UNIVERSITY OF RIJEKA
FACULTY OF INFORMATICS AND DIGITAL TECHNOLOGIES

Romeo Šajina

DEEP NEURAL NETWORK ARCHITECTURE AND MODELS FOR MULTI-PERSON POSE FORECASTING IN DYNAMIC SCENES

DOCTORAL THESIS

Rijeka, 2025

UNIVERSITY OF RIJEKA
FACULTY OF INFORMATICS AND DIGITAL TECHNOLOGIES

Romeo Šajina

DEEP NEURAL NETWORK ARCHITECTURE AND MODELS FOR MULTI-PERSON POSE FORECASTING IN DYNAMIC SCENES

DOCTORAL THESIS

Supervisor: prof. dr. sc. Marina Ivašić-Kos
Faculty of Informatics and Digital Technologies, University of Rijeka

Rijeka, 2025

SVEUČILIŠTE U RIJECI
FAKULTET INFORMATIKE I DIGITALNIH TEHNOLOGIJA

Romeo Šajina

ARHITEKTURA DUBOKIH NEURONSKIH MREŽA I MODELI ZA PREDVIĐANJE POZA VIŠE OSOBA U DINAMIČKIM SCENAMA

DOKTORSKI RAD

Mentor: prof. dr. sc. Marina Ivašić-Kos
Fakultet informatike i digitalnih tehnologija, Sveučilište u Rijeci

Rijeka, 2025

Supervisor: prof. dr. sc. Marina Ivašić-Kos

Faculty of Informatics and Digital Technologies, University of Rijeka

The doctoral thesis was defended on ______________ at the Faculty of Informatics and Digital Technologies at the University of Rijeka, in front of the examining committee consisted of:

1. ______________________________________
2. ______________________________________
3. ______________________________________

Acknowledgements

Abstract

The primary objective of this dissertation is to advance multi-person pose forecasting, a computer vision task with broad applications in human-computer interaction, autonomous systems, and sports analytics. Multi-person pose forecasting involves predicting the future poses of multiple individuals based on historical pose sequences, requiring models that can accurately capture both spatial configurations and temporal dynamics of human motion. A key contribution of this work is the introduction of a lightweight neural network architecture called MPFSIR, specifically designed for multi-person pose forecasting. MPFSIR includes a dedicated component for social interaction prediction, enabling it to model and anticipate interactions between individuals in a shared space. Despite its simplicity, MPFSIR achieves performance comparable to more complex state-of-the-art (SOTA) models, while using up to 30 times fewer parameters, making it highly suitable for real-time and resource-constrained applications. In addition, the dissertation proposes a novel hybrid architecture named GCN-Transformer, which combines Graph Convolutional Networks (GCN) and Transformers to jointly capture spatial dependencies among joints and temporal evolution of motion. The GCN-Transformer consistently outperforms competing methods across multiple datasets. Based on the MPJPE metric, it demonstrates an average 4.15% improvement over the closest SOTA model across four benchmark datasets. Unlike other models whose performance varies across datasets, GCN-Transformer shows consistent results, underscoring its robustness and generalization capability across different domains. This is further supported by the minimal variation in its improvements over the Zero Velocity baseline, with a standard deviation of only 1.69% in two-person scenes and just 0.1% in three-person scenes. Another major contribution is the development of a novel loss function that enhances training effectiveness. This loss integrates two key components: Velocity Loss (VL), which captures movement velocity consistency across frames, and Multi-Person Joint Distance Loss (MPJD), which models spatial coherence between individuals in the scene. These terms guide the model to produce motion that is both realistic and interaction-aware. Finally, the dissertation introduces a new evaluation metric named FJPTE that addresses the limitations of existing approaches. This metric jointly considers both local movement trajectories and the final global position, offering a more comprehensive and fine-grained assessment of forecasting accuracy than traditional metrics such as MPJPE or VIM. Extensive experimental validation on benchmark datasets, including SoMoF, ExPI, CMU-Mocap, and MuPoTS-3D, confirms the effectiveness of the proposed models, training strategies, and evaluation metric. The results highlight meaningful improvements in forecasting precision, efficiency, and generalization, establishing a strong foundation for practical deployment in complex, real-world multi-person scenarios.

