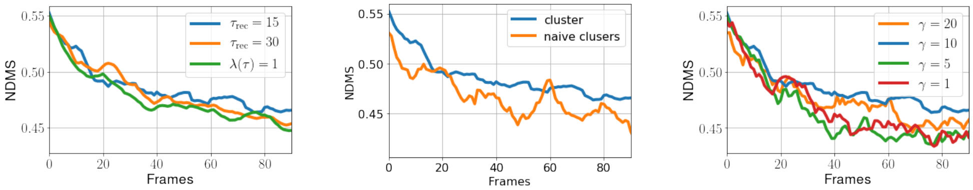


Figure 6.5: NDMS scores averaged over all actions of Human3.6M [85] using 50 samples per input sequence. Left: Comparing various values of τ_{rec} for the reconstruction loss. In case of $\lambda(\tau)$=1, we use always 1 as weight. Middle: Comparing naive clustering with more elaborate clustering that merges cyclic patterns into cohesive clusters. Right: Comparing various γ look-ahead values for human motion anticipation.

values of γ substantially decrease the NDMS score. This shows that it is very important to forecast the intention ahead of time. However, when γ is too large, it also reduces the quality since only the next upcoming action is relevant for a smooth motion transition and longer look-ahead times distract the pose decoder. As a default parameter, we thus set $\gamma = 10$.

6.4.3 Very long motion forecasting

For state-of-the-art method evaluation we obtained sequences of up to 4 seconds. Our method, however, is capable of generating much longer sequences, even for non-cyclic motion. Figure 6.6 shows that our method produces consistent results over very long time horizons of 30 seconds. This is consistent with our observation that the motion, even for non-cyclic motion such as Eating, remains realistic for very long time horizons.

6.4.4 Evaluation Score

6.4.4.1 NDMS vs. Euclidean Distance/Velocity

For our evaluation score, we propose NDMS since other measures like L2 distance or L2 velocity distance are insufficient. To show this, we take two sequences, one containing real ground-truth

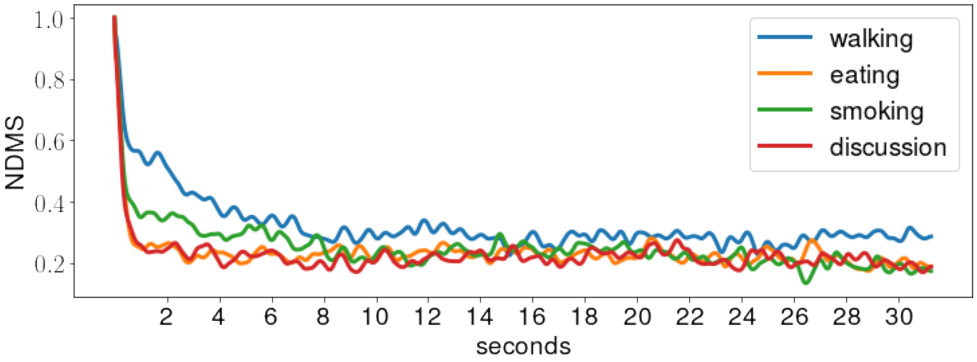


Figure 6.6: Average NDMS score for very long forecasting of 30 seconds on Human3.6M [85].

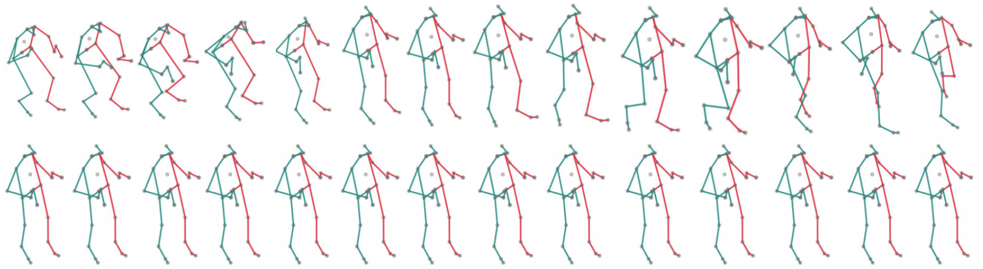


Figure 6.7: Real motion (top row) vs. zero velocity (bottom row) of around two seconds. Zero velocity is very unrealistic as it always produces the same pose as output [132].

motion from the training set and one containing a static pose (zero velocity) (see Figure 6.7). We use the scores to measure the plausibility of a walking motion using the walking sequences of the test data as reference. While the real motion should have a high score or low distance, the zero velocity sequence should perform poorly since it is not a walking motion.

Our results are summarized in Figure 6.8. We observe that NDMS (a) scores the real motion high and the zero velocity very low, as it should be the case. Note that for the distances in (b) and (c) lower values are better, while for (a) higher values are better. For the Euclidean distance (b) and the mean squared error over velocity (c), the zero velocity performs better than the real motion. This shows that neither the L2 distance nor the mean squared error over velocity are useful metrics to measure the plausibility of a sequence.

6.4.4.2 NDMS vs. NPSS

The Pearson correlation coefficient with the user study and NDMS is 0.901. This shows that the proposed measure highly correlates with human perception. The correlation coefficient for NPSS [10] is -0.238. The negative correlation is due to the competitive NPSS of Grammar [156] although the

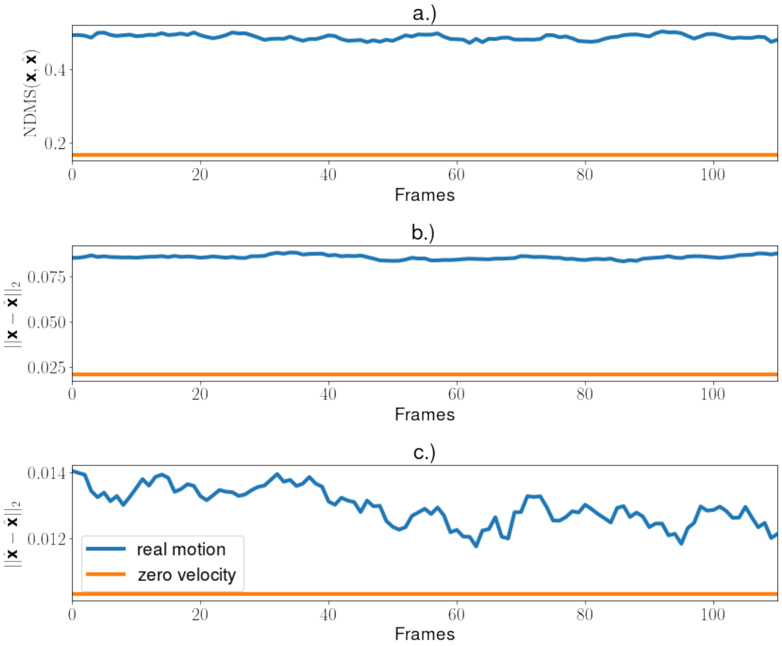


Figure 6.8: Comparing baseline distances with NDMS: a.), b.) and c.) show real motion (blue) and zero velocity prediction (orange) for NDMS (higher is better), L2 distance (lower is better) and L2 velocity distance (lower is better), respectively. While NDMS scores the real motion higher, the L2 and L2 velocity distance would rate the static pose as more plausible than the real motion.

generated motions are perceived as unrealistic by humans.

6.4.4.3 NDMS vs. ADE, FDE, MMADE and MMFDE

DLow [212] proposes ADE, FDE, MMADE and MMFDE to evaluate the quality of multi-modal human motion prediction. In contrast to other state-of-the-art methods they utilize 3D skeletons rather than an angular representation. We train and evaluate our method using the data and evaluation code from [212]. The results in Table 6.9 show that DLow has a higher diversity while our approach has a lower ADE. For the other metrics FDE, MMADE, and MMFDE, DLow performs better. This contradicts the results from the user study where in particular walking sequences generated from DLow are considered as unrealistic. This is due to the motion discontinuity between the observed poses and the forecast poses, but also due to the very high diversity of the generated sequences. The forecast sequences quickly generate motions that are very unlikely to occur after a walking motion. While these issues are not measured by ADE, FDE, MMADE and MMFDE, NDMS penalizes motion discontinuities. In Figure 6.9, we plot NDMS over time. As can be seen, NDMS drops for DLow very quickly and increases after 8 frames. This is due to the discontinuity between observed frames and forecast frames.

Method	APD ↑	ADE ↓	FDE ↓	. MMADE ↓	MMFDE ↓	NDMS ↑
[197]	6.723	0.461	0.560	0.522	0.569	-
[206]	0.403	0.457	0.595	0.716	0.883	-
[16]	7.214	0.858	0.867	0.847	0.858	-
[24]	6.265	0.448	0.533	0.514	0.544	-
[48]	6.769	0.461	0.555	0.524	0.566	-
[74]	6.509	0.483	0.534	0.520	0.545	-
[211]	9.330	0.493	0.592	0.550	0.599	0.294
[212]	**11.741**	0.425	**0.518**	**0.495**	**0.531**	0.311
Ours	3.477	**0.413**	0.631	0.662	0.770	**0.366**
Baseline	*16.418*	*0.429*	*0.451*	*0.520*	*0.478*	0.166

Table 6.8: Evaluation over 2 seconds for multi-modal human motion anticipation on Human3.6M [85] as defined in DLow [212]. The motion is represented as 3D skeletons centered at the origin but with global rotation.

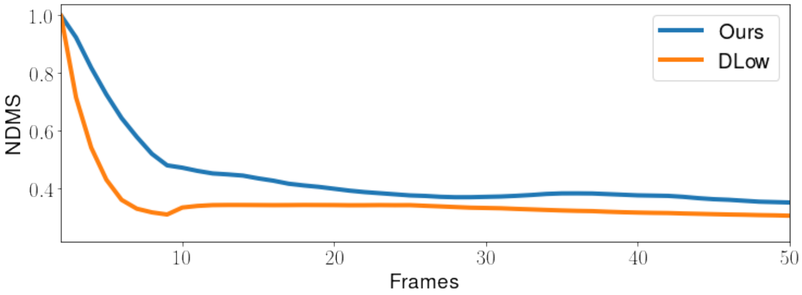


Figure 6.9: NDMS score for 50 samples per input sequence on Human3.6M [85] averaged over 15 actions. The first few frames exhibit very high scores as the motion words contain mostly poses from the observed sequence at the beginning, as described in Section 6.4.4. We observe that DLow has a much sharper decline and a small dent due to the motion discontinuity between observed and forecast frames.

In order to show that the measures ADE, FDE, MMADE and MMFDE can be easily fooled, we construct a very unrealistic multi-modal baseline. In order to generate 50 samples for a single observation, we generate 50 sequences with static poses (zero velocity) as shown in Figure 6.7. To this end, we cluster the poses of the training data to obtain 48 clusters. For each cluster, we take the mean pose as static pose. Note that these 48 sequences are independent of the observation, but they generate a very high diversity (APD). For the remaining two sequences, we take the last pose of the observation and the mean pose of the observed sequence, respectively, as static pose. As shown in Table 6.9, this baseline performs very well for APD, ADE, FDE, MMADE and MMFDE although none of the 50 sequences contains any motion and all of them are highly unrealistic. In contrast, the low NDMS score reliably indicates that the baseline does not generate plausible sequences.

Method	APD ↑	ADE ↓	FDE ↓	. MMADE ↓	MMFDE ↓
[197]	6.723	0.461	0.560	0.522	0.569
[206]	0.403	0.457	0.595	0.716	0.883
[16]	7.214	0.858	0.867	0.847	0.858
[24]	6.265	0.448	0.533	0.514	0.544
[48]	6.769	0.461	0.555	0.524	0.566
[74]	6.509	0.483	0.534	0.520	0.545
[211]	9.330	0.493	0.592	0.550	0.599
[212]	**11.741**	0.425	**0.518**	**0.495**	**0.531**
Ours	3.477	**4.125**	0.631	0.662	0.770

Table 6.9: Evaluation for multi-modal human motion anticipation on Human3.6M [85] as defined in DLow [212]. The motion is represented as 3D skeletons centered at the origin but with global rotation.

Method	IS
Seq2Seq [132]	7.5 ± 0
Trajectory [129]	9.2 ± 0
Grammar [156]	**10.3** ± 0
Yan et al. [206] $\star$	1.9 ± 0.4
Walker et al. [197] $\star$	1.8 ± 0.6
Barsoum et al. [16] $\star$	2.1 ± 1.3
Mix&Match [10] $\star$	7.3 ± 1.4
Mix&Match [10]	7.5 ± 1.1
Ours	9.7 ± 0.6

Table 6.10: Inception score as described in [10] for Human3.6M. $\star$ denotes results reported in [10].

6.4.5 NDMS vs. Inception Score

For evaluating long-term human motion forecasting we report the inception score (IS) as described in [10]. Since the original scoring model is not available, we followed the description [10] and retrained a skeleton-based action classifier [112]. We pass then sequences with 16 observation frames and 60 prediction frames to the scoring model. For methods that forecast multiple sequences for one observation sequence, we generate 50 samples for each single input sequence and calculate the mean inception score as well as the standard deviation, following [10]. For the other methods, we compute the inception score of a single forecast sequence.

The results can be seen in Table 6.10. First, we validate that our newly trained *inception* network works properly by comparing our results of [10] with the results reported in their work. The reproduced result is even slightly better. While our approach outperforms all methods that generate multiple future sequences [206, 197, 16, 10], we observe that Grammar [156] achieves a higher inception score. However, our user study shows that poses generated by Grammar are less realistic than the sequences that are generated by Mix&Match [10] or our approach. This indicates that the inception score is not a very reliable measure for the plausibility of the forecast human motion. The generative grammars [156] achieve a very high inception score, as can be seen in Table 6.10. The high score, however, is not supported by our user study and the qualitative results. This shows the weakness of the inception score, which does not occur for the proposed NDMS score.

6.5 Conclusion

In this work, we presented an approach that forecasts multiple plausible sequences of human motion for a single observation[1]. In this way, the model can deal with the uncertainty of the future. In order to ensure that the forecast sequences remain plausible even for longer time horizons, we proposed a novel network that not only forecasts the human motion but also the intention. By forecasting the intention ahead of time, the network generates plausible transitions between actions. Furthermore, we presented a new quality score that allows to compare methods that generate multiple sequences even for long time horizons. We demonstrated that the new similarity score correlates better with human judgement than NPSS and that the method produces superior results for long-term human motion anticipation.

[1] Source code is available at `https://github.com/jutanke/human_motion_ndms`

CHAPTER 7

Social Diffusion: Long-term Multiple Human Motion Anticipation

The method introduced in the previous chapter utilizes the mode of intention to anticipate much longer motion than previously possible. In this work we utilize the mode of social interactions to achieve long-term social motion anticipation. Social motion anticipation differs from human motion anticipation in that it not only concerns the quality of individual forecast motion but also must ensure that the joint forecast motion remains plausible. Similar to the previous Chapter 6 we utilize a generative model to achieve this challenging task. However, instead of a generative adversarial network we use a Diffusion model, which has a more stable training, due to it being formulated as a maximum likelihood estimation process instead of an adversarial learning scheme. As social motion forecasting is a newly proposed problem we further introduce a novel evaluation scheme and a new evaluation method, *Symbolic Social Cues Protocol*.

Individual Contribution: The following chapter is based on the publication [186]:

Social Diffusion: Long-term Multiple Human Motion Anticipation
Julian Tanke, Linguang Zhang, Amy Zhao, Chengcheng Tang, Yujun Cai, Lezi Wang, Po-Chen Wu, Juergen Gall and Cem Keskin
IEEE International Conference on Computer Vision (ICCV), 2023.

This publication was done in very close collaboration between Linguang Zhang and Julian Tanke. Amy Zhao, Chengcheng Tang, Yujun Cai, Lezi Wang, Po-Chen Wu, Juergen Gall and Cem Keskin provided scientific guidance and supported this work with very valuable feedback and suggestions. The idea of utilizing a Diffusion model for social motion anticipation was proposed by Julian Tanke. The network design, experiments and novel evaluation score was proposed by Julian Tanke.

Contents

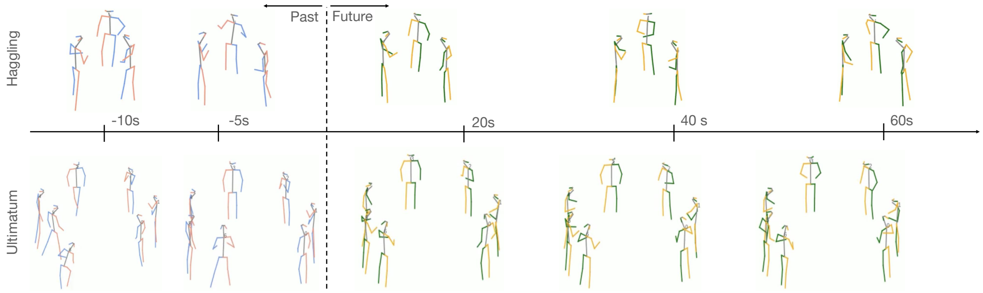


Figure 7.1: Long-term multi-person motion anticipation: given a sequence of human social interactions (blue-red skeletons), the proposed model can forecast long-term multi-person motions (more than one minute). The predicted motions demonstrate diverse poses and plausible social interactions.

7.1 Introduction

Understanding and anticipating social interactions in groups of people is a challenging and highly relevant topic [106, 14, 100, 140, 145, 8, 104]. For instance, it is essential for socially-compliant robots [2], but it is also relevant for neuroscience and social sciences since it allows to develop computational models on how the behavior of other persons is perceived and how it changes the own behavior.

Forecasting realistic social interactions, however, is very challenging for two reasons. First, social interactions tend to last for tens of seconds [102] or even minutes - much longer than the prediction from most of the existing human motion anticipation models [72, 200, 5, 68, 93, 132, 152, 28, 79, 113, 7, 38, 129, 130]. Second, social interactions consist of interdependent motions [168, 111], which requires modeling the relationships among all individuals. For example, in conversational turn-taking, a person's turn to talk highly depends on the start/end of the others' speaking. While multi-person motion anticipation has emerged as a new topic, current approaches [72, 200, 5] do not pay much attention on complex social interactions. For instance, they do not preserve the social role of individuals in a group such that the interactions become socially implausible over time.

To address the limitations of existing models, we propose Social Diffusion to predict motions of multiple people and ensure contextually plausible interactions, as shown in Fig. 7.1. To this end, we learn the distribution of human motion by leveraging a diffusion model [125, 94, 166, 177, 78, 178, 189, 214]. To enforce information exchange among people, which is critical to predicting contextually plausible interactions, we introduce an order-invariant aggregation function to aggregate motion features from all people. For inference, we feed back the input sequence to the signal during the reverse-diffusion steps to condition the motion generation on the past motion. Our method is fully convolutional which allows us to generate sequences of arbitrary size. This allows us to not just forecast the next few seconds of an input motion but also to forecast social interactions that last longer. Furthermore, our approach is very flexible in the sense that the number of persons during training and inference can differ. To the best of our knowledge, our approach represents the first diffusion model that produces multi-person motions at the same time.

As a second contribution, we propose a new evaluation protocol for social interactions based on Symbolic Social Cues, which measures whether the forecast motion is socially plausible. Our key observation is that the probabilities of transitions between social interaction states are highly correlated with the plausibility of predicted social interactions. In a conversation, for example, a person usually starts talking only when a peer stops talking. To evaluate predicted motions, we first build the state transition graph by extracting states from the motions. We then treat the state transition graph as a probability distribution and compare it to the real data distribution.

For evaluation, we use the Haggling dataset [99] which comprises 175 videos of well-defined triadic social interactions. In contrast to other existing multi-person human motion datasets [136, 195, 200], the persons have different social roles that impact their behavior. We furthermore evaluate our approach on the MuPoTS-3D [136], 3DPW [196], and CMU-Mocap [1, 200] dataset. On all four datasets, our approach outperforms the state of the art for multi-person human motion forecasting.

In summary, our contribution is two-fold:

1. We propose Social Diffusion, the first stochastic multi-person motion anticipation model that outperforms the state of the art on common multi-person motion anticipation datasets.

2. We propose a novel social interaction evaluation protocol that considers not only the validity of poses but also the plausibility of social interactions.

7.2 Social Diffusion Model

As illustrated in Figure 7.1, we aim to forecast the motion of multiple persons that interact with each other. The forecast motion should be realistic and socially plausible. For instance, not all persons should talk at the same time. Formally, we represent a human motion sequence with p people of length N as $\mathbf{X}^{1:N} \in \mathbb{R}^{N \times p \times \delta}$ where δ represents the dimension of the individual pose vector at a given frame. Our goal is then to predict the future motion $\hat{\mathbf{X}}^{n+1:N}$ for all people, given their past motions $\mathbf{X}^{1:n}$:

$$\hat{\mathbf{X}}^{n+1:N} = \mathrm{SDM}(\mathbf{X}^{1:n}). \tag{7.1}$$

Before we describe the proposed Social Diffusion Model (SDM) in Section 7.2.2, we will briefly describe a generic diffusion model [78] in Section 7.2.1. In Section 7.3, we will then introduce the social interaction evaluation protocol.

7.2.1 Diffusion Model

A latent representation $\mathbf{X}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{1})$ is obtained via a T step Markov Gaussian noising process $q(\mathbf{X}_T|\mathbf{X}_0)$ where $\mathbf{X}_0 \equiv \mathbf{X}^{1:N}$ is a real motion sequence from the training set. The Markov Gaussian

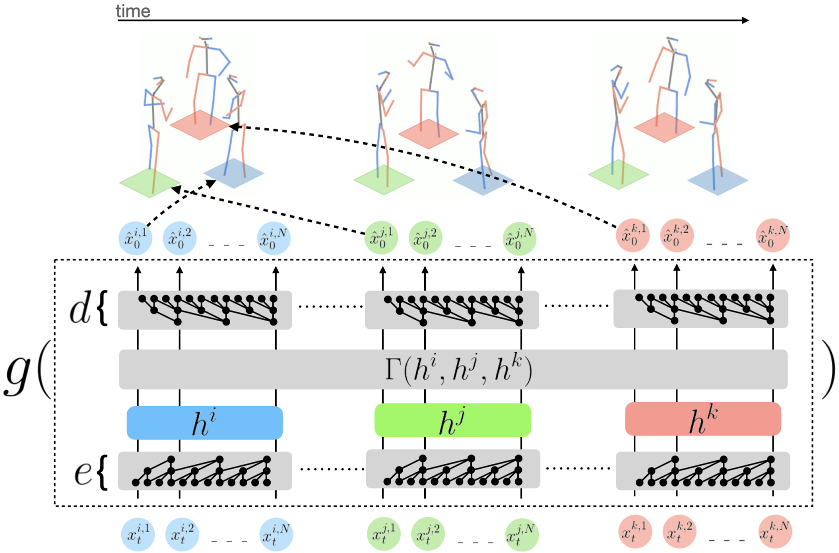

time
$\hat{x}_0^{i,1}$ $\hat{x}_0^{i,2}$ - - - $\hat{x}_0^{i,N}$
$\hat{x}_0^{j,1}$ $\hat{x}_0^{j,2}$ - - - $\hat{x}_0^{j,N}$
$\hat{x}_0^{k,1}$ $\hat{x}_0^{k,2}$ - - - $\hat{x}_0^{k,N}$
$d\{$
$g($
$\Gamma(h^i, h^j, h^k)$
$)$
h^i
h^j
h^k
$e\{$
$x_t^{i,1}$ $x_t^{i,2}$ - - - $x_t^{i,N}$
$x_t^{j,1}$ $x_t^{j,2}$ - - - $x_t^{j,N}$
$x_t^{k,1}$ $x_t^{k,2}$ - - - $x_t^{k,N}$

Figure 7.2: Model overview: the reverse diffusion process g consists of a fully convolutional causal encoder e and a fully convolutional causal decoder d that produces a denoised motion sequence $\hat{\mathbf{x}}_0^i \equiv \hat{\mathbf{x}}_0^{i,1:N}$ for person i, given the bottleneck state h^i and the aggregation function $\Gamma(\{\mathbf{h}_j \mid j \in 1, ..., p\})$ over all people in the scene.

noising process can be written in closed form:

$$q(\mathbf{X}_t|\mathbf{X}_0) = \mathcal{N}(\mathbf{X}_t; \sqrt{\alpha_t}\mathbf{X}_0, (1-\alpha_t)\mathbf{I}) \tag{7.2}$$

where $\alpha_t \in (0,1)$ is a step-dependent fixed hyper-parameter. To sample from the generative model, we learn to invert the noising step using the generator function g:

$$\hat{\mathbf{X}}_{0,t} = g(\mathbf{X}_t, t) \tag{7.3}$$

The key contribution of our model is the novel generator function g, which models social interactions over time and will be described in Section 7.2.2. Following [78, 189, 161], the loss during training is defined by:

$$\mathcal{L} = \mathbb{E}_{\mathbf{X}_0 \sim p(\mathbf{X}), t \sim [1,T]} \Big[\big|\big|\mathbf{X}_0 - g(\mathbf{X}_t, t)\big|\big| \Big] \tag{7.4}$$

For inference, we reverse-iterate over Equation (7.3), starting at sampling step T and latent representation $\mathbf{X}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{1})$. At each iteration t, we slowly denoise the motion sequence using (7.2) and (7.3):

$$\hat{\mathbf{X}}_{0,t-1} = g(q(\hat{\mathbf{X}}_{t-1}|\hat{\mathbf{X}}_{0,t}), t-1) \tag{7.5}$$

The final denoised motion is obtained when $t = 1$.

7.2.2 Multi-person Motion Generator

The diffusion model described in Section 7.2.1 produces unconditioned motion. In order to use the model for motion forecasting, we need to condition the model on the observed motion sequence of all persons $\mathbf{X}_0^{1:n} = \mathbf{X}^{1:n}$. To this end, we modify the inference sampling (7.5) to also include past motion as follows:

$$\hat{\mathbf{X}}_{0,t-1} = g(q(\hat{\mathbf{X}}_{t-1}|\mathbf{X}^{1:n} \cup \hat{\mathbf{X}}_{0,t}^{n+1:N}), t-1) \tag{7.6}$$

The reverse diffusion process $g(\mathbf{X}_t, t)$ consists of three components, a causal temporal convolutional encoder e, a causal temporal convolutional decoder d, and an order-invariant function Γ that aggregates the interaction of the different persons, see Figure 7.2.

The encoder e and the decoder d process each individual sequence independently while Γ ensures that information flows between all persons in the scene. Formally, given $\mathbf{x}_t^i = \hat{\mathbf{X}}_t^{i,1:N} \in \mathbb{R}^{N\times\delta}$, which is the motion sequence for a single person i at scheduled noising step t, we obtain for each person i in the scene a bottleneck encoding $\mathbf{h}^i$:

$$\mathbf{h}^i = e(\mathbf{x}_t^i, t) \;\; \forall i \in 1, ..., p. \tag{7.7}$$

The encoder e consists of a 4-layer temporal convolutional network where each layer progressively reduces the temporal resolution by half via striding. In each layer, the noising step t is fed via sinusoidal positional encoding [194]. To produce the denoised motion $\hat{\mathbf{x}}_0^i$, the decoder d can be utilized as follows:

$$\hat{\mathbf{x}}_0^i = d\big(\mathbf{h}^i, t, \Gamma(\mathbf{h}^j \; \forall j \in 1, ..., p)\big) \tag{7.8}$$

where d is a 4-layer temporal convolutional network and each layer progressively doubles the temporal resolution via linear upsampling. As for the encoder, the noising step t is fed via sinusoidal positional encoding to each layer. In addition, the output of the order-invariant aggregation function Γ is concatenated to $\mathbf{h^i}$ before passing it to the first convolutional layer. The estimated motion sequences $\hat{\mathbf{x}}_0^i$ of each person i at noising step t are then concatenated to obtain $\hat{\mathbf{X}}_{0,t}$ and the approach proceeds to the next step $t-1$.

The order-invariant aggregation function Γ passes information from other people in the scene. In our experiments, we evaluate two aggregation functions, averaging ($\Gamma_{\mathbb{E}}$) over all people and multi-headed attention [194] (Γ_{attn}):

$$\Gamma_{\text{attn}}(\mathbf{H}) = \text{MultiHead}(\mathbf{H}) \tag{7.9}$$

$$\Gamma_{\mathbb{E}}(\mathbf{H}) = \frac{1}{p}\sum_{i=1}^{p} \mathbf{h}^i \tag{7.10}$$

where $\mathbf{H} = \{h^i\}_{i=1}^p$ are the embeddings of all layers of the encoder and for all people in the scene. $\text{MultiHead}(\mathbf{H})$ calculates the self-attention across all people for a given frame.

7.2.3 Implementation Details

We follow state-of-the-art diffusion models [78, 125, 189] and use the cosine variance schedule. We set the number of diffusion steps to $T = 1000$. The encoder e consists of four layers of convolutional blocks with kernel size 3 and stride 2. The decoder d consists of four layers of convolutional blocks with additional upsampling layers that upsample the input sequence by factor two using linear interpolation. We standardize the training data to have zero mean and standard deviation one. We normalize all poses by splitting pose and global translation: each pose is transformed into a hip-centric coordinate frame and the pose is concatenated with the global rotation and translation to form a δ dimensional pose vector.

7.3 Symbolic Social Cues Protocol

We are interested in anticipating the social interactions among multiple people. Multi-person social interactions consist of several intricate and complex behaviours such as paying attention to a specific person [173] and turn-taking [168, 111], which usually take tens of seconds or even minutes. Current state-of-the-art multi-person motion anticipation methods [5, 200] calculate the Mean Per Joint Positional Error (MPJPE) using the ground-truth sequence, which is only meaningful for short time horizons of around one second [10, 65, 167, 185, 212]. More important, however, is that it does not measure the realism of social interaction.

We thus propose the Symbolic Social Cues Protocol (SSCP), which divides the social interactions into a set of discrete interaction classes. In SSCP, we define a social signal function

$$C^{1:N} = s(\mathbf{X}^{1:N}) \tag{7.11}$$

which takes as input a multi-person motion sequence $\mathbf{X}^{1:N} \in \mathbb{R}^{N \times p \times \delta}$ and produces a discrete symbolic representation $C^{1:N} = \{c_n\}_{n=1}^{N}$, where $c_n \in \{1, ..., m\}$ and m represents the total number of symbolic states. A symbolic state is a unique summary of the current state of interaction, e.g., a person is talking and another person is listening. Given a test set $\mathcal{X} = \{\mathbf{X}_i^{1:N}\}_{i=1}^{K}$ with K sequences, we can now calculate the probability distribution p_{sscp} over the social state transitions.

$$p_{\text{sscp}}(\mathcal{X}) = \frac{1}{\zeta} \sum_{i=1}^{K} \text{stm}(s(\mathbf{X}_i^{1:N})) \tag{7.12}$$

where $\text{stm}(C_i^{1:N})$ produces the $m \times m$ state transition matrix for the discrete symbolic sequence $C^{1:N}$ and $\zeta = \sum_{i=1}^{K} \sum_{m',m''} \text{stm}(C^{1:N})_{m',m''}$ is a normalization constant to ensure that p_{sscp} is a valid probability distribution.

To evaluate a motion anticipation model f, we predict the future motion for all K test sequences from a fixed start frame n until the end of the sequence N:

$$\hat{\mathcal{X}}^{n+1:N} = \{f(\mathbf{X}_i^{1:n})\}_{i=1}^{K} \tag{7.13}$$

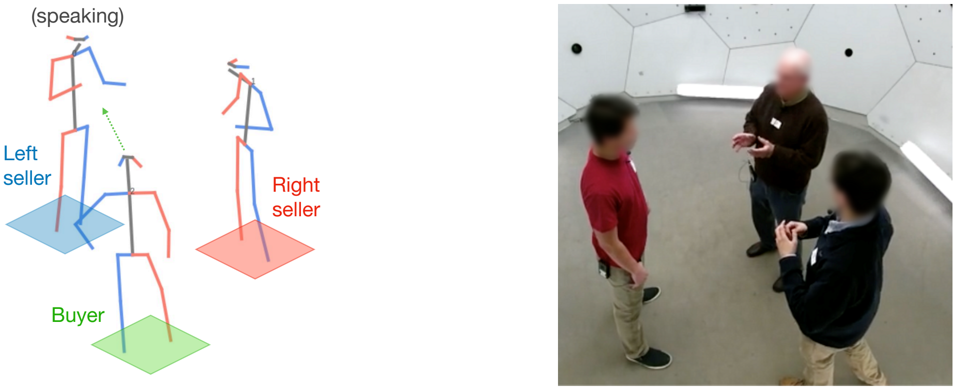
We can now calculate the distance between the generated and ground-truth social motion distribution:

$$D_{\text{JSD}}\big(p_{\text{sscp}}(\mathcal{X}^{n:N}), p_{\text{sscp}}(\hat{\mathcal{X}}^{n:N})\big). \tag{7.14}$$

D_{JSD} is the squared Jensen-Shannon distance [53, 62]:

$$D_{\text{JSD}}(p||q) = \sqrt{\frac{\big(D_{\text{KL}}(p||\frac{p+q}{2}) + D_{\text{KL}}(q||\frac{p+q}{2})\big)}{2}} \tag{7.15}$$

where p and q are probability distributions and D_{KL} is the Kullback-Leibler divergence. Note that we compare the generated motion $\hat{\mathcal{X}}^{n+1:N}$ to the test set $\mathcal{X}^{n+1:N}$ with the same start frame as the data distribution might shift across time.



(a)

(b)

Figure 7.3: A sample frame from the Haggling dataset [99] for evaluating social interactions. (a): 3D poses in a haggling sequence. Blue limbs represent the left body side while red limbs represent the right body side. The buyer's attention is indicated as green arrow. (b): a sample video frame from a haggling sequence.

Method	CMU-Mocap			MuPoTS-3D			3DPW		
	1s	2s	3s	1s	2s	3s	1s	2s	3s
LTD [129]	1.37	2.19	3.26	1.19	1.81	2.34	4.67	7.10	8.71
HRI [130]	1.49	2.60	3.07	0.94	1.68	2.29	4.07	6.32	8.01
SP [4]	1.15	2.71	3.90	0.92	1.67	2.51	4.17	7.17	9.27
MRT [200]	0.96	1.57	2.18	**0.89**	1.59	2.22	3.87	6.12	7.83
Ours	**0.74**	**1.06**	**1.34**	1.15	**1.29**	**1.44**	**1.64**	**2.72**	**3.55**

Table 7.1: MPJPE ↓ in dm on different datasets.

7.4 Experiments

We evaluate our approach on four datasets. Following [200], we report the Mean Per Joint Positional Error (MPJPE) in global and local aligned coordinates for the multi-person human motion datasets MuPoTS-3D [136], 3DPW [196] and CMU-Mocap [1, 200]. MuPoTS-3D [136] contains recordings of 2 to 3 persons in workout settings. Interactions between the persons are rare. 3DPW [196] contains recordings of 1 to 2 persons and the sequences cover a wide range of different activities. The level of interactions range from no interaction and little interaction, like two persons walking, to close interactions such as dancing. CMU [200] combines the motion of different sequences from the CMU-Mocap dataset [1]. The composition of motion sequences, however, does not reflect realistic interactions. For these datasets, 1 second of human motion is observed and 1-3 seconds need to be forecast. As our model is generative, we sample 8 samples for each test sequence and report the average over all 8 samples.

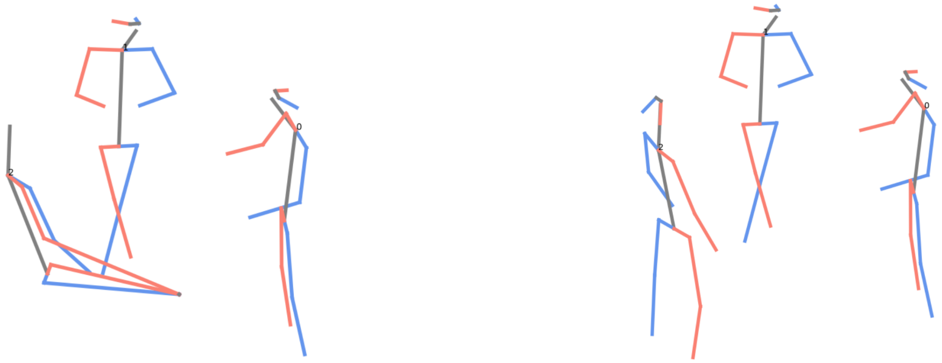


Figure 7.4: Example frame from sequence **170224_haggling_a2** of the Haggling dataset [99]. Left: Original poses from [99]. Right: Poses after our cleaning step.

Method	CMU-Mocap						MuPoTS-3D						3DPW					
	Root			Pose			Root			Pose			Root			Pose		
	1s	2s	3s	1s	2s	3s	1s	2s	3s	1s	2s	3s	1s	2s	3s	1s	2s	3s
LTD [129]	0.97	1.73	2.62	0.98	1.21	1.37	0.89	1.39	1.91	0.88	1.14	1.31	4.28	6.79	8.41	1.54	1.76	1.98
HRI [130]	0.96	2.06	3.11	1.05	1.37	1.58	**0.66**	1.30	2.16	0.73	1.07	1.30	3.67	6.42	8.64	1.43	1.75	1.94
SP [4]	0.96	2.01	2.96	1.03	1.41	1.71	0.96	1.38	2.21	0.72	1.08	1.30	3.76	6.86	9.07	1.60	1.95	2.15
MRT [200]	**0.60**	1.12	1.71	0.79	1.05	1.22	0.67	**1.25**	1.86	0.69	0.99	1.19	3.42	5.69	7.30	1.52	1.75	1.93
Ours	0.72	**1.10**	**1.44**	**0.38**	**0.46**	**0.49**	1.14	1.28	**1.42**	**0.59**	**0.64**	**0.67**	**1.66**	**2.76**	**3.59**	**0.94**	**1.03**	**1.06**

Table 7.2: MPJPE ↓ in dm for root joint and pose. The lowest error is in bold and the second lowest is underscored.

7.4.1 Haggling Forecasting Dataset

Since the three datasets contain multiple persons, but very few social interactions, we prepared a new dataset for multi-person forecasting in the context of social interactions. For this, we utilize the Haggling dataset [99] where 122 participants play a social game with two sellers trying to sell their products to a buyer. Each game lasts one minute and contains interesting triadic interactions such as turn-taking and attention changes. A sample scene is shown in Figure 7.3. The dataset consists of 135 training sequences and 40 test sequences, sampled at 30Hz. Since the 3D human poses in the Haggling dataset [99] have been estimated [96, 98], the original dataset contains many artifacts. We thus cleaned the 3D human poses by marking errors and interpolation. We removed a few sequences which we were not able to correct. An example of our manual pose correction is shown in Figure 7.4. For a video, we refer to `https://github.com/jutanke/social_diffusion`. In total, the dataset consists of $234,907$ training and $69,951$ test frames, each with three people.

For evaluation, we take the first 10% of a sequence as observation and forecast the remaining 90% of the sequence, but we also evaluate the motion at intermediate frames ranging from frame 1 to frame 1300. It needs to be noted that long-term forecasting is highly relevant for neuroscience and social sciences since it allows to develop computational models on how the behavior of other persons is perceived and how it changes the own behavior. For measuring the plausibility of the forecast motion of each individual, we use the Normalized Directional Motion Similarity (NDMS) (Chapter 6) since the measure not only considers static poses but also the motion of the forecast sequence.

Frame	1	5	10	15	20	25	30	60	120	250	500	750	1000	1300
MRT [200]	0.624	0.278	0.194	0.212	0.224	0.215	0.215	0.218	0.205	0.180	0.129	0.079	0.062	0.047
Ours ($\Gamma_{\varnothing}$)	**0.644**	**0.280**	**0.206**	0.215	0.225	0.226	0.229	**0.233**	**0.229**	**0.226**	0.229	0.227	0.225	**0.226**
Ours (Γ_{attn})	0.639	**0.280**	0.199	0.213	0.224	**0.229**	**0.232**	0.227	**0.229**	0.223	**0.233**	0.228	**0.233**	0.223
Ours* ($\Gamma_{\mathbb{E}}$)	0.640	0.279	0.204	**0.216**	**0.227**	0.228	0.230	**0.233**	0.227	0.222	0.230	**0.229**	0.230	**0.226**

Table 7.3: Per-frame average NDMS ↑ score on the Haggling dataset. The highest score is in bold and the second highest is underscored.

Furthermore, it can be applied to sequences of any length. NDMS, however, does not measure if the social interactions are plausible.

For evaluating the social motion quality, we utilize the proposed Symbolic Social Cues Protocol described in Section 7.3. To this end, we need to specify the classes of social interactions. A haggling activity is composed of the sellers trying to convince the buyer to purchase their products and the buyer switching attention between the sellers. Throughout the game, certain social patterns emerged [173, 111]:

1. Most of the time, only a single person speaks
2. For almost all frames, at least one person is talking
3. The sellers speak roughly the same amount of time while the buyer seldom talks
4. The buyer pays attention (looks at) to whoever talks
5. The sellers take turns to speak but sometimes interrupt each other

Given the well-defined structure of the task and the emerging social behaviors, we reduce the haggling game to two key signals:

- **talking**: defines who is talking
- **attention**: defines who of the two sellers has the buyer's attention

Given all possible combinations (e.g., both sellers can talk at the same time, or nobody talks), we end up with 16 possible states for each frame and formulate the social interactions as a symbolic representation over time. Note that we have to distinguish between left/right seller to catch events such as attention switching. Please see the supplementary material for more details.