Keywords: multi-person pose forecasting, neural network architecture, spatial and temporal features, graph convolutional network, Transformer model, pose estimation, pose tracking

Prošireni sažetak

Cilj ove disertacije je unaprijediti područje predviđanja poza više osoba, zadatak iz domene računalnog vida koji ima široku primjenu u interakciji čovjeka i računala, sportu, robotici i autonomnim sustavima. Predviđanje poza više osoba podrazumijeva automatsko modeliranje i predviđanje budućih položaja tijela više pojedinaca u sceni, temeljem prethodnih sekvenci njihovih kretnji. Rješavanje ovog zadatka zahtijeva pouzdano generiranje točnih vremenskih sekvenci poza iz videozapisa, što uključuje korištenje sofisticiranih metoda za estimaciju 2D i 3D poza te njihovog praćenja kroz vrijeme. U ovoj disertaciji predstavljeno je više znanstvenih doprinosa koji se bave ključnim izazovima u području predviđanja poza više osoba.

Prvi doprinos odnosi se na razvoj učinkovite arhitekture duboke neuronske mreže, nazvane MPFSIR, koja koristi prostorne i vremenske značajke sekvence poza kako bi precizno predviđala buduće poze. Osim toga, MPFSIR uključuje i komponentu za predviđanje socijalnih interakcija, što omogućuje modelu da prepozna i predvidi međudjelovanja između pojedinaca u sceni. Ovaj model uspješno održava ravnotežu između točnosti i složenosti, postižući rezultate usporedive s najnaprednijim (state-of-the-art) modelima, ali s do 30 puta manjim brojem parametara, čineći ga pogodnim za primjene u stvarnom vremenu i na uređajima s ograničenim resursima.

Drugi doprinos disertacije predstavlja nova arhitektura temeljena na kombinaciji graf konvolucijskih mreža (engl. *Graph Convolutional Network* – GCN) i Transformer modela. Ova hibridna arhitektura, nazvana GCN-Transformer, koristi snagu GCN-a za modeliranje odnosa između zglobova unutar svake poze, dok Transformer slojevi omogućuju učenje vremenskih odnosa između poza kroz cijelu sekvencu. GCN-Transformer nadmašuje druge modele, pri čemu na temelju MPJPE metrike pokazuje prosječno poboljšanje od 4,15% u odnosu na najbliži SOTA model na četiri evaluirana skupa podataka. Osim toga, za razliku od ostalih modela čije performanse znatno variraju ovisno o skupu podataka, GCN-Transformer pokazuje konzistentne rezultate, što potvrđuje njegovu robusnost u predviđanju poza više osoba i čini ga odličnom osnovom za primjenu u različitim domenama.

Uz arhitektonska rješenja, u disertaciji se predlaže i nova funkcija gubitka koja poboljšava treniranje modela. Funkcija uključuje dva specifična izraza: *Velocity Loss* (VL), koji se odnosi na brzinu kretanja zglobova, te *Multi-person Joint Distance Loss* (MPJD), koji modelira udaljenosti između zglobova različitih osoba. Ova funkcija gubitka omogućuje modelu da uči realističnije obrasce kretanja i bolje razumije prostorne odnose među pojedincima, što omogućuje generiranje realističnijih i vjerodostojnijih predviđanja. Učinkovitost predložene funkcije gubitka je potvrđeno empirijskim poboljšanjem preciznosti modela na standardnim evaluacijskim metrikama.

Dodatno, disertacija uvodi novu metriku za evaluaciju modela, nazvanu FJPTE, koja omogućuje detaljniju evaluaciju performansi prediktivnih modela. Za razliku od tradicionalnih metrika koje promatraju samo trenutni položaj zglobova ili završni okvir, FJPTE uključuje i evaluaciju cijele putanje kretanja te razlikuje lokalnu dinamiku pokreta od globalnog pomaka tijela. Time se omogućuje dublji uvid u stvarne prednosti i slabosti pojedinih modela.