We define the social signal function $s(\mathbf{X}^{1:N})$ as Equation (7.11), which takes a sequence of multi-person motion $\mathbf{X}^{1:N}$ as input and generates one of the 16 distinct states per frame. We define the following states:

- : nobody talks
- $\mathbf{L}$: the left seller is talking
- $\mathbf{R}$: the right seller is talking
- $\mathbf{B}$: the buyer is talking

as well as the following attention states:

- $\hat{\mathrm{L}}$: the left seller has the buyers attention
- $\hat{\mathrm{R}}$: the right seller has the buyers attention

This way we get 16 interaction states:

1. $\hat{\mathrm{L}}$
2. , $\hat{\mathrm{R}}$
3. $\mathbf{L}$, $\hat{\mathrm{L}}$
4. $\mathbf{L}$, $\hat{\mathrm{R}}$
5. $\mathbf{R}$, $\hat{\mathrm{L}}$
6. $\mathbf{R}$, $\hat{\mathrm{R}}$
7. $\mathbf{LR}$, $\hat{\mathrm{L}}$
8. $\mathbf{LR}$, $\hat{\mathrm{R}}$
9. $\mathbf{B}$, $\hat{\mathrm{L}}$
10. $\mathbf{B}$, $\hat{\mathrm{R}}$
11. $\mathbf{BL}$, $\hat{\mathrm{L}}$
12. $\mathbf{BL}$, $\hat{\mathrm{R}}$
13. $\mathbf{BR}$, $\hat{\mathrm{L}}$
14. $\mathbf{BR}$, $\hat{\mathrm{R}}$
15. $\mathbf{BLR}$, $\hat{\mathrm{L}}$
16. $\mathbf{BLR}$, $\hat{\mathrm{R}}$

For the social signal function s to work on any haggling motion sequence, we need to determine three pieces of information:

1. who the buyer is,
2. whom the buyer is paying attention to,
3. whether someone is speaking.

To solve (1) we train a simple buyer detection network, consisting of three layers of bi-directional Gated Recurrent Units, which gets as input a haggling motion and outputs the likelihood of each participant being the buyer. In Table 7.5, we report our accuracy of this approach. We see that the buyer detector correctly identifies the buyer all the time on the test set.

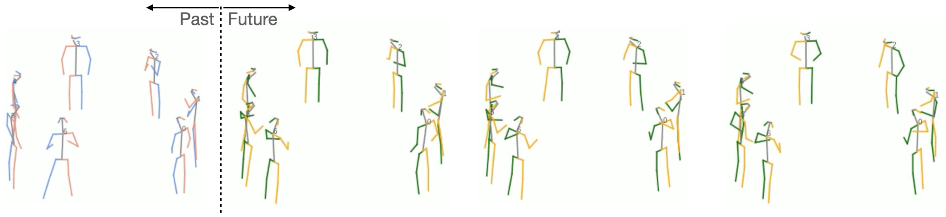


Figure 7.5: Our model is capable of generating realistic motion for 7 people from the *Ultimatum* sequence of Panoptic Studio [96], even though it was trained only on the triadic Haggling dataset. The *Ultimatum* sequence shares similarities with the Haggling dataset such as persons taking turns and talking to each other.

For (2), we define the buyer's attention as whomever they look at, which can be easily calculated from the 3D body pose:

$$\underset{i\in\{\text{left,right}\}}{\text{argmin}}\left[n^T\left(\frac{d_i}{|d_i|}\right)\right] \tag{7.16}$$

where d_{left} and d_{right} are the directional vectors from the buyer nose to the left and right seller nose, respectively, projected onto the ground plane and n is the 2D unit vector that is perpendicular to the left eye→right eye vector of the buyer, projected onto the ground plane.

Last but not least, to determine if someone is speaking we utilize an off-the-shelf action classification network consisting of three layers of bi-directional Gated Recurrent Units. For training, we use the annotation of the Haggling dataset [99] which indicates if a person is speaking or not. The classifier achieves 87% accuracy in speech detection on the test set.

7.4.2 Multi-Person Forecasting

We first report the results for the multi-person human motion datasets MuPoTS-3D [136][1], 3DPW [196][1] and CMU-Mocap [1, 200]. We follow [200] and report the Mean Per Joint Positional Error (MPJPE) using global coordinates in Table 7.1. Our approach outperforms the methods LTD [129], HRI [130], SP [4], and MRT [200] by a large margin. While some methods perform better for the first second on MuPoTS-3D [136], our approach achieves a much lower error for all other settings and datasets. On the most difficult dataset 3DPW [196], the error is reduced by 57.6%, 55.6%, and 54.7% for 1, 2, and 3 seconds, respectively. As in [200], we also report the error of the position of the root joint and pose error in local coordinates, i.e., setting the root position for all frames to zero, in Table 7.2. The results show that our approach forecasts by far the most accurate poses and outperforms the state of the art by a large margin on all datasets. Only the position of the root joint is slightly better estimated by other methods at the beginning of the datasets MuPoTS-3D and CMU-Mocap. At 3 seconds, our approach also achieves the lowest root joint error for all three datasets.

[1] Data access and processing was conducted at University of Bonn

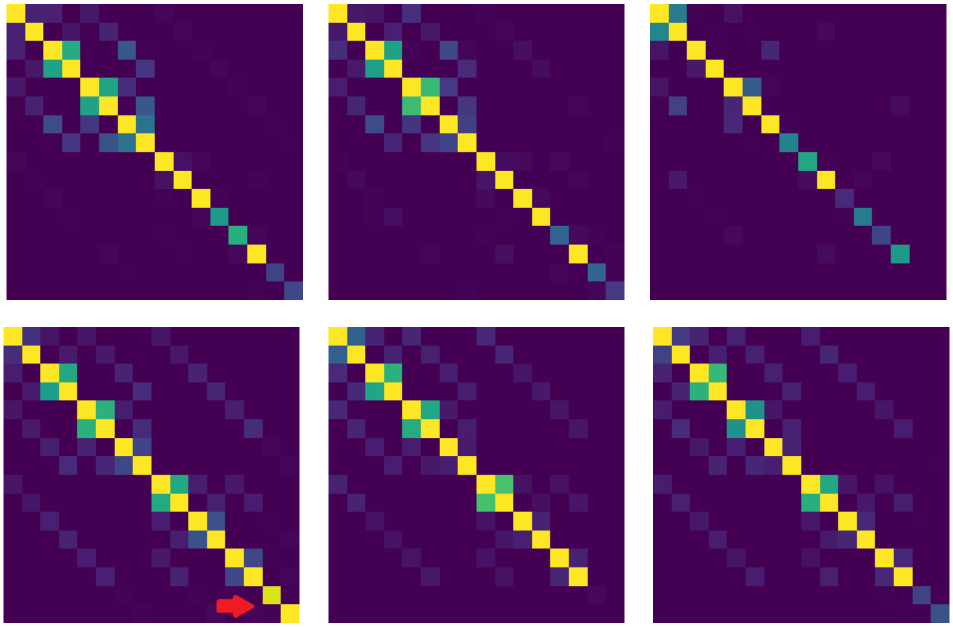


Figure 7.6: State transition matrices for the states defined in Section 7.4.1 for the ground-truth of the train (a) and test set (b). The other transition matrices are obtained by the forecast of the current state-of-the-art model MRT [200] on the test set (c) and our model without context (d), with context via attention (e), and with context via averaging (f). The more similar the transition matrix is to the test set (b), the closer it matches the test motion.

7.4.3 Multi-Person Forecasting in the Context of Social Interactions

For the remaining experiments, we evaluate our approach on the newly prepared Haggling dataset since it contains more social interactions as the other datasets. We compare our approach to Multi-Range Transformers (MRT) [200], which performed better than other approaches in Section 7.4.2. We used the publicly available source code and adjusted the approach to work with 30Hz. We kept all other settings as is.

We first report the per-frame NDMS (higher is better) at different frames in Table 8.4. Our experiments show that MRT [200] is capable of generating realistic motion for a few seconds. However, after 120 frames ($4s$) the NDMS score drops significantly. This is caused by the auto-regressive motion forecasting strategy adopted by MRT, which results in error accumulation over time. In contrast, our method continues to predict motions with good NDMS scores well into the future. As discussed in Section 7.2.2, we compare different variants of the aggregation function Γ. In terms of Γ, there are no major differences in terms of NDMS, but they all perform much better than MRT [200]. In Table 7.4, we report the average NDMS score over the entire forecast sequence. We observe that $\Gamma_{\varnothing}$, i.e., using no aggregation information yields a slightly higher NDMS score than the other versions, but the differences are very small. However, Γ_{attn} and $\Gamma_{\mathbb{E}}$ forecast much more plausible social interactions (lower SSCP) as we will discuss next.

	Train	MRT [200]	Ours ($\Gamma_{\varnothing}$)	Ours (Γ_{attn})	Ours*($\Gamma_{\mathbb{E}}$)
NDMS ↑	-	0.1015	**0.2301**	0.2270	0.2297
SSCP ↓	0.0999	0.4839	0.3576	0.3278	**0.3252**

Table 7.4: Mean NDMS [185] ↑ score over all frames.

Test set	MRT [200]	Ours ($\Gamma_{\varnothing}$)	Ours (Γ_{attn})	Ours* ($\Gamma_{\mathbb{E}}$)
1.0	0.3250	0.8812	0.8875	**0.8938**

Table 7.5: Accuracy ↑ of the buyer detection network. For MRT [200], the model randomly selects a person to be the buyer, as there is a $1/3$ chance of selecting the buyer with random chance.

7.4.3.1 Social Motion Evaluation

The SSCP scores are presented in Table 7.4 where a lower value corresponds to more plausible forecast social interactions. All our proposed variants outperform the state-of-the-art approach MRT [200]. Using no aggregation information ($\Gamma_{\varnothing}$) performs worse than the aggregation functions Γ_{attn} (7.9) and $\Gamma_{\mathbb{E}}$ (7.10), which is expected since the aggregation function passes information from other people in the scene. The average aggregation ($\Gamma_{\mathbb{E}}$) performs slightly better than the more complex multi-headed attention approach (Γ_{attn}). We conjecture that averaging bottleneck encodings over all people introduces an inductive bias to pay the same attention to everyone, which works well for modelling the haggling game. For completeness, we also report the SSCP score of the training set.

We can draw some insights about what motion each model generates by looking at the state probability transition matrices in Figure 7.6. For example, MRT [200] (Figure 7.6, bottom left) produces mostly self-loops (diagonal of transition matrix) indicating that the motion gets stuck over time. When no context information is provided ($\Gamma_{\varnothing}$), our method produces motions where all three people are talking at the same time, as can be seen in Figure 7.6 bottom left, where the last two entries (red arrow) in the transition matrix represent states with all three people talking. This is sensible as the model sees two sellers and only one buyer during training and thus it is more likely to produce motion that resembles a seller, who talks most of the time. When the context is provided, our approach overcomes this limitation as expected and rarely produces motion where all three people are talking at the same time as shown in Figure 7.6 bottom middle and 7.6 bottom right.

These observations are also confirmed when measuring the buyer detection accuracy on the forecast motion, which is reported in Table 7.5. The detector fails to identify the correct buyer in the sequences that are forecast by MRT [200] and it nearly chooses the buyer at random with $1/3$ accuracy. This confirms that MRT does not forecast socially consistent sequences where the social role of the persons, namely buyer or seller, is preserved. In contrast, our method predicts motion where the buyer can be easily determined most of the time. As for the SSCP scores reported in Table 7.4, $\Gamma_{\mathbb{E}}$ performs best.

In summary, the aggregation function $\Gamma_{\mathbb{E}}$ outperforms the other aggregation functions on the Haggling dataset [99] as it produces the most socially plausible motion according to our Symbolic

2 seconds			10 seconds		
gt	ours	MRT [200]	gt	ours	MRT [200]
0.948	0.567	0.520	0.953	0.582	0.01

Table 7.6: User study on the Haggling dataset.

Social Cues Protocol while also generating highly plausible 3D body motion.

7.5 User Study

We provide a user study in Table 7.6 were 11 subjects were asked to judge short (2 seconds) or long (10 seconds) sequences of forecast human motion of 24 randomly selected sequences. We showed randomly the ground-truth (gt), results from our approach, or from [200]. The subjects rated the sequences by realistic (1), unsure (0.5), or unrealistic (0). We report the mean rating (higher is better). The results are consistent with the results in the paper. MRT [200] performs well for short time horizons but accumulates artifacts over time, making it look unrealistic. Since 10s are slightly better to judge for humans than 2s, the rating for ground-truth and ours is slightly higher for 10s.

7.5.1 Ablation Study

Average velocity over time Freezing or unrealistically expanding motion are common failures in human motion anticipation. While NDMS penalizes in contrast to MPJPE errors in the velocity, visualizing the average motion velocity can give interesting insights. In Figure 7.7, we plot the average velocity over all frames for all our summary function variants, the test set, and the state-of-the-art method MRT [200]. Note that the beginning of the sequence has a higher velocity due to people walking into the scene. We observe that MRT suffers from error accumulation caused by the auto-regressive inference scheme. The velocity produced by our motion tightly follows the test set velocity for roughly 250 frames after which the test set velocity is slightly larger. We attribute this to the higher degree of stochasticity of real motion, which results in sudden jerks and swings that increase the average velocity.

NDMS score over time In Figure 7.8, we visualize how the NDMS scores of all proposed variants and MRT evolve over time. For reference, we also calculate the NDMS score of the training data which is guaranteed to be realistic. Note that NDMS is 1 for the observed part of the test sequences. As shown in the figure, our method achieves almost the same level of realism as the training data while the quality of MRT slowly degenerates over time.

Anticipating more than three people We have trained and evaluated our model on the Haggling dataset where each sequence consists of triadic interaction. However, the fully convolutional nature of our approach as well as the order-invariance of the summarization function Γ allow us to forecast any number of people. To demonstrate this capability, we predict 7 people from the *Ultimatum* sequence of Panoptic Studio [96] using only the model trained on the Haggling dataset. This works well because the Haggling and *Ultimatum* sequences share many social behaviors, such as turn-taking, talking, and paying attention while standing in a circle. Our results can be seen in Figure 7.5 where our model is able to predict realistic motion for 7 people, even though it was trained only on the triadic Haggling dataset. Our method can not only be applied to multiple persons, but also

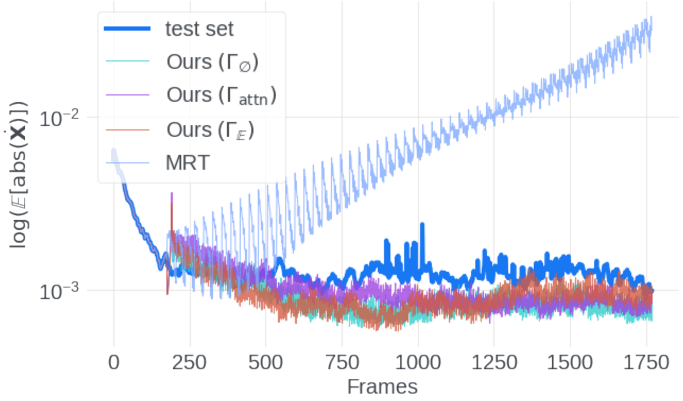


Figure 7.7: Average velocity over time for the entire test motion $\mathcal{X}$ and generated motions $\hat{\mathcal{X}}$. The x-axis represents the frames while the y-axis represents the (log) average velocity of the data.

steps	1	10	100	500	800	1000
NDMS	0.103	0.170	0.176	0.207	0.210	0.230

Table 7.7: Impact of the number of diffusion steps on the Haggling dataset.

to persons interacting with an object. Figure 7.9 shows an example from the Sports dataset from Panoptic Studio [96, 98].

Impact of Number of Diffusion Steps We evaluate the impact of the number of diffusion steps in Table 7.7. We follow the original diffusion implementation and train our method with 1000 diffusion steps. We observe that the motion quality saturates after 500 diffusion steps.

7.6 Conclusion

In this work, we present Social Diffusion, a stochastic multi-person motion anticipation model. The approach not only forecasts realistic motions on the individual level, but also plausible social interactions where the social roles of individuals are preserved over time. The approach is very flexible. It can be used for short and long-term forecasting and can be applied to larger groups than observed during training. As a second contribution, we proposed a new evaluation protocol to measure the realism of forecast social interactions. We furthermore derived a dataset for multi-person social interaction forecasting from the Haggling dataset [99] where the persons have different social roles that impact their behavior. We evaluated our approach on four multi-person datasets and demonstrated that our approach outperforms the state-of-the-art for short-term and long-term anticipation both in realism of forecast motion and social interaction. The approach has still some limitations. For instance, the global positions of the root joints can be better estimated. Future directions also include extending the model to predict motions of dynamic groups of people, e.g., at a cocktail party where any individual can freely disengage from the current conversation group and join another one.

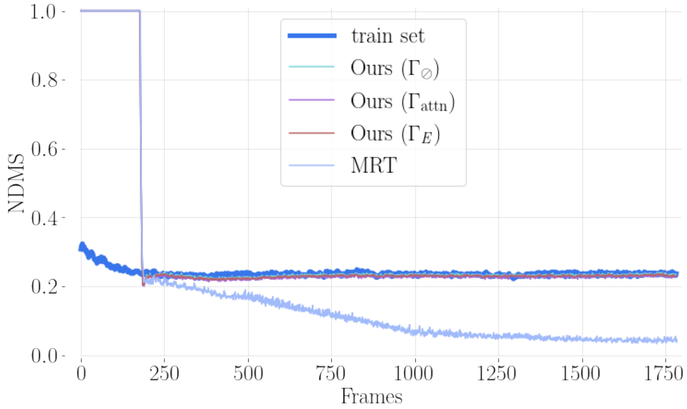


Figure 7.8: Average NDMS [185] score ↑ of all proposed variants, MRT, and the train set over time.

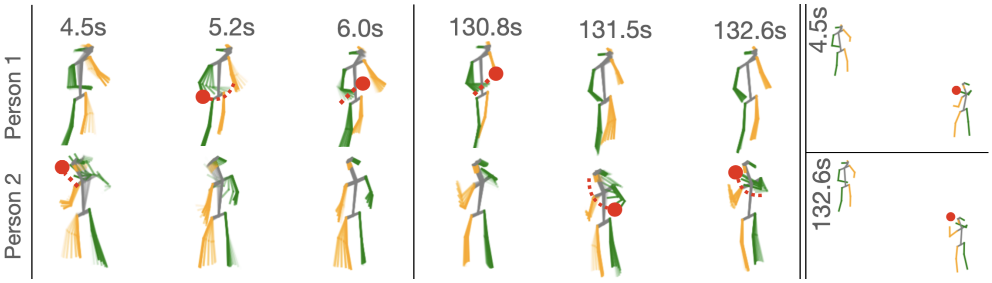


Figure 7.9: Forecast motion on the *Sports* sequence from [96, 98], consisting of two persons throwing a ball. The ball and its past trajectory are marked in red and each column represents 1 second of motion. On the right hand side, we show the global view, which shows that the distance between the persons is correctly maintained.

CHAPTER 8

Humans in Kitchens: A Dataset for Multi-Person Human Motion Forecasting with Scene Context

The methods introduced in Chapter 6 and Chapter 7 utilize different modes of human motion, namely intention and social cues. For realistic human motion all relevant modes, intention, social cues and even scene geometry must be considered but no existing dataset does cover all modes. In this Chapter we built the first large-scale human motion dataset that covers all those modes. Our novel dataset was recorder in four kitchens at the University of Bonn, and at times contains up to 16 persons in the scene at the same time. The recordings are continuous shots of 2h recordings at 25Hz and 82 activities as well as the 3D scene geometry are annotated. This dataset required significant efforts in data annotation and data cleaning and the 3D human poses where extracted using the methods introduced in Chapter 4 and Chapter 5. We show that our methods Intention RNN (Chapter 6) and Social Diffusion (Chapter 7) outperform state-of-the-art on long-term human motion forecasting.

Individual Contribution: The following chapter is based on the publication [187]:

Humans in Kitchens: A Dataset for Multi-Person Human Motion Forecasting with Scene Context
Julian Tanke, Oh-Hun Kwon, Felix Mueller, Andreas Doering and Juergen Gall
Conference on Neural Information Processing Systems (NeurIPS), 2023.

This publication was done by Julian Tanke with strong support from Oh-Hun Kwon, Felix Mueller and Andreas Doering. Juergen Gall provided scientific guidance and supported this work with very valuable feedback and suggestions. The idea for a large-scale human motion dataset recorded in kitchen environments was proposed by Juergen Gall. Julian Tanke organized the recordings and Julian Tanke and Oh-Hun Kwon run the recordings during the shoots. Oh-Hun Kwon synchronized the videos and calibrated the cameras which were later fine-tuned by Julian Tanke. Julian Tanke oversaw the building of an annotation tool as well as the annotation of the dataset. The 2D multi-person pose estimation module, which is a crucial component of the 3D pose estimation method [183, 109], was supplied by Andreas Doering. Julian Tanke implemented the dataset API and organized the dataset structure. Oh-Hun Kwon and Felix Mueller run baseline methods against the API for the experiments in forecasting. The paper was written by Juergen Gall and Julian Tanke.

Contents

8.1 Introduction

Understanding and anticipating human motion within groups is very challenging and essential in the context of socially-compliant autonomous robots [2, 14, 26, 100, 140, 145, 8, 104], as they must possess the ability to understand and respond appropriately to human behavior. Moreover, this topic has relevance in the fields of neuroscience and social sciences [35, 22, 91], as it enables the development of computational models that explore the perception of others' behavior and its influence on one's own behavior. For example, imagine a group of persons sitting on a sofa and another person walking towards it, one would expect the person to sit down on an unoccupied seat on the sofa. Similarly, two persons in front of a whiteboard are expected to discuss their ideas. However, it is more plausible that only one of them writes onto the whiteboard while the other observes.

Capturing those complex social and object-person interactions necessitates a large dataset for effective training and evaluation. Such dataset must possess three essential characteristics: (a) it should encompass natural interactions between multiple individuals recorded in a real environment (b) it should have annotated scene geometry to account for human-object interactions, and (c) it should include annotated per-person action labels to balance the evaluation and to avoid a strong bias towards simple activities like standing, walking, and sitting. Currently, such dataset with 3D human poses does not exist as shown in Table 5.1. The largest multi-person human motion dataset is Panoptic Studio [96]. The dataset, however, has been recorded in a studio. Although the persons interact, they mainly stand due to the small recording area. It also does not include a real environment and the subjects need to act in an unfamiliar environment with many cameras and light sources, which can induce a behaviour that differs from real-life behaviour.

In order to address these issues, we propose *Humans in Kitchens*, a large-scale multiple person 3D human motion dataset with annotated scene geometry and per-person activities. Our dataset consists of more than 4M unique poses of 90 individuals in total. We recorded persons in four real kitchen for over 7.3h. Each of the four kitchen sequences was continuously recorded for 1.5h to 1.9h. Persons could freely enter or leave the scene and received minimal instructions, resulting in a very natural behavior and interactions. For each scene, we annotated objects that people may interact with, such as sinks, dishwashers, chairs, or whiteboards - and objects that determine the geometry of the scene, such was walls. Some interactive objects, such as chairs and kettles, are annotated per frame as they may be moved around the scene. The scenes span between 38m^2 to 80m^2, which is much larger than the 19m^2 of Panoptic Studio [96]. The maximum number of individuals in the scene at the same time is 16, twice the maximum number of persons in Panoptic Studio. For each person, we annotate their frame-wise activity, such as *walking*, *sitting*, *writing on whiteboard* or *making coffee*. We represent humans by the SMPL [124] body model, which includes the 3D skeleton pose. We

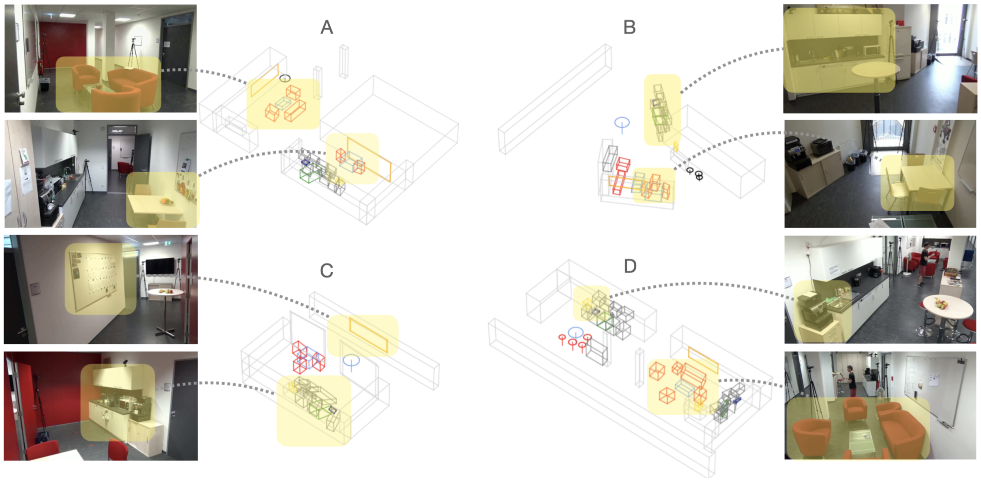


Figure 8.1: Overview of each of the four kitchens. Each kitchen contains sofas and chairs (red 🟥), tables (blue 🟦) and at least one whiteboard (orange 🟧), fridge (green 🟩), coffee machine (yellow 🟨) and sink (dark blue 🟦). Best viewed using zoom in PDF viewer.

belief that Humans in Kitchens will contribute to advance multi-person human motion forecasting as well as modeling scene context for social behaviour understanding and anticipation.

We provide details on the acquisition and annotation of the dataset, dataset statistics, and evaluate state-of-the-art methods for multi-person human motion forecasting. We further discuss limitations and potential risks of the dataset.

8.2 Humans in Kitchens

In order to obtain a dataset that on one hand contains realistic behaviour of multiple interacting persons and on the other hand is GDPR conform, which includes the informed consent of each subject in the dataset, we followed a different approach than previous datasets with 3D human poses. Instead of asking subjects to perform certain motions or games in a recording studio, we collected data in four real office kitchens for a duration between 1.5h and 1.9h. Persons were are allowed to enter and leave the scene and persons that did not want to participate were asked to use another kitchen during the recording session. Despite of the informed consent sheet, only a subset of the participants received minimal instructions as we will discuss in Sec. 8.2.1.1. The dataset acquisition and annotation process will be described in Sec. 8.2.1 and Sec. 8.2.2.

8.2.1 Dataset Acquisition

For the data acquisition, 11-12 calibrated cameras that were synchronized via the audio signal have been used. The recording has been performed in four different office kitchens on different days. The layout of the kitchens is shown in Figure 8.1. While the hardware setup is described in [184], the corresponding repository [184] contains only the raw data where the estimated 3D human poses are very noisy due to severe occlusions and limited view of each camera. The raw data itself can thus not

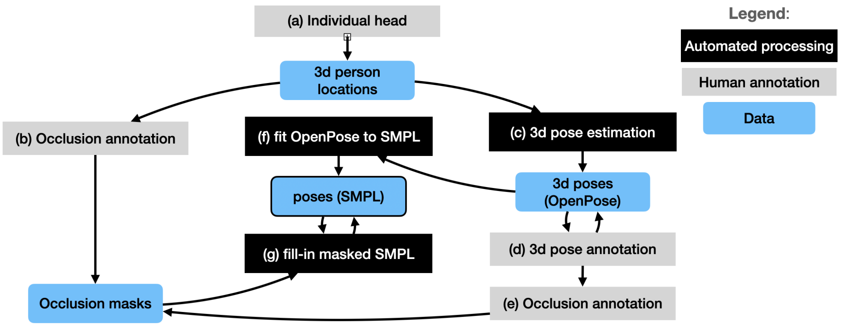


Figure 8.2: Overview of the 3D human pose annotation process. First, (a) the heads of individual persons are annotated in each frame, represented as a 3D point in global coordinates and unique identity. Given the person annotations, we annotate occlusions for each person and frame (b) and automatically extract 3D human poses (c). We manually correct 3D human poses (d) or mask them (e) if even the annotators cannot correct them. We estimate SMPL parameters (f) and inpaint masked regions (g) to obtain for each a full SMPL pose representation. Black boxes represent automated processes without human intervention, gray boxes represent human annotation processes and blue boxes represent data.

be used for training or evaluation. In Sec. 8.2.1.2, we describe how the raw data has been manually annotated to obtain the Humans in Kitchens dataset for multi-person human motion forecasting with scene context.

8.2.1.1 Behavior Protocol

For each of the four recordings, a cake has been provided to attract participants to the kitchen. To facilitate behavior as natural as possible, we provided only minimal instructions to 10 persons in each recording, where they were asked to randomly perform 3 of the following activity at any time and in any order:

Make coffee: Prepare a coffee and drink it; **Make tea**: Use the kettle to prepare a tea and drink it; **Eat cake**: Take a slice of the cake and eat it; **Eat fruit**: Eat some of the provided fruits; **Drink water**: Drink water from the tap; **Explain on Whiteboard**: Explain a topic of your choice on the whiteboard; **Use Laptop**: Work on a laptop; **Use Microwave**: Use the microwave to heat milk for coffee; **Read paper**: Read a paper; **Make a phone call**: Make a phone call; **Clean dish**: Clean your dish(es) in the sink; **Place in dishwasher**: Put used dishes in the dishwasher.

While 10 persons were instructed, the other persons present in the scene were not instructed to perform any of the above mentioned activities. However, each person was allowed to perform any activity, e.g., anyone could make a coffee, eat a cake or clean dishes.

8.2.1.2 Pose and Activity Annotation

We annotated the 3D human poses of each person in each frame where each person has a unique identity through the sequence. To extract 3D human poses in our very challenging environment, we annotated the 3D poses in five phases:

Manual nose annotations: We first manually annotated each individual in each frame at their nose in the 3D scene, using a custom annotation tool. This also allowed us to re-identify persons who left the scene but later returned to the recording. We verified the correct annotations in a second pass where additionally the 82 activities where annotated per frame per person. In total, annotating this phase took around 2,000 person hours.

Automated human pose estimation: We ran an off-the-shelf 3D human pose estimation method [183] to extract the 3D human poses from the multiple cameras. We match the estimated 3D poses to the closest manually annotated nose and drop all leftover poses. If a 3D nose annotation is not matched to an estimated 3D pose, we linearly interpolate between the previous and the next frame. The extracted 3D human poses are represented as 3D skeletons using the OpenPose keypoints [32].

Manual occlusion masking and human pose correction: The automated pose estimation method fails in heavily occluded scenes, requiring us to manually correct the 3D skeletons in those frames. We manually annotated and corrected 20,000 3D poses, around 0.5% of the dataset. Additionally, we manually annotated occlusion masks for head, upper body and lower body where even human annotators were not able to determine the correct 3D joint positions. Differentiating between head, upper and lower body is a compromise between accuracy and annotation speed, as often only certain parts of a person, e.g., the legs, where occluded. In total, 6% of the poses have at least one body part masked. This annotation phase required 600 person hours.

Fitting SMPL: We use an off-the-shelf optimization framework [151] to extract SMPL parameters from the 3D OpenPose keypoints that have been annotated in the previous steps. We extract the SMPL parameters for all poses, even the masked once. We can do this as the masked poses still represent valid 3D poses as we just interpolate between the known frames.

SMPL Inpainting: We utilize an unconditioned human motion diffusion model (MDM) [189], trained on AMASS [128], to inpaint the masked poses. For this, we subsample AMASS to 25Hz to match the frame-rate of our dataset. During inference of the diffusion model, we replace the unmasked keypoints by the annotated keypoints at each diffusion step. In this way, the annotated and verified 3D keypoints remain unchanged, but the occluded and masked parameters will be filled by the diffusion model.

Figure 8.2 provides an overview of our pose annotation process.

8.2.1.3 Scene Annotation

We annotate the scene geometry either as 3D box or as cylinder, where we annotate trash bins, stools and circular tables as cylinders and everything else as box. In each of the four kitchens we annotate the following 13 objects: **Whiteboard**, **MicrowaveKettle**, **Coffee Machine**, **Table** (sofa table, bar table, kitchen table), **Sittable** (sofas, arm chairs, chairs, and stools), **Cupboard** (floor and hanging cupboards), **Occluder** (walls and pillars), **Dishwasher**, **Drawer**, **Sink**, **Trash**, and **Out-of-Bound-Marker**, which marks the boundary of the visible area. The objects in the scene are annotated per frame since the objects can move during the recording. Fig. 8.1 shows four examples.

8.2.2 Statistics

Our dataset is a large-scale multi-person motion dataset, recorded at 25Hz, with over 4M individual 3D human poses of 90 individuals and over 650k frames, a total of 7.3h of recording, as summarized

	A	B	C	D	total
# frames	128,959 (1.43h)	179,097 (1.99h)	175,392 (1.95h)	176,264 (1.96h)	659,712 (7.33h)
# annotated poses	573,253	1,132,422	908,380	1,415,189	4,029,244 (44.76h)
mean persons / frame	4.42	6.32	5.17	7.97	-
median persons / frame	4	6	5	7	-
max. persons / frame	9	14	9	16	-
# individuals	18	32	16	24	90
surface area	76.35m^2	57.22m^2	38.28m^2	80.40m^2	-
# camera views	11	11	12	12	-
# scene objects	37	40	29	50	-

Table 8.1: Dataset statistics for the four kitchen environments A, B, C and D.

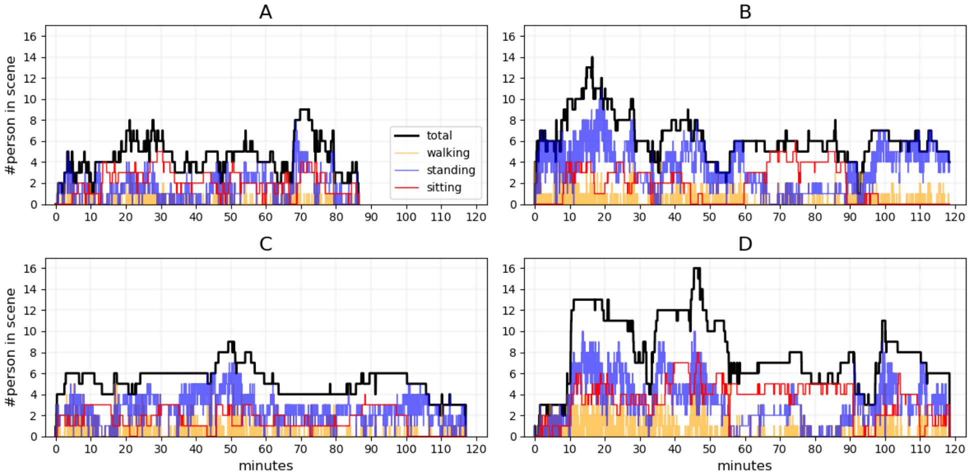


Figure 8.3: Number of persons that are *Walking*, *Standing* or *Sitting* at a frame. The x-axis represents the elapsed time in minutes while the y-axis represents the number of person that perform the corresponding activity. The black curve is the total number of persons at a frame.

in Table 8.1. Compared to other datasets with 3D human poses, it contains more numbers of persons in the scene and more diverse activities recorded in a real environment as summarized in Tab. 5.1. Furthermore, the dataset not only includes the context of a static environment, but also moving objects that have been annotated. Each pose is further labeled with one or multiple activities out of 82 total actions, for example, sitting in a chair, writing on a whiteboard or washing hands[1]. Scene geometry is annotated per frame as either a 3D box or as 3D cylinder as well as with an object class (13 in total), e.g., coffee machine, table or whiteboard. The dataset was recorded in 4 kitchens, A, B, C and D, with common scene geometry such as coffee machines, chairs and whiteboards, but with different room layouts, as can be seen in Figure 8.1. The number of annotated objects varies between 29 and 50. Each kitchen sequence has been recorded continuously, taking between 1.5h (A) to $2h$ (B, C, D). While there are in average between 4.42 and 7.97 persons at the same time visible in a scene,

[1] For a full list see supplementary material

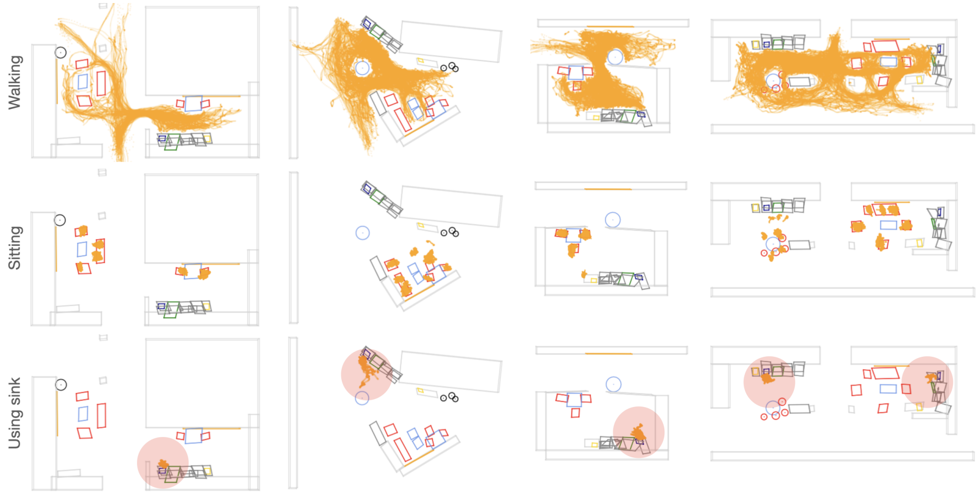


Figure 8.4: Occurrence map for the activities *Walking*, *Sitting* and *Using sink*. The maps are generated by plotting the location of the root joint of each person in the entire recording when persons perform the corresponding activity. For the last activity, *Using sink*, we highlight the location with a red circle as the activity only occurs close to the sink (dark blue).

Action	A	B	C	D	sum
Walking	39,454	73,051	79,195	119,230	310,930 (3.45h)
Sitting	265,521	314,791	263,652	631,673	1,475,637 (16.39h)
Standing	209,525	635,835	464,940	569,712	1,880,012 (20.89h)
Leaning	25,491	46,538	56,846	44,685	173,560 (1.9h)
Kneeling	0	702	0	50	752 (30s)

Table 8.2: Number of frames and persons per dataset for the posture activities *Walking*, *Sitting*, *Standing*, *Leaning* and *Kneeling*.

the number of persons varies largely during a sequence since persons enter and leave the scene as shown in Figure 8.3.

From the 82 annotated activities, we define 5 as posture activities: *Walking*, *Sitting*, *Standing*, *Leaning* and *Kneeling*. At each point in time, a person exhibits one of the 5 postures. We differentiate between *Leaning* and *Standing* by defining that a person leans if the person's weight is supported by a scene object, e.g., a cupboard. *Kneeling* is very rare and is only briefly observed in two kitchens: we decided to annotate it for completeness as it would not fit any of the four other postures. The postures greatly vary in frequency as we show in Table 8.2. The most common postures are *Sitting* and *Standing*, which is expected from natural social human interactions. In Figure 8.3, we plot the three most common postures, *Walking*, *Standing* and *Sitting* over time. We observe that in B and D there is a time window at the end of the recording where most persons sit. This is not observed in A and C. This shows that the distributions even of the basic posture activities varies over time and from scene to scene. In Figure 8.4, we show a bird eye view for all four kitchens and plot where three example activities occur. As expected, *Walking* covers almost the entire scene while human-object

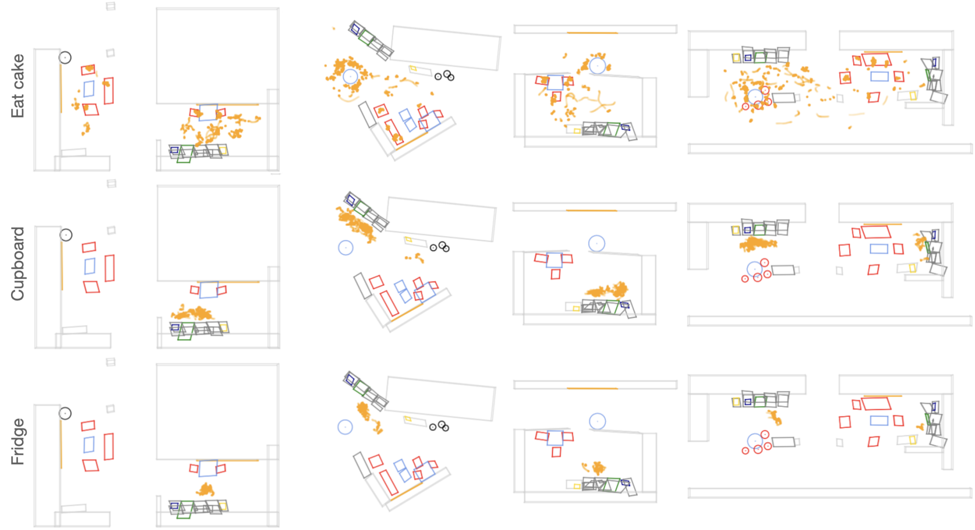


Figure 8.5: Occurrence map for the activities *Eat cake*, *Cupboard* and *Fridge*. The maps are generated by plotting the location of the root joint of each person in the entire recording when persons perform the corresponding activity.

interactions such as *Sitting* and *Using sink* are localized at the corresponding scene objects. Although chairs move in the scene, they are not moved to completely different locations since the kitchens offer sufficient chairs.

8.2.3 License and Consent

The dataset and API are free to download[2] and use non-commercially. The API is under MIT-License while the dataset utilizes a custom license. All subjects signed forms consenting that recordings of them or derived of them can be used for non-commercial research purposes. The recording has been approved by the ethical review committee of the University of Bonn. In contrast to the raw data [184], Humans in Kitchens does not contain personally identifiable information. The raw data is also accessible[3], but requires to sign a license agreement and is subject to export regulations. Note that the raw data is not required for using Humans in Kitchens. The data does not contain any offensive content.