Sve predložene metode i doprinosi su validirani kroz opsežna eksperimentalna ispitivanja na standardnim skupovima podataka, uključujući SoMoF Benchmark, ExPI, CMU-Mocap i MuPoTS-3D. Rezultati pokazuju značajna poboljšanja u točnosti predikcije i učinkovitosti modela, kao i sposobnost prilagodbe različitim vrstama scena i interakcija među osobama. Zaključno, ova disertacija donosi niz inovacija koje zajednički unapređuju područje predviđanja poza više osoba. Predložene arhitekture, specijalizirana funkcija gubitka i nova metrika evaluacije omogućuju izgradnju naprednijih modela koji preciznije, učinkovitije i robusnije predviđaju kretanje ljudi u složenim scenarijima. Time se postavljaju čvrsti temelji za daljnji napredak u ovom području računalnog vida.

Ključne riječi: predviđanje poza više osoba, arhitektura neuronske mreže, prostorne i vremenske značajke, graf konvolucijska mreža, Transformer model, predviđanje poza sa slike, praćenje poza

Contents

1. INTRODUCTION

In recent years, the field of computer vision has seen significant advancements driven by the growing computational power and sophisticated algorithms. One of the challenging areas within computer vision is pose forecasting, particularly for scenarios involving multiple individuals. Multi-person pose forecasting in dynamic scenes involves predicting the future poses of several people in a scene based on historical sequences of poses [52, 48, 36, 55, 35, 37, 13]. This task can be applied in numerous applications, including autonomous driving, human-computer interaction, surveillance, and sports analytics [11, 16, 28, 30].

Pose forecasting is a sequence prediction problem where the model predicts the future positions of body joints based on observed past movements. This requires a deep understanding of both spatial and temporal dynamics. Spatial dynamics refer to the relationships and dependencies between different body parts, while temporal dynamics involve the changes in these relationships over time. Capturing these intricate patterns is particularly challenging in multi-person scenarios due to the interactions between individuals and varying motion patterns. Early methods for pose forecasting often relied on Multi-Layered Perceptron networks (MLP) [6, 12, 43] or Recurrent Neural Networks (RNN) [33], which struggle to capture the complex nature of human motion. Recent approaches have leveraged advanced deep learning techniques, particularly Graph Convolutional Networks (GCNs) [28, 51, 37] and Transformer models [52, 48, 55, 35], to effectively model these complexities. However, these methods still face limitations in handling long-term dependencies and interactions between multiple individuals.

The increasing availability of video data from surveillance systems, mobile devices, and other sources has provided lots of data for training and evaluating pose forecasting models. However, transforming this raw data into accurate sequences of poses involves several steps, including pose estimation and tracking. Pose estimation detects the positions of key body joints in individual frames, while pose tracking ensures the consistency of these positions across frames, connecting them into coherent sequences. Despite advancements in these areas, the quality of pose data remains an essential factor influencing the performance of forecasting models.

1.1. Purpose of research

The main objective of this research is to advance the field of multi-person pose forecasting by tackling the significant challenges existing methods encounter. This dissertation focuses on the creation of sophisticated deep neural network architectures designed to capture both spatial and temporal dynamics in multi-person environments effectively. This study aims to develop more precise and reliable forecasting models using techniques like Multi-Layered Perceptions (MLP), Graph Convolutional Networks (GCNs), and Transformer models. Moreover, the research includes the design of a specialized loss function and a comprehensive evaluation metric to enhance the training and performance evaluation of these models. The ultimate goal is to provide practical solutions that can be utilized across various applications, improving the interaction between humans and machines in dynamic settings.

1.2. Research motivation

This research is motivated by the increasing need for intelligent systems capable of predicting human actions and responding effectively. In autonomous driving, predicting pedestrian movements can be used for enhancing safety and optimizing traffic flow. In human-computer interaction, predicting user behavior can lead to more intuitive and responsive interfaces. In sports analytics, anticipating athletes' movements offers valuable insights for improving performance and making strategic decisions. Despite these potential benefits, many existing pose forecasting methods still face challenges with forecasting error and efficiency, particularly in complex scenarios involving multiple individuals. This dissertation aims to address these shortcomings by utilizing the latest advancements in deep learning to develop more effective and practical pose forecasting models. The goal is to contribute meaningfully to the field of multi-person pose forecasting and its real-world applications.

1.3. Hypotheses and scientific contributions of research

As previously described, this dissertation's primary objective is to advance the field of multi-person pose forecasting by addressing its key challenges through innovative solutions.