8.3 Experiments

We evaluate various state-of-the-art methods on the human motion forecasting task of our dataset. More precisely, the goal is to learn a function f that takes as input a human pose sequence $\mathbf{X}^{1:t} = (\mathbf{x}_1^{1:t}, \mathbf{x}_2^{1:t}, \cdots, \mathbf{x}_n^{1:t})$ of n persons, where $\mathbf{x}_i^{1:t} \in \mathbb{R}^{t\times(29\times3)}$ represents a 3D motion sequence of t

[2] https://drive.google.com/drive/folders/1cmR2M0lPhQsvGfT2aXg3WrY8Qvahzd9G

[3] `https://github.com/bonn-activity-maps/bonn_activity_maps`

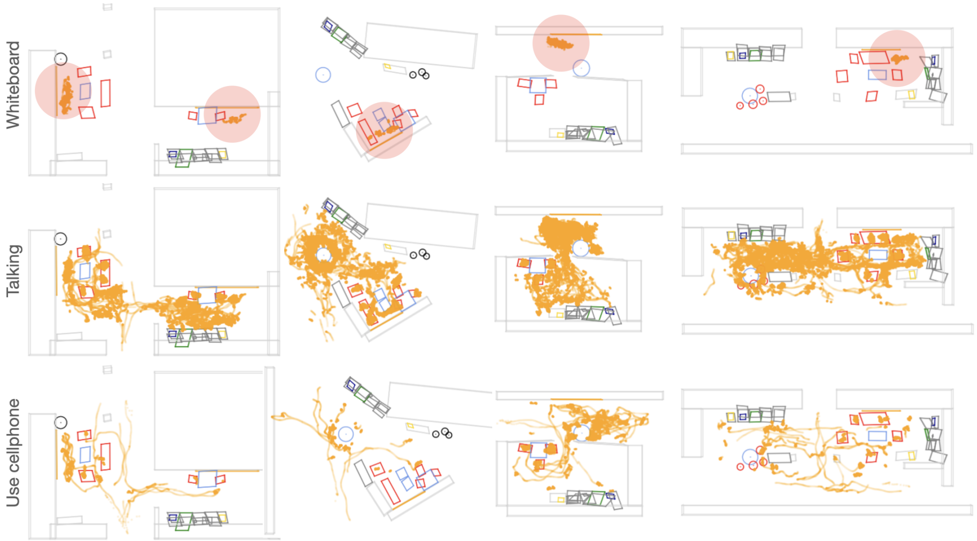


Figure 8.6: Occurrence map for the activities *Whiteboard*, *Talking* and *Use cellphone*. The maps are generated by plotting the location of the root joint of each person in the entire recording when persons perform the corresponding activity. For the first activity, *Whiteboard*, we highlight the location with a red circle as the activity only occurs close to the whiteboard.

frames for person i, and forecast the future motion for all n persons in global coordinates:

$$\hat{\mathbf{X}}^{t+1:T} = f(\mathbf{X}^{1:t}), \quad \hat{\mathbf{X}}^{t+1:T} \in \mathbb{R}^{T-t \times n \times (29 \times 3)}. \tag{8.1}$$

As our dataset contains natural behavior over very long time, we select 2 interesting motion activities, namely *Walking* and *Sitting down*, and 4 interesting human-object interactions with the objects *Whiteboard*, *Sink*, *Cupboard* and *Coffee Machine* for evaluation. Note that most of these actions involve social interactions since the persons sit in groups and discuss at the whiteboard. For training, we use the kitchens `A`, `B` and `C` and we evaluate on `D`, which is the largest among the four kitchens. For the human-object interactions, we select the last observed frame t as the first frame of the annotated action, while for the motion activities we select frame $t - 10$ such that a few frames of the motion are already observed. Overall, we sample all occurrences of the given activity in the test set. We only evaluate the person that performs the activity and not for any other person in the scene. This allows to report accuracy per activity, but it still requires to model the context of the other persons. We consider two distinct protocols: *short-term* and *long-term* motion forecasting. In the short-term protocol, we employ the widely used Mean Per Joint Positional Error (MPJPE) metric [132] to measure the positional disparity between the predicted motion and the ground truth. For the long-term protocol, we use the Normalized Directional Motion Similarity Score (NDMS) (Chapter 6), which effectively assesses the quality of motion sequences of any given length.

Baseline methods: We evaluate 4 recent single-person human motion forecasting methods, namely siMLPe [73], CHICO [170], pgbig [127] and History-Repeats-Itself [130], and the multi-person forecasting method Multi-Range Transformers (MRT) [200]. While the single-person ap-

	5	15	25	5	15	25	5	15	25	5	15	25	5	15	25	5	15	25
		walking			sitting down			whiteboard			sink			cupboard			coffee	
MRT* [200]	0.40	0.91	**1.40**	0.38	0.97	1.42	0.24	0.56	0.80	0.29	0.80	1.16	0.28	0.61	**0.89**	0.26	0.69	1.19
siMLPe [73]	0.39	0.97	1.59	0.37	0.98	1.40	0.24	0.53	0.72	0.17	0.43	0.59	0.23	0.64	0.98	0.22	0.55	0.88
CHICO [170]	0.37	0.88	1.42	0.36	0.96	1.36	0.27	0.55	0.69	0.19	0.46	0.61	0.23	0.65	1.05	0.22	0.55	**0.86**
pgbig [127]	0.34	0.85	**1.40**	0.34	0.87	**1.22**	0.23	0.52	**0.66**	0.16	0.42	0.58	0.20	0.59	0.93	0.20	0.54	**0.86**
HistRep [130]	**0.30**	**0.83**	1.57	**0.30**	**0.84**	1.23	**0.19**	**0.48**	0.68	**0.12**	**0.40**	**0.57**	**0.16**	**0.57**	0.99	**0.17**	**0.50**	0.95
Intention (Chapter 6)	0.39	0.98	1.61	0.35	0.97	1.36	0.25	0.55	0.79	0.19	0.49	0.65	0.20	0.62	0.99	0.23	0.59	1.01
Social Diffusion* (Chapter 7)	0.37	0.96	1.59	0.34	0.92	1.30	0.24	0.53	0.73	0.19	0.47	0.63	0.21	0.59	0.91	0.22	0.58	0.92

Table 8.3: MPJPE ↓ in dm. Methods denoted with * forecast all persons at the same time.

	1	25	100	200	250	1	25	100	200	250	1	25	100	200	250
			walking					sitting down					whiteboard		
MRT* [200]	0.81	0.34	0.18	0.13	0.12	0.81	0.28	0.14	0.11	0.10	0.80	0.26	0.14	0.11	0.10
siMLPe [73]	**0.87**	0.41	0.24	0.20	0.19	0.87	0.37	0.22	0.19	0.18	0.88	0.35	0.21	0.18	0.17
CHICO [170]	0.84	0.40	0.25	0.20	0.19	0.85	0.36	0.22	0.19	0.18	0.86	0.35	0.22	0.18	0.17
pgbig [127]	0.86	0.43	0.25	0.21	0.20	0.87	0.38	0.23	0.20	0.19	0.87	0.37	0.23	0.20	0.19
HistRep [130]	**0.87**	0.46	0.21	0.15	0.14	**0.88**	**0.43**	0.19	0.14	0.13	**0.89**	**0.43**	0.19	0.14	0.13
Intention (Chapter 6)	0.86	**0.48**	0.31	0.30	0.23	0.84	0.42	0.29	0.28	0.30	0.85	0.42	0.30	0.31	0.27
Social Diffusion* (Chapter 7)	0.83	0.40	**0.34**	**0.31**	**0.30**	0.83	0.39	**0.33**	**0.32**	**0.31**	0.88	0.38	**0.34**	**0.31**	**0.29**
			sink					cupboard					coffee		
MRT* [200]	0.81	0.26	0.13	0.10	0.10	0.81	0.29	0.15	0.11	0.10	0.82	0.30	0.16	0.12	0.11
siMLPe [73]	**0.90**	0.36	0.22	0.19	0.18	0.86	0.37	0.22	0.19	0.18	**0.86**	0.38	0.22	0.19	0.19
CHICO [170]	0.87	0.34	0.22	0.19	0.18	0.84	0.36	0.22	0.19	0.18	0.83	0.36	0.23	0.19	0.18
pgbig [127]	0.89	0.37	0.23	0.20	0.20	0.86	0.39	0.23	0.20	0.19	0.85	0.39	0.24	0.21	0.20
HistRep [130]	**0.90**	**0.43**	0.19	0.14	0.13	**0.87**	**0.43**	0.19	0.14	0.13	**0.86**	**0.44**	0.20	0.15	0.14
Intention (Chapter 6)	0.86	0.41	**0.34**	0.28	0.20	0.82	0.40	0.26	0.23	0.22	0.84	0.41	0.27	0.26	0.24
Social Diffusion* (Chapter 7)	0.85	0.39	0.33	**0.29**	**0.21**	0.82	0.40	**0.32**	**0.25**	**0.24**	0.85	0.42	**0.29**	**0.28**	**0.25**

Table 8.4: NDMS ↑. Methods denoted with * forecast all persons at the same time.

proaches forecasts the motion of each person independently, MRT forecasts the motion of all persons jointly. The source code of all methods is publicly available and we utilize the original hyperparameters. We only adjust the input and output dimensions to fit our skeleton representation. The adapted source code of all methods is publicly accessible via our API. We normalize the human poses in a pre-processing step as in Chapter 6 such that the root joint of the last observed frame t is translated to the origin and the hip is axis-aligned with the x-axis. After the forecasting, we apply the inverse transformation to convert the forecast motion back into global 3D coordinates. For MRT, we use the normalization proposed in [200], i.e., we shift all kitchens so that the mean pose location is at the origin to prevent drift.

We further utilize two generative models, single-person Intention-based RNN [185], which we describe in Chapter 6 and multi-person Social Diffusion [186], which we describe in Chapter 7 of this thesis. For the Intention-based RNN we utilize the pose-specific labels *Standing*, *Walking*, *Sitting*, *Sitting down*, *Standing up*, *Leaning*, *Leaning down*, and *Kneeling* as they are guaranteed to be exclusive and annotated at every frame.

Short-Term Forecasting: For short-term forecasting, the methods receive 50 frames (2s) of input motion and forecast 25 frames (1s). As metric we utilize MPJPE. The results in Tab. 8.3 show that History-Repeats-Itself [130] performs best for the first 15 frames, but at frame 25 pgbig [127] performs best for most activities. MRT [200] performed best on the *Cupboard* sequences at frame 25. We will see in the long-term experiments that History-Repeats-Itself is very strong for very short time horizons but not suitable for longer sequences, whereas pgbig [127] is a more general approach that performs well also for longer horizons.

In general, the lowest errors are observed for *Whiteboard*, *Sink*, *Cupboard* and *Coffee* since the global position changes less than for the activities *Walking* and *Sitting down*.

Long-Term Forecasting: For long-term forecasting, the methods forecast 250 frames (10s) given 50 frames (2s). In general, we suggest for future works to use 250 frames for the observation as well since restricting the observations to 50 frames is not necessary. However, state-of-the-art approaches operate on a fixed input window, an important hyper-parameter, that sometimes can only be changed by modifying the architecture. We thus kept the input sequence as for the short-term forecasting. As metric, we utilize NDMS with kernel size 8 [185]. The results in Tab. 8.4 show that all methods produce a reasonable motion up to 1 second but the quality deteriorates afterwards. Similar to the short-term forecasting, History-Repeats-Itself [130] performs best for the shorter time horizons. We find that generative approaches such as Intention RNN (Chapter 6) and Social Diffusion (Chapter 7) perform best for longer time horizons as they better model the underlying human motion. We find that the Diffusion-based approach outperforms the Adversarial-based approaches MRT [200] and Intention RNN (Chapter 6). We attribute that to the fact that Diffusion-based training is more stable than adversarial training, resulting in a better coverage of modes. Interestingly, MRT [200], which is one of two multi-person method, performs worse than the other methods. The difference might be the normalization used in MRT, which is based on the mean pose and not the last observed pose. Overall, the results show that modeling the interactions of multiple persons is not well handled by the current state-of-the-art. This is somewhat mitigated by better pose normalization in Social Diffusion, which also forecasts multiple persons at the time. Furthermore, none of the current approaches is able to incorporate scene context. Humans in Kitchens thus opens new research directions to study multi-person human motion forecasting with scene context.

8.4 Conclusion

In this chapter, we have introduced Humans in Kitchens, a significant advancement in the field of human motion datasets. By capturing over 4 million unique poses from 90 individuals across four real kitchen environments, this dataset is uniquely positioned to address the critical need for real data in the study of multi-person interactions. Unlike previous datasets, such as Panoptic Studio, which are limited by studio settings and constrained interaction spaces, Humans in Kitchens was recorded in a real environment which was known to the actors beforehand, resulting in more natural behavior. The dataset's design, emphasizing minimal interference and real-world environments, ensures that the recorded behaviors reflect genuine social and object-related interactions. Additionally, the inclusion of detailed annotations for scene geometry and per-person activities allows for a nuanced analysis of human-object and human-human interactions, addressing the lack of such detailed annotations in existing datasets. This will undoubtedly aid researchers in better modeling and predicting human behavior in various contexts, thereby advancing the state of the art in human motion forecasting. We have also presented two baseline benchmarks for short and long-term motion forecasting based on our dataset, setting a standard for future research in this area.

Despite its strengths, we recognize the dataset's limitations and potential risks, such as the representativeness of the sample and the ethical considerations surrounding data collection and use. These aspects are crucial for guiding future improvements and ensuring responsible utilization of the data.

CHAPTER 9

Conclusion

Contents

9.1 Summary

While remarkable progress has been made in short-term human motion forecasting of around 1 second, anticipating motion beyond at most 4 seconds remains largely unexplored. One reason for this is the inherent stochastic nature of human motion and its dependencies on a range of external factors such as intentions, environmental context, and social interactions, which are not adequately captured in current state-of-the-art models. Furthermore, existing datasets tend to focus on isolated aspects of human motion rather than providing a holistic representation that encompasses the full array of elements essential for realistic human motion simulation. This thesis addressed these more challenging cases of 3D human motion forecasting for single and multiple persons. We proposed methods that can anticipate motion over much longer time horizons than current state-of-the-art methods, established a large-scale dataset and benchmark, developed comprehensive evaluation metrics, and conducted extensive experiments to analyze the developed baselines and approaches.

In case of single human motion we prevent motion freeze-up, a common motion forecasting artifact, and mode collapse, a common artifact in generative models, by first forecasting a humans intention, represented as discrete per-frame signals. Utilizing this we are able of forecasting realistic motion of much longer time horizons, much longer than any state-of-the-art model is capable of. Furthermore, we introduce a new evaluation score that takes into consideration both structure of poses and the actual motion. We conduct a user study to confirm that our score correlates stronger to human perception than any other human motion quality score. Crucially, the score function does not require a ground-truth sequence but instead utilizes the test set as representation for valid motion. This allows for evaluating motion of arbitrary length.

In case of multiple human motion forecasting we propose a novel evaluation protocol that actually takes into account social interactions and not just independently evaluates the motion of individuals.

We introduce a novel social summarization function, which is order-invariant, and which allows us to work on arbitrary number of persons. We train a Diffusion model, called Social Diffusion, which utilizes diffusion inpainting during inference for motion forecasting. We show that this method can forecast social motion for 1 minute, much longer than state-of-the-art methods.

Last but not least we created a large-scale social human motion dataset which contains social interactions on an unprecedented scale. Our dataset consists of 3D human motion of up to 16 persons, represented as SMPL, 3D scene geometry as a static 3D scan as well as per-frame box annotations to cover movable objects, and per-person per-frame human action annotations. Together with our dataset we release an evaluation protocol to compare various state-of-the-art methods. We release the source code for all our works under the permissive MIT license.

9.2 Contributions and Discussion

The work presented in this thesis contributed towards three main directions, as presented below.

9.2.1 Long-term Human Motion Forecasting

The field of long-term human motion forecasting, forecasting more than 1 seconds of 3D human motion, remained relatively unexplored prior to our works. The high degree of stochasticity of human motion necessitates modeling the distribution of human motion to not suffer from common artifacts such as motion freeze up. We contribute two state-of-the-art generative motion prediction models, one for single-person motion forecasting and one for social human motion forecasting.

Single Human Motion Forecasting: In Chapter 6 we introduced Intention RNN [185] where we contributed an efficient single-human long-term motion forecasting approach. We found that using a generative model training framework, namely generative adversarial training, and modelling the distribution of human motion, allowed us to forecast motion for much longer time horizons than other approaches. However, we found that on complex motion datasets such as Human3.6M [85] with various diverse motions, such as walking, eating or discussion, this method alone produces motion only of the main motion mode after a few seconds, which is not a realistic continuation of the motion. Our core insight is that forecasting the human intention, represented as a discrete per-frame label, is crucial to prevent this mode collapse. This way we first forecast a discrete latent intention for a desired prediction length and then generate 3D human motion based on the forecast frames. This hierarchical forecasting strategy allows us to generate smooth transitions between different motion modes, as illustrated in Figure 6.1. We show that our method outperforms other state-of-the-art methods for long-term human motion forecasting on two human motion datasets and we further introduce a novel evaluation scheme which closer follows human perception. As a generative model our method is able to sample multiple possible future motions. In comparison to some other generative models we produce less diverse output. However, other methods sample from the entire distribution and thus produce unlikely motion. For example, a person walking might receive forecast motion of immediately leaning down. In contrast, our method produces contiguous walking motion.

Social Human Motion Forecasting: While some works [4, 5, 200] address the challenging task of multiple human motion forecasting we argue that those methods do not truly evaluate on social motion forecasting. For example, in Figure 9.1 we show the various degrees of interaction on the 3DPW [196] dataset, a commonly used dataset for multiple human motion forecasting. The dataset

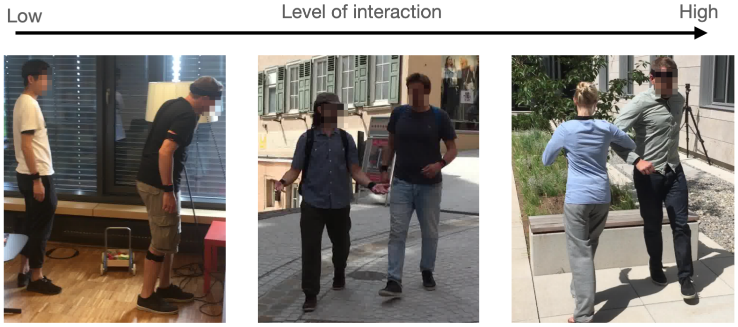


Figure 9.1: Various degrees of interactions in 3DPW [196].

encompasses various levels of social interactions, ranging from no interaction between actors to tightly interlocked motion. However, it is not clear how to evaluate the quality of social interactions in those datasets as models might produce motion of very high quality individually, but those motion sequences might not fit to each other. For example, two persons might be talking to each other, but not face each other, which is a highly unusual social interaction. Furthermore, human social interactions tend to last 10s of seconds while current multi-human motion forecasting methods only evaluate on 1 to 4 seconds of motion, which is too short for typical social interactions to play out. We address this by introducing a true social motion evaluation scheme based on the Haggling [99] dataset. This dataset includes well-defined triadic social interactions which allows us to train and evaluate not just the individual motion quality but also the quality of the generated social interaction. For this we introduce a novel evaluation metric which allows the evaluation of social interactions.

Equipped with this novel evaluation scheme we propose social diffusion in Chapter 7, a causal TCN with a novel social summarization function to pass information between various individuals. We train the model as unconditioned diffusion model to learn the distribution of social motion on the Haggling dataset. During inference we overwrite the historic part of the motion with the ground-truth input motion at each diffusion step, ensuring an inference-based conditioning on the past motion. For our summarizing function we compare two versions: multi-headed attention and simple averaging: we found that using averaging works best on the Haggling dataset. We argue that this is due to the strong explicit prior of averaging, giving equal importance to all actors. This works well for Haggling where all actors are linked in a strong triadic interaction. However, for more complex social interactions we believe that the attention mechanism works better, due to being able to asign importance based on context. Our experiments show that our method outperforms state-of-the-art multi-person forecasting methods on social forecasting.

9.2.2 Human Motion Generation Evaluation

Long-term human motion forecasting necessitates generative models. However, evaluating generative models is an open problem. We contribute two novel evaluation schemes, in Chapter 6 to evaluate the motion quality of arbitrary length of a single person, and in Chapter 7 we introduce a novel score to evaluate the quality of generated social interactions.

Single Human Motion Quality Score: To evaluate the quality of a single human motion of arbitrary length, relying solely on a single ground-truth sequence is inadequate due to the high degree of stochasticity in human motion. Instead, we introduce NDMS [185]. This approach considers the entire test set as potential ground-truth references. Specifically, we segment the generated motion into discrete units, each approximately one-third of a second long, referred to as *motion words*. For each motion word in the generated sequence, we identify the closest matching motion word in the test set. The quality of the generated motion is then assessed based on how closely these matched pairs resemble each other, using a motion similarity score. This evaluation metric is straightforward to implement, as it requires no additional training beyond the test set preparation and is designed to be easily shareable. It also provides robust results by considering both the structural alignment of poses and the dynamics of the motion through the similarity scoring. Our experiments demonstrate that this evaluation score correlates more strongly with human judgment than other existing metrics.

Social Motion Quality Score: In Chapter 7, we introduce the first attempt at systematically evaluating the quality of synthesized social interactions. To address this, we developed the Symbolic Social Cues Protocol (SSCP), which provides a novel framework for assessing the realism of social interactions in multi-person motion forecasting. Rather than relying solely on direct comparisons with ground-truth sequences, which may not capture the stochastic and intricate nature of human interactions, the SSCP assesses the quality of synthesized interactions through a probabilistic comparison of social state transitions.

The protocol operates by categorizing per-frame multi-person interactions into discrete classes, representing distinct per-frame social states such as one person talking while another listens. For each motion sequence, we derive a symbolic representation that summarizes the ongoing social interactions. These summaries are then used to construct a state transition matrix for each sequence in the test set, capturing the dynamics of social interactions over time.

To evaluate a given motion prediction model, we first predict future motion sequences based on initial frames and then apply the SSCP to these predictions. The evaluation metric is the Jensen-Shannon distance (JSD) between the state transition probabilities derived from the predicted and actual motion sequences. This distance measures the divergence between the predicted and real distributions of social states, providing a quantitative assessment of how well the model captures the complexity of human social behavior.

9.2.3 Social Human Motion Dataset

In Chapter 8 we introduce the first large-scale human motion dataset which attempts to cover all modes of realistic human motion: intention, social interactions, and scene geometry. Our dataset contains up to 16 persons at the same time, more than double of any other dataset. Furthermore, the recording scenes are much larger and in real kitchen environment, rather than in lab setups. For our dataset we build a long-term motion anticipation evaluation protocol and compare various state-of-the-art methods on it.

9.3 Future Work

9.3.1 Better Representations for Multiple Human Motion Forecasting

This thesis has explored various approaches to human motion forecasting, detailing common representations in Section 3.1 tailored for individual human poses. While these representations effectively handle motion for single individuals by using individual coordinate frames, thereby making the predictions translation and rotation invariant, challenges arise when extending these methods to multiple interacting humans. Current strategies, such as normalizing other individuals' poses relative to a reference person [72], prove inadequate in complex scenarios involving more than two individuals or in dynamically changing group settings. Such approaches can oversimplify interactions and fail to capture the subtleties of spatial relationships and inter-person dynamics critical in larger groups. The representation of multiple human poses in a unified yet distinct manner remains an unresolved issue in the field.

Future research needs to focus on developing novel representations that can scale with the complexity of interactions and the number of participants. One potential direction could be the exploration of dynamic reference systems that adjust based on the group's configuration and interaction dynamics. Alternatively, graph-based models that maintain individuality while capturing interconnections could offer a way to represent collective movements without losing the context of individual actions. These efforts would not only improve the accuracy of motion forecasting models but also enhance their applicability in real-world situations where human interactions are varied and complex. Finding effective ways to represent multiple humans in a shared space is crucial for advancing the field and developing applications that require nuanced understanding of human dynamics, such as socially aware robotics and advanced surveillance systems.

9.3.2 Conditioning on Scene Geometry

Scene geometry is an important mode in human motion which we have not explored in this work, yet it represents a crucial aspect that can significantly enhance the realism and accuracy of our models. Several approaches can be adopted to incorporate scene geometry: One method is to represent the scene as a collection of individual meshes, where each mesh corresponds to a distinct object, such as a coffee machine or a wall. Point cloud summarization techniques, such as PointNet [159] and Basis Point Sets [158], can then be utilized to extract meaningful features from each object. However, integrating these features into a comprehensive model that accounts for the interaction between multiple objects and multiple humans poses a significant challenge. A notable challenge in representing scene geometry is the large variability in the sizes of objects within a scene, from large walls to compact appliances like microwaves. This variability raises critical questions:

- How can we ensure that larger objects do not disproportionately influence the model's output, overshadowing smaller yet potentially more interaction-relevant objects?
- Which parts of large objects should the model focus on to effectively predict human interactions?

To effectively model the relationship between multiple humans and multiple objects, attention-based or graph-based approaches could be utilized to learn the object-human interactions.

Implicit scene representations could be another avenue for 3D geometry representation for object-human interactions, where the scene implicitly learns where certain activities happen, given human interactions with the scene. However, this would require a very large corpus of various human-scene interactions. Furthermore, it is not yet clear how this could be extended to novel scenes, as implicit scene representations are usually learned by overfitting a known 3D scene.

9.3.3 Hierarchical Action Planning and Motion Synthetization

In Chapter 6, we demonstrated the effectiveness of hierarchical approaches to motion forecasting for individual humans. This method shows promising potential for extension to multi-human scenarios, where it could enhance the realism and precision of motion forecasting over longer horizons and in more complex environments. However, applying hierarchical planning to multiple humans introduces significant challenges: In multi-human settings, each individual may require their own high-level planning function, reflecting their unique intentions and interactions within the group. These planning functions must not only operate independently but also interact dynamically with one another, adapting as the group dynamics evolve. The challenge here lies in coordinating these multiple planning functions so that they accurately reflect both individual behaviors and the interdependencies among the group members. As individuals interact within a scene, their plans may need to be continuously updated based on the actions and reactions of others. This requires a mechanism for real-time adjustment of plans that can accommodate new information from the environment and other individuals, ensuring that the forecasted motions remain coherent and contextually appropriate.

A possible approach to address these challenges could be the use of Large Language Models (LLMs). This way we could develop scripts that describe expected human actions within a scenario. These scripts would serve dual functions: Each participant could have a tailored script that outlines their expected actions, providing a clear plan that guides the motion prediction model. A comprehensive script could then be generated, given the individual scripts, to outline the overall scenario dynamics, specifying the general actions and interactions expected to occur among all participants. This global script can then be fed back to the individual script creation process to produce updated high-level scripts for each person, containing interactions with other individuals.

Bibliography

[1] CMU. Carnegie-Mellon Mocap Database. (Cited on pages xiv, 14, 15, 20, 45, 47, 61, 63, 64, 73, 77 and 81.)

[2] A survey on human-aware robot navigation. *Robotics and Autonomous Systems*, 145:103837, 2021. (Cited on pages 1, 72 and 88.)

[3] Aa, N.P. van der; Luo, X.; Giezeman, G.J.; Tan, R.T., and Veltkamp, R.C. Utrecht Multi-Person Motion (UMPM) benchmark: A multi-person dataset with synchronized video and motion capture data for evaluation of articulated human motion and interaction. In *Workshop on Human Interaction in Computer Vision*, 2011. (Cited on pages viii, xiii, 34, 37 and 38.)

[4] Adeli, Vida; Adeli, Ehsan; Reid, Ian; Niebles, Juan Carlos, and Rezatofighi, Hamid. Socially and contextually aware human motion and pose forecasting. *Robotics and Automation Letters*, 2020. (Cited on pages 77, 78, 81 and 100.)

[5] Adeli, Vida; Ehsanpour, Mahsa; Reid, Ian; Niebles, Juan Carlos; Savarese, Silvio; Adeli, Ehsan, and Rezatofighi, Hamid. Tripod: Human trajectory and pose dynamics forecasting in the wild. In *International Conference on Computer Vision*, 2021. (Cited on pages 2, 9, 11, 12, 72, 76 and 100.)

[6] Aksan, Emre; Kaufmann, Manuel, and Hilliges, Otmar. Structured prediction helps 3d human motion modelling. In *International Conference on Computer Vision*, 2019. (Cited on page 9.)

[7] Aksan, Emre; Kaufmann, Manuel; Cao, Peng, and Hilliges, Otmar. A spatio-temporal transformer for 3d human motion prediction. In *International Conference on 3D Vision*, 2021. (Cited on pages 1 and 72.)

[8] Alahi, Alexandre; Goel, Kratarth; Ramanathan, Vignesh; Robicquet, Alexandre; Fei-Fei, Li, and Savarese, Silvio. Social lstm: Human trajectory prediction in crowded spaces. In *Conference on Computer Vision and Pattern Recognition*, 2016. (Cited on pages 1, 72 and 88.)

[9] Aliakbarian, Sadegh; Saleh, Fatemeh Sadat; Petersson, Lars; Gould, Stephen, and Salzmann, Mathieu. Contextually plausible and diverse 3d human motion prediction. *arXiv preprint arXiv:1912.08521*, 2020. (Cited on pages xiv, 63 and 64.)

[10] Aliakbarian, Sadegh; Saleh, Fatemeh Sadat; Salzmann, Mathieu; Petersson, Lars, and Gould, Stephen. A stochastic conditioning scheme for diverse human motion prediction. In *Conference on Computer Vision and Pattern Recognition*, 2020. (Cited on pages xiv, 2, 10, 13, 29, 54, 55, 57, 62, 63, 64, 66, 69 and 76.)

[11] Aliakbarian, Sadegh; Saleh, Fatemeh; Petersson, Lars; Gould, Stephen, and Salzmann, Mathieu. Contextually plausible and diverse 3d human motion prediction. In *International Conference on Computer Vision*, 2021. (Cited on page 10.)

[12] Amin, Sikandar; Andriluka, Mykhaylo; Rohrbach, Marcus, and Schiele, Bernt. Multi-view pictorial structures for 3d human pose estimation. In *British Machine Vision Conference*, 2013. (Cited on page 19.)

[13] Andriluka, Mykhaylo; Roth, Stefan, and Schiele, Bernt. Discriminative appearance models for pictorial structures. *International Journal of Computer Vision*, 2012. (Cited on page 19.)

[14] Bagautdinov, Timur; Alahi, Alexandre; Fleuret, François; Fua, Pascal, and Savarese, Silvio. Social scene understanding: End-to-end multi-person action localization and collective activity recognition. In *Conference on Computer Vision and Pattern Recognition*, 2017. (Cited on pages 1, 72 and 88.)

[15] Bai, Shaojie; Kolter, J Zico, and Koltun, Vladlen. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018. (Cited on page 25.)

[16] Barsoum, Emad; Kender, John, and Liu, Zicheng. Hp-gan: Probabilistic 3d human motion prediction via gan. In *Conference on Computer Vision and Pattern Recognition workshops*, 2018. (Cited on pages 10, 63, 68 and 69.)

[17] Belagiannis, Vasileios; Amin, Sikandar; Andriluka, Mykhaylo; Schiele, Bernt; Navab, Nassir, and Ilic, Slobodan. 3d pictorial structures for multiple human pose estimation. In *Conference on Computer Vision and Pattern Recognition*, 2014. (Cited on pages vii, ix, xiii, 8, 32, 36, 37, 42, 48 and 51.)

[18] Belagiannis, Vasileios; Wang, Xinchao; Schiele, Bernt; Fua, Pascal; Ilic, Slobodan, and Navab, Nassir. Multiple human pose estimation with temporally consistent 3d pictorial structures. In *European Conference on Computer Vision*, 2014. (Cited on pages 8, 42 and 48.)

[19] Belagiannis, Vasileios; Amin, Sikandar; Andriluka, Mykhaylo; Schiele, Bernt; Navab, Nassir, and Ilic, Slobodan. 3d pictorial structures revisited: Multiple human pose estimation. *Transactions on Pattern Analysis and Machine Intelligence*, 2015. (Cited on pages 8 and 48.)

[20] Belagiannis, Vasileios; Amin, Sikandar; Andriluka, Mykhaylo; Schiele, Bernt; Navab, Nassir, and Ilic, Slobodan. 3d pictorial structures revisited: Multiple human pose estimation. *Transactions on Pattern Analysis and Machine Intelligence*, 2016. (Cited on pages 8, 32, 36, 37 and 42.)

[21] Bernardin, Keni; Elbs, Alexander, and Stiefelhagen, Rainer. Multiple object tracking performance metrics and evaluation in a smart room environment. In *International Workshop on Visual Surveillance*, 2006. (Cited on pages xiii, 38, 48 and 49.)

[22] Beyan, Cigdem; Murino, Vittorio; Venture, Gentiane, and Wykowska, Agnieszka. Computational approaches for human-human and human-robot social interactions, 2020. (Cited on page 88.)

[23] Bhatnagar, Bharat Lal; Xie, Xianghui; Petrov, Ilya A; Sminchisescu, Cristian; Theobalt, Christian, and Pons-Moll, Gerard. Behave: Dataset and method for tracking human object interactions. In *Conference on Computer Vision and Pattern Recognition*, 2022. (Cited on page 14.)

[24] Bhattacharyya, Apratim; Schiele, Bernt, and Fritz, Mario. Accurate and diverse sampling of sequences based on a "best of many" sample objective. In *Conference on Computer Vision and Pattern Recognition*, 2018. (Cited on pages 68 and 69.)

[25] Bregler, Christoph; Malik, Jitendra, and Pullen, Katherine. Twist based acquisition and tracking of animal and human kinematics. *International Journal of Computer Vision*, 2004. (Cited on page 20.)

[26] Bruckschen, Lilli; Amft, Sabrina; Tanke, Julian; Gall, Jürgen, and Bennewitz, Maren. Detection of generic human-object interactions in video streams. In *Social Robotics*, 2019. (Cited on page 88.)

[27] Burenius, Magnus; Sullivän, Josephine, and Carlsson, Stefan. 3D pictorial structures for multiple view articulated pose estimation. In *Conference on Computer Vision and Pattern Recognition*, 2013. (Cited on pages 32, 33, 36, 42 and 48.)

[28] Bütepage, Judith; Black, Michael J; Kragic, Danica, and Kjellstrom, Hedvig. Deep representation learning for human motion prediction and classification. In *Conference on Computer Vision and Pattern Recognition*, 2017. (Cited on pages 1, 9 and 72.)

[29] Bütepage, Judith; Kjellström, Hedvig, and Kragic, Danica. Anticipating many futures: Online human motion prediction and generation for human-robot interaction. In *International Conference on Robotics and Automation*, 2018. (Cited on page 10.)

[30] Cai, Yujun; Huang, Lin; Wang, Yiwei; Cham, Tat-Jen; Cai, Jianfei; Yuan, Junsong; Liu, Jun; Yang, Xu; Zhu, Yiheng; Shen, Xiaohui, and others, . Learning progressive joint propagation for human motion prediction. In *European Conference on Computer Vision*, 2020. (Cited on pages 1, 9 and 54.)

[31] Cao, Z.; Hidalgo Martinez, G.; Simon, T.; Wei, S., and Sheikh, Y. A. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *Transactions on Pattern Analysis and Machine Intelligence*, 2019. (Cited on page 43.)

[32] Cao, Zhe; Simon, Tomas; Wei, Shih-En, and Sheikh, Yaser. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. In *Conference on Computer Vision and Pattern Recognition*, 2017. (Cited on pages viii, 7, 34, 43, 45, 46, 47, 51 and 91.)

[33] Cao, Zhe; Gao, Hang; Mangalam, Karttikeya; Cai, Qi-Zhi; Vo, Minh, and Malik, Jitendra. Long-term human motion prediction with scene context. In *European Conference on Computer Vision*, 2020. (Cited on page 14.)

[34] Cervantes, Pablo; Sekikawa, Yusuke; Sato, Ikuro, and Shinoda, Koichi. Implicit neural representations for variable length human motion generation. In *European Conference on Computer Vision*, 2022. (Cited on page 12.)

[35] Charpentier, Caroline J and O'Doherty, John P. The application of computational models to social neuroscience: promises and pitfalls. *Social Neuroscience*, 2018. (Cited on page 88.)

[36] Chen, Ching-Hang and Ramanan, Deva. 3d human pose estimation= 2d pose estimation+ matching. In *Conference on Computer Vision and Pattern Recognition*, 2017. (Cited on page 8.)

[37] Chen, Di; Döring, Andreas; Zhang, Shanshan; Yang, Jian; Gall, Juergen, and Schiele, Bernt. Keypoint message passing for video-based person re-identification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022. (Cited on page 7.)

[38] Chen, Lujing; Liu, Rui; Yang, Xin; Zhou, Dongsheng; Zhang, Qiang, and Wei, Xiaopeng. Sttg-net: a spatio-temporal network for human motion prediction based on transformer and graph convolution network. *Visual Computing for Industry, Biomedicine, and Art*, 2022. (Cited on pages 1 and 72.)

[39] Chen, Yilun; Wang, Zhicheng; Peng, Yuxiang; Zhang, Zhiqiang; Yu, Gang, and Sun, Jian. Cascaded pyramid network for multi-person pose estimation. In *Conference on Computer Vision and Pattern Recognition*, 2018. (Cited on pages 7, 8 and 34.)

[40] Cho, Kyunghyun; Van Merriënboer, Bart; Gulcehre, Caglar; Bahdanau, Dzmitry; Bougares, Fethi; Schwenk, Holger, and Bengio, Yoshua. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014. (Cited on page 24.)

[41] Chu, Hau; Lee, Jia-Hong; Lee, Yao-Chih; Hsu, Ching-Hsien; Li, Jia-Da, and Chen, Chu-Song. Part-aware measurement for robust multi-view multi-human 3d pose estimation and tracking. In *Conference on Computer Vision and Pattern Recognition*, 2021. (Cited on page 8.)

[42] Ci, Hai; Wang, Chunyu; Ma, Xiaoxuan, and Wang, Yizhou. Optimizing network structure for 3d human pose estimation. In *International Conference on Computer Vision*, 2019. (Cited on page 8.)

[43] Dai, Angela; Nießner, Matthias; Zollöfer, Michael; Izadi, Shahram, and Theobalt, Christian. Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration. *Transactions on Graphics*, 2017. (Cited on page 14.)

[44] Dai, Jifeng; Li, Yi; He, Kaiming, and Sun, Jian. R-fcn: Object detection via region-based fully convolutional networks. *Advances in Neural Information Processing Systems*, 2016. (Cited on page 8.)

[45] Dang, Lingwei; Nie, Yongwei; Long, Chengjiang; Zhang, Qing, and Li, Guiqing. Diverse human motion prediction via gumbel-softmax sampling from an auxiliary space. In *International Conference on Multimedia*, 2022. (Cited on page 10.)

[46] Deutscher, Jonathan and Reid, Ian. Articulated body motion capture by stochastic search. *International Journal of Computer Vision*, 2005. (Cited on page 20.)

[47] Diller, Christian; Funkhouser, Thomas, and Dai, Angela. Forecasting characteristic 3d poses of human actions. In *Conference on Computer Vision and Pattern Recognition*, 2022. (Cited on pages 1 and 11.)

[48] Dilokthanakul, Nat; Mediano, Pedro AM; Garnelo, Marta; Lee, Matthew CH; Salimbeni, Hugh; Arulkumaran, Kai, and Shanahan, Murray. Deep unsupervised clustering with gaussian mixture variational autoencoders. *arXiv preprint arXiv:1611.02648*, 2016. (Cited on pages 68 and 69.)

[49] Doering, Andreas; Iqbal, Umar, and Gall, Juergen. JointFlow: Temporal Flow Fields for Multi Person Tracking. *British Machine Vision Conference*, 2018. (Cited on page 7.)

[50] Doering, Andreas; Chen, Di; Zhang, Shanshan; Schiele, Bernt, and Gall, Juergen. Posetrack21: A dataset for person search, multi-object tracking and multi-person pose tracking. In *Conference on Computer Vision and Pattern Recognition*, 2022. (Cited on page 7.)

[51] Dong, Junting; Jiang, Wen; Huang, Qixing; Bao, Hujun, and Zhou, Xiaowei. Fast and robust multi-person 3d pose estimation from multiple views. In *Conference on Computer Vision and Pattern Recognition*, 2019. (Cited on pages 8, 42, 48 and 51.)

[52] Elhayek, Ahmed; de Aguiar, Edilson; Jain, Arjun; Tompson, Jonathan; Pishchulin, Leonid; Andriluka, Micha; Bregler, Chris; Schiele, Bernt, and Theobalt, Christian. Efficient ConvNet-based marker-less motion capture in general scenes with a low number of cameras. In *Conference on Computer Vision and Pattern Recognition*, 2015. (Cited on page 32.)

[53] Endres, Dominik Maria and Schindelin, Johannes E. A new metric for probability distributions. *Transactions on Information theory*, 2003. (Cited on page 76.)

[54] Ershadi-Nasab, Sara; Noury, Erfan; Kasaei, Shohreh, and Sanaei, Esmaeil. Multiple human 3d pose estimation from multiview images. *Multimedia Tools and Applications*, 2018. (Cited on pages 8, 33, 36, 37, 42 and 48.)

[55] Fang, Hao-Shu; Xie, Shuqin; Tai, Yu-Wing, and Lu, Cewu. Rmpe: Regional multi-person pose estimation. In *International Conference on Computer Vision*, 2017. (Cited on page 8.)

[56] Felzenszwalb, Pedro F and Huttenlocher, Daniel P. Pictorial structures for object recognition. *International journal of computer vision*, 2005. (Cited on page 19.)

[57] Fieraru, Mihai; Khoreva, Anna; Pishchulin, Leonid, and Schiele, Bernt. Learning to refine human pose estimation. In *Conference on Computer Vision and Pattern Recognition Workshops*, 2018. (Cited on pages 7 and 34.)

[58] Fieraru, Mihai; Zanfir, Mihai; Oneata, Elisabeta; Popa, Alin-Ionut; Olaru, Vlad, and Sminchisescu, Cristian. Three-dimensional reconstruction of human interactions. In *Conference on Computer Vision and Pattern Recognition*, 2020. (Cited on pages 14 and 15.)

[59] Fischler, Martin A and Elschlager, Robert A. The representation and matching of pictorial structures. *Transactions on computers*, 1973. (Cited on page 19.)

[60] Fleuret, Francois; Berclaz, Jerome; Lengagne, Richard, and Fua, Pascal. Multicamera people tracking with a probabilistic occupancy map. *Transactions on Pattern Analysis and Machine Intelligence*, 2007. (Cited on pages ix, xiii, 37, 48 and 51.)