Scientific hypotheses are:

- **H1:** A model utilizing spatial and temporal pose features can achieve equivalent multi-person pose forecasting performance as more complex SOTA models while using significantly fewer parameters.
- **H2:** Combining the architectures of graph convolutional networks and Transformers can create a model that has a lower error in multi-person pose forecasting compared to existing SOTA model architectures.
- **H3:** A loss function that includes movement velocity error and joint distance error between individuals contributes to the effective training of the model.

Realized scientific contributions are:

- A lightweight neural network architecture and model, named MPFSIR, for multi-person pose forecasting based on spatial and temporal features.
- A neural network architecture and model, named GCN-Transformer, for multi-person pose forecasting comprising a graph convolutional network and a Transformer.
- A loss function for effective training of pose forecasting models that includes movement velocities (Velocity Loss - VL) and joint distance between individuals (Multi-person Joint Distance - MPJD).
- An evaluation metric, named FJPTE, for pose forecasting that considers the movement trajectory and the final position.

The organization of this doctoral thesis is designed to systematically explore and validate the research hypotheses, detailing scientific contributions across multiple sections. The thesis began with an introduction that outlined the purpose of the research, the motivations driving the study, and a clear presentation of the research hypotheses and scientific contributions. Following the introduction, the elaboration section dives deep into the subject of multi-person pose forecasting. It discusses social interactions, problem formulation, metrics, and the datasets used in this study, setting the stage for subsequent detailed discussions on individual contributions. Each major contribution is then explored in its dedicated subsection. The thesis first presents a lightweight neural network architecture designed for efficient multi-person pose forecasting, followed by comprehensive experimental results across the SoMoF Benchmark, CMU-Mocap and MuPoTS-3D datasets. Next, it discusses a combined approach using Graph Convolutional Networks and Transformer to enhance multi-person pose forecasting, demonstrating results on the SoMoF Benchmark, ExPI, CMU-Mocap and MuPoTS-3D datasets. Further, the thesis discusses an innovative loss function tailored to effectively train pose forecasting models, including an ablation study to showcase its efficacy. This is complemented by a section on a new evaluation metric specifically developed for pose forecasting evaluation, alongside results highlighting its advantages over standard metrics. The practical application of these methodologies is then illustrated through a detailed pipeline for real-world pose forecasting, focusing on 2D and 3D pose estimation, tracking, and a thorough evaluation on a specific HBS dataset. The conclusion synthesizes the findings, reaffirming the validity of the scientific contributions and hypotheses, and outlines future research directions. This is followed by abstracts of articles from the doctoral research, which provide summaries of key papers published during the study.

2. ELABORATION

2.1. Multi-person pose forecasting

The task of multi-person pose forecasting has gained substantial attention in recent years, driven by the need for accurate and efficient models that can predict future human poses based on historical data. Early models in this field, such as the Zero Velocity model, established a simple yet effective benchmark for pose forecasting. This model predicts future poses by repeating the last observed pose for all future frames, effectively assuming no additional movement will occur. While conceptually straightforward and requiring no learning process, the Zero Velocity model has proven to be a surprisingly strong baseline, often matching or even outperforming more complex methods, especially in short-term forecasting scenarios where minimal movement occurs between frames.

Initial work predominantly concentrated on single-person pose forecasting. The LTD model, introduced by Mao et al. in [28], stands out for its use of Graph Convolutional Networks (GCNs). This model employs 12 GCN blocks with residual connections, along with additional graph convolutional layers at the start and end. These components work together to encode temporal information and refine features for pose prediction. Another noteworthy contribution is the Future Motion model by Wang et al. in [51], which also uses 12 GCN blocks but enhances performance through data augmentation, curriculum learning, and Online Hard Keypoints Mining (OHKM) loss. In contrast, Parsaeifard et al. in [33] proposed DViTA, which decomposes human movement into the global trajectory and local pose dynamics. This model utilizes a Long-Short Term Memory (LSTM) encoder-decoder network for trajectory forecasting and a Variational AutoEncoder (VAE) LSTM for local pose dynamics. Although innovative, the model's reliance on separate encoders for different dynamics introduces complexity that may impact scalability. The work of Chiu et al. in [11] and Mao, Liu, and Salzmann [26] explored hierarchical RNNs and GNNs for motion prediction, respectively. The latter utilizes graph attention networks to improve prediction across multiple entities. Guo et al. in [12] demonstrated that a simple MLP network with skip connections could outperform state-of-the-art models with significantly fewer parameters. Models like HR-STAN proposed by Medjaouri and Desai in [30] and GAGCN proposed by Zhong et al. in [56] have made strides by combining spatial and temporal components in their architectures. HR-STAN utilizes high-resolution spatio-temporal attention mechanisms to directly map a fixed-length pose history to a fixed-length pose forecasting sequence, eliminating the need for separate encoding and decoding steps and using dilated convolutions to increase the receptive field without compressing features. GAGCN, on the other hand, employs spatial and temporal gating networks to adaptively blend dependencies, deriving blending coefficients through a Kronecker product that captures the spatio-temporal dependencies for better motion representation.

Recent developments in multi-person pose forecasting have increasingly integrated social interactions and dependencies. Guo et al. in [13] introduced a model with two parallel pipelines for leader and follower individuals, incorporating attention mechanisms and GCN-based predictors. The Multi-Range Transformer (MRT) model proposed by Wang et al. in [52] utilizes a transformer-based architecture to capture both local individual motion and global social interactions. The MRT decoder forecasts future poses by attending to features from both local and global encoders, incorporating a motion discriminator to maintain natural motion characteristics. This approach enhances the model's robustness but may be computationally intensive. Similarly, the SoMoFormer model proposed by Vendrow et al. in [48] employs a standard Transformer Encoder to jointly predict pose trajectories for multiple individuals by encoding joint positions as Discrete Cosine Transform (DCT)-encoded padded trajectories. Peng et al. in [36] proposed model SoMoFormer^2, which captures both local and global pose dynamics using components like the displacement sub-sequence encoder (DSE), social interaction encoder (SIE), and Transformer predictor. The DSE employs multiple GCN units to extract features from sub-sequences, while the SIE simultaneously models individual motion and social interactions by capturing past displacements, temporal information, spatial relations, and social-aware attention. In contrast, JRTransformer, proposed by Xu et al. in [55], models future joint positions and relationships by analyzing temporal differentiation and explicit joint relations. TBIFormer model proposed by Peng, Mao, and Wu in [35] approaches the problem by decomposing poses into body parts and modeling their interactions separately. The Temporal Body Partition Module transforms sequences into body-part sequences, which are then processed by a Transformer Decoder for forecasting. Recent approaches like SocialTGCN, proposed by Peng et al. in [37], integrate a Pose Refine Module with GCN layers and a Social Temporal GCN encoder, which includes GCN and Temporal Convolutional Network (TCN) layers.

2.1.1. Social Interaction

Effective modeling of social interactions in multi-person settings has become an important aspect of pose forecasting. Models typically handle social interaction implicitly by processing all individuals in the scene simultaneously, requiring the model to implicitly learn social dynamics through interaction patterns. However, some models have introduced additional mechanisms or modules specifically designed to enhance the model's ability to capture and represent these social interaction dependencies more explicitly. Early models like SocialPool [1] aggregated information based on proximity but failed to account for the absence of social interaction despite spatial closeness. SocialPool uses pooling operations to combine neighboring individuals' features, which are then integrated with individual features for subsequent layers. While this approach simplifies interaction modeling, it does not fully capture the nuances of social dynamics. SoMoFormer [48] addressed some limitations of SocialPool by incorporating a grid positioning method to represent social connections. Each cell in the grid has a learnable positional embedding, and individuals are associated with specific cells based on their joint positions. This method improves spatial understanding but still lacks consideration for cases where individuals are close without social interaction. In contrast, Guo et al. in [13] introduced Cross-Interaction Attention (XIA), which models social interactions between dancers through a cross-interaction attention module. XIA refines pose information by updating keys and values using multi-head self-attention, enhancing the accuracy of motion forecasting through collaborative human motion prediction. Similarly, Peng et al. in [36] proposed a social interaction encoder (SIE) based on the Transformer model. SIE includes components for time encoding, spatial encoding, and social-aware motion attention. This approach effectively models social dynamics by integrating individual and social interactions, improving multi-person motion forecasting. Peng et al. in [37] addresses social interaction by constructing a Spatial Adjacency Matrix based on Euclidean distances between body root trajectories, which is fed to the model. Peng, Mao, and Wu in [35] proposed the Social Body Interaction Multi-Head Self-Attention module, which uses an attention mechanism to model both the dynamics of individual body parts and the interaction dependencies between body parts of multiple individuals. Overall, these advancements highlight the importance of incorporating social interactions into pose forecasting models. However, challenges remain in fully capturing the complexity of social dynamics and integrating them seamlessly into pose forecasting models, an aspect that this dissertation directly addresses through the development of models and loss functions specifically designed to model interpersonal interactions.