[61] Fragkiadaki, Katerina; Levine, Sergey; Felsen, Panna, and Malik, Jitendra. Recurrent network models for human dynamics. In *International Conference on Computer Vision*, 2015. (Cited on pages 1, 9 and 54.)

[62] Fuglede, Bent and Topsoe, Flemming. Jensen-Shannon divergence and Hilbert space embedding. In *International Symposium on Information Theory*, 2004. (Cited on page 76.)

[63] Gal, Yarin and Ghahramani, Zoubin. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, 2016. (Cited on pages 43 and 47.)

[64] Gall, Juergen; Rosenhahn, Bodo; Brox, Thomas, and Seidel, Hans-Peter. Optimization and filtering for human motion capture: A multi-layer framework. *International Journal of Computer Vision*, 2010. (Cited on pages 20 and 43.)

[65] Gopalakrishnan, Anand; Mali, Ankur; Kifer, Dan; Giles, Lee, and Ororbia, Alexander G. A neural temporal model for human motion prediction. In *Conference on Computer Vision and Pattern Recognition*, 2019. (Cited on pages xiv, 9, 13, 29, 54, 55, 61 and 76.)

[66] Gou, Jianping; Yu, Baosheng; Maybank, Stephen J, and Tao, Dacheng. Knowledge distillation: A survey. *International Journal of Computer Vision*, 2021. (Cited on page 9.)

[67] Grund, Christian; Tanke, Julian, and Gall, Juergen. Ellipose: Stereoscopic 3d human pose estimation by fitting ellipsoids. In *Winter Conference on Applications of Computer Vision*, pages 2871–2881, 2023. (Cited on pages 8 and 9.)

[68] Gui, Liang-Yan; Wang, Yu-Xiong; Liang, Xiaodan, and Moura, José MF. Adversarial geometry-aware human motion prediction. In *European Conference on Computer Vision*, 2018. (Cited on pages 1, 9, 54, 64 and 72.)

[69] Guo, Chuan; Zuo, Xinxin; Wang, Sen; Zou, Shihao; Sun, Qingyao; Deng, Annan; Gong, Minglun, and Cheng, Li. Action2motion: Conditioned generation of 3d human motions. In *International Conference on Multimedia*, 2020. (Cited on page 12.)

[70] Guo, Chuan; Zou, Shihao; Zuo, Xinxin; Wang, Sen; Ji, Wei; Li, Xingyu, and Cheng, Li. Generating diverse and natural 3d human motions from text. In *Conference on Computer Vision and Pattern Recognition*, 2022. (Cited on page 12.)

[71] Guo, Hengkai; Tang, Tang; Luo, Guozhong; Chen, Riwei; Lu, Yongchen, and Wen, Linfu. Multi-Domain Pose Network for Multi-Person Pose Estimation and Tracking. In *European Conference on Computer Vision*, 2018. (Cited on pages 7 and 34.)

[72] Guo, Wen; Bie, Xiaoyu; Alameda-Pineda, Xavier, and Moreno-Noguer, Francesc. Multi-person extreme motion prediction. In *Conference on Computer Vision and Pattern Recognition*, 2022. (Cited on pages 2, 11, 14, 72 and 103.)

[73] Guo, Wen; Du, Yuming; Shen, Xi; Lepetit, Vincent; Alameda-Pineda, Xavier, and Moreno-Noguer, Francesc. Back to mlp: A simple baseline for human motion prediction. In *Winter Conference on Applications of Computer Vision*, 2023. (Cited on pages 9, 10, 95 and 96.)

[74] Gurumurthy, Swaminathan; Kiran Sarvadevabhatla, Ravi, and Venkatesh Babu, R. Deligan: Generative adversarial networks for diverse and limited data. In *Conference on Computer Vision and Pattern Recognition*, 2017. (Cited on pages 68 and 69.)

[75] Hartley, Richard and Zisserman, Andrew. *Multiple view geometry in computer vision*. Cambridge university press, 2003. (Cited on pages vii, 21 and 22.)

[76] Hassan, Mohamed; Choutas, Vasileios; Tzionas, Dimitrios, and Black, Michael J. Resolving 3d human pose ambiguities with 3d scene constraints. In *International Conference on Computer Vision*, 2019. (Cited on pages 14 and 15.)

[77] Hassan, Mohamed; Ceylan, Duygu; Villegas, Ruben; Saito, Jun; Yang, Jimei; Zhou, Yi, and Black, Michael J. Stochastic scene-aware motion prediction. In *International Conference on Computer Vision*, 2021. (Cited on page 14.)

[78] Ho, Jonathan; Jain, Ajay, and Abbeel, Pieter. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 2020. (Cited on pages 72, 73, 74 and 75.)

[79] Holden, Daniel; Saito, Jun; Komura, Taku, and Joyce, Thomas. Learning motion manifolds with convolutional autoencoders. In *SIGGRAPH Asia technical briefs*. 2015. (Cited on pages 1 and 72.)

[80] Holden, Daniel; Saito, Jun, and Komura, Taku. A deep learning framework for character motion synthesis and editing. *Transactions on Graphics*, 2016. (Cited on pages 9 and 13.)

[81] Hossain, Mir Rayat Imtiaz and Little, James J. Exploiting temporal information for 3d human pose estimation. In *European Conference on Computer Vision*, 2018. (Cited on page 8.)

[82] Huang, Chun-Hao P.; Yi, Hongwei; Höschle, Markus; Safroshkin, Matvey; Alexiadis, Tsvetelina; Polikovsky, Senya; Scharstein, Daniel, and Black, Michael J. Capturing and inferring dense full-body human-scene contact. In *Computer Vision and Pattern Recognition*, 2022. (Cited on pages 14 and 15.)

[83] Huang, Siyuan; Wang, Zan; Li, Puhao; Jia, Baoxiong; Liu, Tengyu; Zhu, Yixin; Liang, Wei, and Zhu, Song-Chun. Diffusion-based generation, optimization, and planning in 3d scenes. In *Conference on Computer Vision and Pattern Recognition*, 2023. (Cited on page 12.)

[84] Huang, Xun; Liu, Ming-Yu; Belongie, Serge, and Kautz, Jan. Multimodal unsupervised image-to-image translation. In *European Conference on Computer Vision*, 2018. (Cited on page 13.)

[85] Ionescu, Catalin; Papava, Dragos; Olaru, Vlad, and Sminchisescu, Cristian. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *Transactions on Pattern Analysis and Machine Intelligence*, 2014. (Cited on pages ix, x, xiv, 14, 20, 45, 47, 55, 61, 62, 63, 64, 65, 66, 68, 69 and 100.)

[86] Iqbal, Umar; Milan, Anton, and Gall, Juergen. Posetrack: Joint multi-person pose estimation and tracking. In *Conference on Computer Vision and Pattern Recognition*, 2017. (Cited on page 8.)

[87] Iqbal, Umar; Milan, Anton, and Gall, Jürgen. PoseTrack: Joint Multi-Person Pose Estimation and Tracking. In *Conference on Computer Vision and Pattern Recognition*, 2017. (Cited on page 7.)

[88] Iqbal, Umar; Doering, Andreas; Yasin, Hashim; Krüger, Björn; Weber, Andreas, and Gall, Juergen. A dual-source approach for 3D human pose estimation from single images. *Computer Vision and Image Understanding*, 2018. (Cited on page 32.)

[89] Iqbal, Umar; Molchanov, Pavlo; Breuel Jürgen Gall, Thomas, and Kautz, Jan. Hand pose estimation via latent 2.5D heatmap regression. In *European Conference on Computer Vision*, 2018. (Cited on page 32.)

[90] Iqbal, Umar; Molchanov, Pavlo, and Kautz, Jan. Weakly-supervised 3d human pose learning via multi-view images in the wild. In *Conference on Computer Vision and Pattern Recognition*, 2020. (Cited on page 8.)

[91] Isik, Leyla; Mynick, Anna; Pantazis, Dimitrios, and Kanwisher, Nancy. The speed of human social interaction perception. *NeuroImage*, 2020. (Cited on page 88.)

[92] Iskakov, Karim; Burkov, Egor; Lempitsky, Victor, and Malkov, Yury. Learnable triangulation of human pose. In *International Conference on Computer Vision*, 2019. (Cited on page 8.)

[93] Jain, Ashesh; Zamir, Amir R; Savarese, Silvio, and Saxena, Ashutosh. Structural-rnn: Deep learning on spatio-temporal graphs. In *Conference on Computer Vision and Pattern Recognition*, 2016. (Cited on pages 1, 9, 13, 54 and 72.)

[94] Janner, Michael; Du, Yilun; Tenenbaum, Joshua B, and Levine, Sergey. Planning with diffusion for flexible behavior synthesis. *International Conference on Learning Representations*, 2022. (Cited on page 72.)

[95] Jiang, Nan; Liu, Tengyu; Cao, Zhexuan; Cui, Jieming; Chen, Yixin; Wang, He; Zhu, Yixin, and Huang, Siyuan. Chairs: Towards full-body articulated human-object interaction. *arXiv preprint arXiv:2212.10621*, 2022. (Cited on page 14.)

[96] Joo, Hanbyul; Liu, Hao; Tan, Lei; Gui, Lin; Nabbe, Bart; Matthews, Iain; Kanade, Takeo; Nobuhara, Shohei, and Sheikh, Yaser. Panoptic Studio: A Massively Multiview System for Social Motion Capture. In *International Conference on Computer Vision*, 2015. (Cited on pages vii, ix, x, xi, xiii, 15, 32, 37, 38, 48, 49, 51, 78, 81, 84, 85, 86 and 88.)

[97] Joo, Hanbyul; Simon, Tomas; Li, Xulong; Liu, Hao; Tan, Lei; Gui, Lin; Banerjee, Sean; Godisart, Timothy; Nabbe, Bart; Matthews, Iain, and others, . Panoptic studio: A massively multiview system for social interaction capture. *Transactions on Pattern Analysis and Machine Intelligence*, 2017. (Cited on pages 8 and 18.)

[98] Joo, Hanbyul; Simon, Tomas; Li, Xulong; Liu, Hao; Tan, Lei; Gui, Lin; Banerjee, Sean; Godisart, Timothy Scott; Nabbe, Bart; Matthews, Iain; Kanade, Takeo; Nobuhara, Shohei, and Sheikh, Yaser. Panoptic studio: A massively multiview system for social interaction capture. *Transactions on Pattern Analysis and Machine Intelligence*, 2017. (Cited on pages xi, 14, 15, 78, 85 and 86.)

[99] Joo, Hanbyul; Simon, Tomas; Cikara, Mina, and Sheikh, Yaser. Towards social artificial intelligence: Nonverbal social signal prediction in a triadic interaction. In *Conference on Computer Vision and Pattern Recognition*, 2019. (Cited on pages x, 14, 18, 73, 77, 78, 81, 83, 85 and 101.)

[100] Kantorovitch, Julia; Väre, Janne; Pehkonen, Vesa; Laikari, Arto, and Seppälä, Heikki. An assistive household robot–doing more than just cleaning. *Journal of Assistive Technologies*, 2014. (Cited on pages 1, 72 and 88.)

[101] Kazemi, Vahid; Burenius, Magnus; Azizpour, Hossein, and Sullivan, Josephine. Multi-view body part recognition with random forests. In *British Machine Vision Conference*, 2013. (Cited on pages xiii, 32, 33, 36 and 42.)

[102] Keitel, Anne; Daum, Moritz M, and others, . The use of intonation for turn anticipation in observed conversations without visual signals as source of information. *Frontiers in psychology*, 2015. (Cited on pages 1 and 72.)

[103] Khirodkar, Rawal; Chari, Visesh; Agrawal, Amit, and Tyagi, Ambrish. Multi-instance pose networks: Rethinking top-down pose estimation. In *International Conference on Computer Vision*, 2021. (Cited on page 8.)

[104] Kidziński, Łukasz; Yang, Bryan; Hicks, Jennifer L; Rajagopal, Apoorva; Delp, Scott L, and Schwartz, Michael H. Deep neural networks enable quantitative movement analysis using single-camera videos. *Nature communications*, 2020. (Cited on pages 1, 72 and 88.)

[105] Kocabas, Muhammed; Karagoz, Salih, and Akbas, Emre. MultiPoseNet: Fast multi-person pose estimation using pose residual network. In *European Conference on Computer Vision*, 2018. (Cited on pages 7 and 34.)

[106] Kosaraju, Vineet; Sadeghian, Amir; Martín-Martín, Roberto; Reid, Ian; Rezatofighi, Hamid, and Savarese, Silvio. Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. *Advances in Neural Information Processing Systems*, 2019. (Cited on pages 1 and 72.)

[107] Kostrikov, Ilya and Gall, Jürgen. Depth Sweep Regression Forests for Estimating 3D Human Pose from Images. In *British Machine Vision Conference*, 2014. (Cited on page 32.)

[108] Kratzer, Philipp; Bihlmaier, Simon; Midlagajni, Niteesh Balachandra; Prakash, Rohit; Toussaint, Marc, and Mainprice, Jim. Mogaze: A dataset of full-body motions that includes workspace geometry and eye-gaze. *Robotics and Automation Letters*, 2020. (Cited on page 14.)

[109] Kwon, Oh-Hun; Tanke, Julian, and Gall, Juergen. Recursive bayesian filtering for multiple human pose tracking from multiple cameras. In *Asian Conference on Computer Vision*, 2020. (Cited on pages 3, 41 and 87.)

[110] Kwon, Taesoo and Hodgins, Jessica. Momentum-mapped inverted pendulum models for controlling dynamic human motions. *Transactions on Graphics*, 2017. (Cited on page 9.)

[111] Levinson, Stephen C and Torreira, Francisco. Timing in turn-taking and its implications for processing models of language. *Frontiers in psychology*, 2015. (Cited on pages 2, 72, 76 and 79.)

[112] Li, Chao; Zhong, Qiaoyong; Xie, Di, and Pu, Shiliang. Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation. *arXiv preprint arXiv:1804.06055*, 2018. (Cited on pages 30 and 69.)

[113] Li, Chen; Zhang, Zhen; Sun Lee, Wee, and Hee Lee, Gim. Convolutional sequence to sequence model for human dynamics. In *Conference on Computer Vision and Pattern Recognition*, 2018. (Cited on pages 1, 9, 54, 64 and 72.)

[114] Li, Ruilong; Yang, Shan; Ross, David A, and Kanazawa, Angjoo. Ai choreographer: Music conditioned 3d dance generation with aist++. In *International Conference on Computer Vision*, 2021. (Cited on pages 12 and 14.)

[115] Libin Liu, Jessica Hodgins. Learning basketball dribbling skills using trajectory optimization and deep reinforcement learning. *Transactions on Graphics*, August 2018. (Cited on page 9.)

[116] Lin, Angela S; Wu, Lemeng; Corona, Rodolfo; Tai, Kevin; Huang, Qixing, and Mooney, Raymond J. Generating animated videos of human activities from natural language descriptions. *Learning*, 2018. (Cited on page 12.)

[117] Lin, Tsung-Yi; Maire, Michael; Belongie, Serge; Hays, James; Perona, Pietro; Ramanan, Deva; Dollár, Piotr, and Zitnick, C Lawrence. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014. (Cited on page 48.)

[118] Lin, Tsung-Yi; Dollár, Piotr; Girshick, Ross; He, Kaiming; Hariharan, Bharath, and Belongie, Serge. Feature pyramid networks for object detection. In *Conference on Computer Vision and Pattern Recognition*, 2017. (Cited on page 8.)

[119] Liu, Jun; Shahroudy, Amir; Perez, Mauricio; Wang, Gang; Duan, Ling-Yu, and Kot, Alex C. Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. *Transactions on Pattern Analysis and Machine Intelligence*, 2019. (Cited on pages 14 and 15.)

[120] Liu, Libin and Hodgins, Jessica. Learning to schedule control fragments for physics-based characters using deep q-learning. *Transactions on Graphics*, 2017. (Cited on page 9.)

[121] Liu, Yebin; Stoll, Carsten; Gall, Juergen; Seidel, Hans-Peter, and Theobalt, Christian. Markerless motion capture of interacting characters using multi-view image segmentation. In *Conference on Computer Vision and Pattern Recognition*, 2011. (Cited on pages 8, 32, 33 and 42.)

[122] Liu, Yebin; Gall, Juergen; Stoll, Carsten; Dai, Qionghai; Seidel, Hans-Peter, and Theobalt, Christian. Markerless motion capture of multiple characters using multiview image segmentation. *Transactions on Pattern Analysis and Machine Intelligence*, 2013. (Cited on pages 8 and 42.)

[123] Liu, Zhenguang; Wu, Shuang; Jin, Shuyuan; Liu, Qi; Lu, Shijian; Zimmermann, Roger, and Cheng, Li. Towards natural and accurate future motion prediction of humans and animals. In *Conference on Computer Vision and Pattern Recognition*, 2019. (Cited on page 9.)

[124] Loper, Matthew; Mahmood, Naureen; Romero, Javier; Pons-Moll, Gerard, and Black, Michael J. Smpl: A skinned multi-person linear model. *Transactions on Graphics*, 2015. (Cited on pages xiii, 13, 14, 20 and 88.)

[125] Lugmayr, Andreas; Danelljan, Martin; Romero, Andres; Yu, Fisher; Timofte, Radu, and Van Gool, Luc. Repaint: Inpainting using denoising diffusion probabilistic models. In *Conference on Computer Vision and Pattern Recognition*, 2022. (Cited on pages 72 and 75.)

[126] Ma, Haoyu; Chen, Liangjian; Kong, Deying; Wang, Zhe; Liu, Xingwei; Tang, Hao; Yan, Xiangyi; Xie, Yusheng; Lin, Shih-Yao, and Xie, Xiaohui. Transfusion: Cross-view fusion with transformer for 3d human pose estimation. *arXiv preprint arXiv:2110.09554*, 2021. (Cited on pages 8 and 9.)

[127] Ma, Tiezheng; Nie, Yongwei; Long, Chengjiang; Zhang, Qing, and Li, Guiqing. Progressively generating better initial guesses towards next stages for high-quality human motion prediction. In *Conference on Computer Vision and Pattern Recognition*, 2022. (Cited on pages 9, 95 and 96.)

[128] Mahmood, Naureen; Ghorbani, Nima; Troje, Nikolaus F.; Pons-Moll, Gerard, and Black, Michael J. AMASS: Archive of motion capture as surface shapes. In *International Conference on Computer Vision*, 2019. (Cited on pages 13, 14 and 91.)

[129] Mao, Wei; Liu, Miaomiao; Salzmann, Mathieu, and Li, Hongdong. Learning trajectory dependencies for human motion prediction. *International Conference on Compuver Vision*, 2019. (Cited on pages 1, 9, 61, 62, 63, 64, 69, 72, 77, 78 and 81.)

[130] Mao, Wei; Liu, Miaomiao, and Salzmann, Mathieu. History repeats itself: Human motion prediction via motion attention. In *European Conference on Computer Vision*, 2020. (Cited on pages 1, 9, 61, 62, 63, 72, 77, 78, 81, 95, 96 and 97.)

[131] Mao, Wei; Liu, Miaomiao, and Salzmann, Mathieu. Generating smooth pose sequences for diverse human motion prediction. In *International Conference on Computer Vision*, 2021. (Cited on page 10.)

[132] Martinez, Julieta; Black, Michael J, and Romero, Javier. On human motion prediction using recurrent neural networks. In *Conference on Computer Vision and Pattern Recognition*, 2017. (Cited on pages x, 1, 9, 13, 43, 44, 54, 61, 62, 63, 64, 66, 69, 72 and 95.)

[133] Martinez, Julieta; Hossain, Rayat; Romero, Javier, and Little, James J. A simple yet effective baseline for 3d human pose estimation. In *International Conference on Computer Vision*, 2017. (Cited on pages 8 and 32.)

[134] Medsker, Larry R and Jain, LC. Recurrent neural networks. *Design and Applications*, 2001. (Cited on page 24.)

[135] Mehta, Dushyant; Rhodin, Helge; Casas, Dan; Fua, Pascal; Sotnychenko, Oleksandr; Xu, Weipeng, and Theobalt, Christian. Monocular 3D human pose estimation in the wild using improved CNN supervision. In *International Conference on 3D Vision*, 2017. (Cited on page 32.)

[136] Mehta, Dushyant; Sotnychenko, Oleksandr; Mueller, Franziska; Xu, Weipeng; Sridhar, Srinath; Pons-Moll, Gerard, and Theobalt, Christian. Single-shot multi-person 3D pose estimation from monocular RGB. In *International Conference on 3D Vision*, 2018. (Cited on pages 14, 15, 32, 73, 77 and 81.)

[137] Mirza, Mehdi and Osindero, Simon. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. (Cited on page 58.)

[138] Moon, Gyeongsik; Chang, Ju Yong, and Lee, Kyoung Mu. V2v-posenet: Voxel-to-voxel prediction network for accurate 3d hand and human pose estimation from a single depth map. In *Conference on Computer Vision and Pattern Recognition*, 2018. (Cited on page 8.)

[139] Moreno-Noguer, Francesc. 3d human pose estimation from a single image via distance matrix regression. In *Conference on Computer Vision and Pattern Recognition*, 2017. (Cited on page 8.)

[140] Morris, Brendan Tran and Trivedi, Mohan Manubhai. A survey of vision-based trajectory learning and analysis for surveillance. *Transactions on Circuits and Systems for Video Technology*, 2008. (Cited on pages 1, 72 and 88.)

[141] Munkres, James. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, 1957. (Cited on pages 34 and 35.)

[142] Muñoz-Salinas, Rafael; Medina-Carnicer, R; Madrid-Cuevas, Francisco José, and Carmona-Poyato, A. Particle filtering with multiple and heterogeneous cameras. *Pattern Recognition*, 2010. (Cited on page 43.)

[143] Newell, Alejandro; Huang, Zhiao, and Deng, Jia. Associative embedding: End-to-end learning for joint detection and grouping. In *Advances in Neural Information Processing Systems*, 2017. (Cited on pages 7 and 34.)

[144] Nie, Yinyu; Dai, Angela; Han, Xiaoguang, and Nießner, Matthias. Pose2room: understanding 3d scenes from human activities. In *European Conference on Computer Vision*, pages 425–443. Springer, 2022. (Cited on page 12.)

[145] Oh, Sangmin; Hoogs, Anthony; Perera, Amitha; Cuntoor, Naresh; Chen, Chia-Chih; Lee, Jong Taek; Mukherjee, Saurajit; Aggarwal, JK; Lee, Hyungtae; Davis, Larry, and others, . A large-scale benchmark dataset for event recognition in surveillance video. In *Conference on Computer Vision and Pattern Recognition*, 2011. (Cited on pages 1, 72 and 88.)

[146] Oord, Aaron van den; Dieleman, Sander; Zen, Heiga; Simonyan, Karen; Vinyals, Oriol; Graves, Alex; Kalchbrenner, Nal; Senior, Andrew, and Kavukcuoglu, Koray. Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*, 2016. (Cited on page 25.)

[147] Paden, Brian; Čáp, Michal; Yong, Sze Zheng; Yershov, Dmitry, and Frazzoli, Emilio. A survey of motion planning and control techniques for self-driving urban vehicles. *Transactions on Intelligent Vehicles*, 2016. (Cited on pages 9 and 54.)

[148] Papandreou, George; Zhu, Tyler; Kanazawa, Nori; Toshev, Alexander; Tompson, Jonathan; Bregler, Chris, and Murphy, Kevin. Towards accurate multi-person pose estimation in the wild. In *Conference on Computer Vision and Pattern Recognition*, 2017. (Cited on page 8.)

[149] Pavlakos, Georgios; Zhou, Xiaowei; Derpanis, Konstantinos G, and Daniilidis, Kostas. Harvesting Multiple Views for Marker-less 3D Human Pose Annotations. In *Conference on Computer Vision and Pattern Recognition*, 2017. (Cited on pages 33 and 36.)

[150] Pavlakos, Georgios; Zhou, Xiaowei; Derpanis, Konstantinos G, and Daniilidis, Kostas. Coarse-to-fine volumetric prediction for single-image 3d human pose. In *Conference on Computer Vision and Pattern Recognition*, 2017. (Cited on page 32.)

[151] Pavlakos, Georgios; Choutas, Vasileios; Ghorbani, Nima; Bolkart, Timo; Osman, Ahmed A. A.; Tzionas, Dimitrios, and Black, Michael J. Expressive body capture: 3d hands, face, and body from a single image. In *Conference on Computer Vision and Pattern Recognition*, 2019. (Cited on pages 15 and 91.)

[152] Pavllo, Dario; Grangier, David, and Auli, Michael. Quaternet: A quaternion-based recurrent model for human motion. In *British Machine Vision Conference*, 2018. (Cited on pages 1, 9, 28 and 72.)

[153] Pavllo, Dario; Feichtenhofer, Christoph; Grangier, David, and Auli, Michael. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Conference on Computer Vision and Pattern Recognition*, 2019. (Cited on pages 1, 8 and 54.)

[154] Petrovich, Mathis; Black, Michael J, and Varol, Gül. Action-conditioned 3d human motion synthesis with transformer vae. In *International Conference on Computer Vision*, 2021. (Cited on page 12.)

[155] Petrovich, Mathis; Black, Michael J, and Varol, Gül. Temos: Generating diverse human motions from textual descriptions. In *European Conference on Computer Vision*, 2022. (Cited on page 12.)

[156] Piergiovanni, AJ; Angelova, Anelia; Toshev, Alexander, and Ryoo, Michael S. Adversarial generative grammars for human activity prediction. *European Conference on Computer Vision*, 2020. (Cited on pages 2, 10, 54, 57, 61, 62, 63, 64, 66 and 69.)

[157] Prince, Simon JD. *Computer vision: models, learning, and inference*. Cambridge University Press, 2012. (Cited on page 23.)

[158] Prokudin, Sergey; Lassner, Christoph, and Romero, Javier. Efficient learning on point clouds with basis point sets. In *International Conference on Computer Vision*, 2019. (Cited on pages 13 and 103.)

[159] Qi, Charles R; Su, Hao; Mo, Kaichun, and Guibas, Leonidas J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Conference on computer vision and pattern recognition*, 2017. (Cited on pages 13 and 103.)

[160] Radford, Alec; Kim, Jong Wook; Hallacy, Chris; Ramesh, Aditya; Goh, Gabriel; Agarwal, Sandhini; Sastry, Girish; Askell, Amanda; Mishkin, Pamela; Clark, Jack, and others, . Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021. (Cited on page 12.)

[161] Ramesh, Aditya; Dhariwal, Prafulla; Nichol, Alex; Chu, Casey, and Chen, Mark. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. (Cited on page 74.)

[162] Reddy, N Dinesh; Guigues, Laurent; Pishchulin, Leonid; Eledath, Jayan, and Narasimhan, Srinivasa G. Tessetrack: End-to-end learnable multi-person articulated 3d pose tracking. In *PConference on Computer Vision and Pattern Recognition*, 2021. (Cited on page 8.)

[163] Redmon, Joseph and Farhadi, Ali. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. (Cited on page 8.)

[164] Ren, Shaoqing; He, Kaiming; Girshick, Ross, and Sun, Jian. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 2015. (Cited on page 8.)

[165] Rogez, Gregory; Weinzaepfel, Philippe, and Schmid, Cordelia. Lcr-net++: Multi-person 2d and 3d pose detection in natural images. *Transactions on Pattern Analysis and Machine Intelligence*, 2019. (Cited on pages 7 and 34.)

[166] Rombach, Robin; Blattmann, Andreas; Lorenz, Dominik; Esser, Patrick, and Ommer, Björn. High-resolution image synthesis with latent diffusion models. In *Conference on Computer Vision and Pattern Recognition*, 2022. (Cited on page 72.)

[167] Ruiz, Alejandro Hernandez; Gall, Juergen, and Moreno-Noguer, Francesc. Human motion prediction via spatio-temporal inpainting. *International Conference on Computer Vision*, 2019. (Cited on page 76.)

[168] Sacks, Harvey; Schegloff, Emanuel A, and Jefferson, Gail. A simplest systematics for the organization of turn taking for conversation. In *Studies in the organization of conversational interaction*. 1978. (Cited on pages 2, 72 and 76.)

[169] Salimans, Tim; Goodfellow, Ian; Zaremba, Wojciech; Cheung, Vicki; Radford, Alec, and Chen, Xi. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016. (Cited on page 29.)

[170] Sampieri, Alessio; D'Amely, Guido; Avogaro, Andrea; Cunico, Federico; Skenderi, Geri; Setti, Francesco; Cristani, Marco, and Galasso, Fabio. Pose forecasting in industrial human-robot collaboration. *European Conference on Computer Vision*, 2022. (Cited on pages 9, 14, 95 and 96.)

[171] Särkkä, Simo. *Bayesian filtering and smoothing*. Cambridge University Press, 2013. (Cited on page 43.)

[172] Shahroudy, Amir; Liu, Jun; Ng, Tian-Tsong, and Wang, Gang. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *Conference on Computer Vision and Pattern Recognition*, 2016. (Cited on pages 15 and 18.)

[173] Shavit-Cohen, Keren and Zion Golumbic, Elana. The dynamics of attention shifts among concurrent speech in a naturalistic multi-speaker virtual environment. *Frontiers in Human Neuroscience*, 2019. (Cited on pages 76 and 79.)

[174] Shaw, Peter; Uszkoreit, Jakob, and Vaswani, Ashish. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*, 2018. (Cited on page 28.)

[175] Sigal, Leonid; Bhatia, Sidharth; Roth, Stefan; Black, Michael J, and Isard, Michael. Tracking loose-limbed people. In *Conference on Computer Vision and Pattern Recognition*, 2004. (Cited on page 20.)

[176] Sminchisescu, Cristian and Triggs, Bill. Estimating articulated human motion with covariance scaled sampling. *The International Journal of Robotics Research*, 2003. (Cited on page 20.)

[177] Sohn, Kihyuk; Lee, Honglak, and Yan, Xinchen. Learning structured output representation using deep conditional generative models. In *Advances in Neural Information Processing Systems*, 2015. (Cited on pages 10 and 72.)

[178] Song, Yang; Sohl-Dickstein, Jascha; Kingma, Diederik P; Kumar, Abhishek; Ermon, Stefano, and Poole, Ben. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. (Cited on page 72.)

[179] Starke, Sebastian; Zhang, He; Komura, Taku, and Saito, Jun. Neural state machine for character-scene interactions. *Transactions on Graphics*, 2019. (Cited on pages 9 and 12.)

[180] Starke, Sebastian; Zhao, Yiwei; Komura, Taku, and Zaman, Kazi. Local motion phases for learning multi-contact character movements. *Transactions on Graphics*, 2020. (Cited on page 9.)

[181] Sutskever, Ilya; Vinyals, Oriol, and Le, Quoc V. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, 2014. (Cited on page 9.)

[182] Szegedy, Christian; Vanhoucke, Vincent; Ioffe, Sergey; Shlens, Jon, and Wojna, Zbigniew. Rethinking the inception architecture for computer vision. In *Conference on Computer Vision and Pattern Recognition*, 2016. (Cited on page 29.)

[183] Tanke, Julian and Gall, Juergen. Iterative greedy matching for 3d human pose tracking from multiple views. In *German Conference on Pattern Recognition*, 2019. (Cited on pages 3, 31, 42, 45, 47, 48, 87 and 91.)

[184] Tanke, Julian; Kwon, Oh-Hun; Stotko, Patrick; Rosu, Radu Alexandru; Weinmann, Michael; Errami, Hassan; Behnke, Sven; Bennewitz, Maren; Klein, Reinhard; Weber, Andreas, and others, . Bonn activity maps: Dataset description. *arXiv preprint arXiv:1912.06354*, 2019. (Cited on pages 89 and 94.)

[185] Tanke, Julian; Zaveri, Chintan, and Gall, Juergen. Intention-based long-term human motion anticipation. In *2021 International Conference on 3D Vision*, 2021. (Cited on pages x, xiv, 1, 2, 4, 53, 76, 83, 86, 96, 97, 100 and 102.)

[186] Tanke, Julian; Zhang, Linguang; Zhao, Amy; Tang, Chengcheng; Cai, Yujun; Wang, Lezi; Wu, Po-Chen; Gall, Juergen, and Keskin, Cem. Social diffusion: Long-term multiple human motion anticipation. In *International Conference on Computer Vision*, 2023. (Cited on pages 2, 4, 71 and 96.)

[187] Tanke, Julian Alexander; Kwon, Oh-Hun; Mueller, Felix Benjamin; Doering, Andreas, and Gall, Juergen. Humans in kitchens: A dataset for multi-person human motion forecasting with scene context. In *Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. (Cited on pages 2, 3 and 87.)

[188] Tevet, Guy; Gordon, Brian; Hertz, Amir; Bermano, Amit H, and Cohen-Or, Daniel. Motionclip: Exposing human motion generation to clip space. In *European Conference on Computer Vision*, 2022. (Cited on page 12.)

[189] Tevet, Guy; Raab, Sigal; Gordon, Brian; Shafir, Yonatan; Cohen-Or, Daniel, and Bermano, Amit H. Human motion diffusion model. *International Conference on Learning Representations*, 2023. (Cited on pages 12, 72, 74, 75 and 91.)

[190] Tome, Denis; Russell, Christopher, and Agapito, Lourdes. Lifting from the deep: Convolutional 3d pose estimation from a single image. *Conference on Computer Vision and Pattern Recognition*, 2017. (Cited on page 32.)

[191] Tome, Denis; Toso, Matteo; Agapito, Lourdes, and Russell, Chris. Rethinking pose in 3d: Multi-stage refinement and recovery for markerless motion capture. In *Conference on 3D vision*, 2018. (Cited on page 8.)

[192] Tu, Hanyue; Wang, Chunyu, and Zeng, Wenjun. Voxelpose: Towards multi-camera 3d human pose estimation in wild environment. In *European Conference on Computer Vision*, 2020. (Cited on page 8.)

[193] Usman, Ben; Tagliasacchi, Andrea; Saenko, Kate, and Sud, Avneesh. Metapose: Fast 3d pose from multiple views without 3d supervision. In *Conference on Computer Vision and Pattern Recognition*, 2022. (Cited on page 8.)

[194] Vaswani, Ashish; Shazeer, Noam; Parmar, Niki; Uszkoreit, Jakob; Jones, Llion; Gomez, Aidan N; Kaiser, Łukasz, and Polosukhin, Illia. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. (Cited on pages vii, 9, 26, 27, 28 and 75.)

[195] von Marcard, Timo; Henschel, Roberto; Black, Michael; Rosenhahn, Bodo, and Pons-Moll, Gerard. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *European Conference on Computer Vision*, 2018. (Cited on pages 14, 15 and 73.)

[196] Von Marcard, Timo; Henschel, Roberto; Black, Michael J; Rosenhahn, Bodo, and Pons-Moll, Gerard. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *European conference on computer vision*, 2018. (Cited on pages xi, 73, 77, 81, 100 and 101.)

[197] Walker, Jacob; Marino, Kenneth; Gupta, Abhinav, and Hebert, Martial. The pose knows: Video forecasting by generating pose futures. In *International Conference on Computer Vision*, 2017. (Cited on pages 10, 63, 68 and 69.)

[198] Wandt, Bastian and Rosenhahn, Bodo. Repnet: Weakly supervised training of an adversarial reprojection network for 3d human pose estimation. In *Conference on Computer Vision and Pattern Recognition*, 2019. (Cited on page 8.)

[199] Wang, Borui; Adeli, Ehsan; Chiu, Hsu-kuang; Huang, De-An, and Niebles, Juan Carlos. Imitation learning for human pose prediction. In *International Conference on Computer Vision*, 2019. (Cited on pages 9, 54 and 64.)

[200] Wang, Jiashun; Xu, Huazhe; Narasimhan, Medhini, and Wang, Xiaolong. Multi-person 3d motion prediction with multi-range transformers. *Advances in Neural Information Processing Systems*, 2021. (Cited on pages x, xiv, 2, 9, 11, 72, 73, 76, 77, 78, 79, 81, 82, 83, 84, 95, 96, 97 and 100.)

[201] Wang, Jinbao; Tan, Shujie; Zhen, Xiantong; Xu, Shuo; Zheng, Feng; He, Zhenyu, and Shao, Ling. Deep 3d human pose estimation: A review. *Computer Vision and Image Understanding*, 2021. (Cited on page 8.)

[202] Wang, Jue; Huang, Shaoli; Wang, Xinchao, and Tao, Dacheng. Not all parts are created equal: 3d pose estimation by modeling bi-directional dependencies of body parts. In *International Conference on Computer Vision*, 2019. (Cited on page 8.)

[203] Wang, Zan; Chen, Yixin; Liu, Tengyu; Zhu, Yixin; Liang, Wei, and Huang, Siyuan. Humanise: Language-conditioned human motion generation in 3d scenes. In *Advances in Neural Information Processing Systems*, 2022. (Cited on page 14.)

[204] Xiao, Bin; Wu, Haiping, and Wei, Yichen. Simple baselines for human pose estimation and tracking. In *European Conference on Computer Vision*, 2018. (Cited on pages 7, 8 and 34.)

[205] Xu, Sirui; Wang, Yu-Xiong, and Gui, Liangyan. Stochastic multi-person 3d motion forecasting. In *International Conference on Learning Representations*, 2022. (Cited on page 12.)

[206] Yan, Xinchen; Rastogi, Akash; Villegas, Ruben; Sunkavalli, Kalyan; Shechtman, Eli; Hadap, Sunil; Yumer, Ersin, and Lee, Honglak. Mt-vae: Learning motion transformations to generate multimodal human dynamics. In *European Conference on Computer Vision*, 2018. (Cited on pages 10, 63, 68 and 69.)

[207] Yang, Sen; Quan, Zhibin; Nie, Mu, and Yang, Wankou. Transpose: Keypoint localization via transformer. In *International Conference on Computer Vision*, 2021. (Cited on page 8.)

[208] Yang, Wei; Ouyang, Wanli; Wang, Xiaolong; Ren, Jimmy; Li, Hongsheng, and Wang, Xiaogang. 3d human pose estimation in the wild by adversarial learning. In *Conference on Computer Vision and Pattern Recognition*, 2018. (Cited on page 8.)

[209] Yao, Angela; Gall, Juergen; Gool, Luc V, and Urtasun, Raquel. Learning probabilistic non-linear latent variable models for tracking complex activities. In *Advances in Neural Information Processing Systems*, 2011. (Cited on page 43.)

[210] Yasin, Hashim; Iqbal, Umar; Kruger, Bjorn; Weber, Andreas, and Gall, Juergen. A dual-source approach for 3d pose estimation from a single image. In *Conference on Computer Vision and Pattern Recognition*, 2016. (Cited on page 8.)

[211] Yuan, Ye and Kitani, Kris. Diverse trajectory forecasting with determinantal point processes. *International Conference on Learning Representations*, 2020. (Cited on pages 68 and 69.)

[212] Yuan, Ye and Kitani, Kris. Dlow: Diversifying latent flows for diverse human motion prediction. In *European Conference on Computer Vision*, 2020. (Cited on pages xiv, 2, 10, 13, 54, 55, 57, 62, 63, 64, 67, 68, 69 and 76.)

[213] Zhang, Jason Y; Felsen, Panna; Kanazawa, Angjoo, and Malik, Jitendra. Predicting 3d human dynamics from video. In *International Conference on Computer Vision*, 2019. (Cited on page 1.)

[214] Zhang, Mingyuan; Cai, Zhongang; Pan, Liang; Hong, Fangzhou; Guo, Xinying; Yang, Lei, and Liu, Ziwei. Motiondiffuse: Text-driven human motion generation with diffusion model. *arXiv preprint arXiv:2208.15001*, 2022. (Cited on pages 12 and 72.)

[215] Zhang, Si; Tong, Hanghang; Xu, Jiejun, and Maciejewski, Ross. Graph convolutional networks: a comprehensive review. *Computational Social Networks*, 2019. (Cited on page 9.)

[216] Zhang, Siwei; Zhang, Yan; Bogo, Federica; Pollefeys, Marc, and Tang, Siyu. Learning motion priors for 4d human body capture in 3d scenes. In *International Conference on Computer Vision*, 2021. (Cited on page 8.)

[217] Zhang, Siwei; Ma, Qianli; Zhang, Yan; Qian, Zhiyin; Kwon, Taein; Pollefeys, Marc; Bogo, Federica, and Tang, Siyu. Egobody: Human body shape and motion of interacting people from head-mounted devices. In *European Conference on Computer Vision*, 2022. (Cited on pages 14 and 15.)

[218] Zhang, Yuxiang; An, Liang; Yu, Tao; Li, Xiu; Li, Kun, and Liu, Yebin. 4d association graph for realtime multi-person motion capture using multiple video cameras. In *Conference on Computer Vision and Pattern Recognition*, 2020. (Cited on pages 8, 42, 45 and 48.)

[219] Zhao, Long; Peng, Xi; Tian, Yu; Kapadia, Mubbasir, and Metaxas, Dimitris N. Semantic graph convolutional networks for 3d human pose regression. In *Conference on Computer Vision and Pattern Recognition*, 2019. (Cited on page 8.)

[220] Zheng, Liang; Shen, Liyue; Tian, Lu; Wang, Shengjin; Wang, Jingdong, and Tian, Qi. Scalable person re-identification: A benchmark. In *International Conference on Computer Vision*, 2015. (Cited on pages 32 and 42.)

[221] Zheng, Yang; Yang, Yanchao; Mo, Kaichun; Li, Jiaman; Yu, Tao; Liu, Yebin; Liu, C Karen, and Guibas, Leonidas J. Gimo: Gaze-informed human motion prediction in context. In *European Conference on Computer Vision*, 2022. (Cited on page 14.)

[222] Zhou, Yi; Li, Zimo; Xiao, Shuangjiu; He, Chong; Huang, Zeng, and Li, Hao. Auto-conditioned recurrent networks for extended complex human motion synthesis. *International Conference on Learning Representations*, 2018. (Cited on page 9.)