2.1.2. Problem formulation

In the multi-person pose forecasting task, the goal is to forecast the movements of multiple individuals within a given scene. Each individual is represented by a set of anatomical joints, such as elbows, knees, and shoulders. The objective is to forecast the trajectories of these joints over a future period, typically defined as T timesteps. To achieve this, the model is provided with a sequence of historical poses for each individual. These historical poses contain the positional data of each joint in three-dimensional Cartesian coordinates within a global coordinate system. For any individual $n = 1, \ldots, N$, each historical pose is described by a vector of J dimensions, where J represents the number of tracked joints. Therefore, the complete historical sequence for individual n is denoted as $X^n_{1:t}$, capturing the temporal progression of poses up to the current time. The length of the input sequence of poses is represented as t and determines the number of past poses utilized by the model for making predictions. The index n ranges from 1 to N, where N is the total number of individuals observed in the scene. The model's primary task is to generate future poses for each individual, denoted as $X^n_{t+1:T}$. Here, T indicates the number of timesteps into the future that the model needs to forecast. The problem formulation is visually represented in Figure 1.

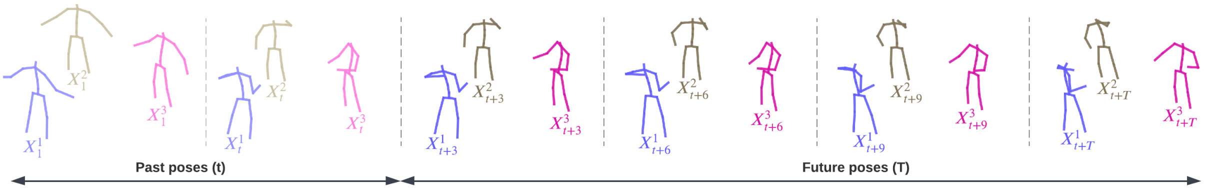


Figure 1: Problem formulation for predicting the future poses of multiple individuals. Each individual is represented by joints, and the task is to forecast their trajectories over T timesteps. The model uses a historical sequence of poses $X_{1:t}^n$ for each individual, containing joint positions in 3D coordinates, to predict future poses $X_{t+1:T}^n$ [45].

2.1.3. Metrics

To evaluate the performance of pose forecasting models, several metrics are commonly used. The Mean Per Joint Position Error (MPJPE) is a key metric for assessing the error of predicted poses. It calculates the average Euclidean distance between predicted and ground truth joint positions across all joints. A lower MPJPE value indicates a closer alignment of predicted poses to the actual positions. The MPJPE is defined as:

$$E_{\text{MPJPE}}(\hat{y}, y, \varphi) = \frac{1}{J_\varphi} \sum_{j=1}^{J_\varphi} \left\| P_{\hat{y},\varphi}^{(f)}(j) - P_{y,\varphi}^{(f)}(j) \right\|_2 \tag{1}$$

where f denotes the time step, and φ represents the skeleton. Here, $P_{\hat{y},\varphi}^{(f)}(j)$ and $P_{y,\varphi}^{(f)}(j)$ are the predicted and ground truth positions of joint j, respectively. J_φ is the total number of joints, and $\|\cdot\|_2$ denotes the Euclidean distance.

Another important metric is the Visibility-Ignored Metric (VIM), introduced by Adeli et al. in [2]. VIM measures the mean Euclidean distance between predicted and actual joint positions at the final pose T. This metric involves flattening the joint positions into a single vector with dimensionality $3J$, where J is the number of joints. The VIM score is defined as:

$$E_{\text{VIM}}(\hat{y}, y, \varphi) = \frac{1}{3J_\varphi} \sum_{j=1}^{3J_\varphi} \left\| P_{\hat{y},\varphi}^{(j)} - P_{y,\varphi}^{(j)} \right\|_2 \tag{2}$$

where $P_{y,\varphi}^{(i)}$ and $P_{\hat{y},\varphi}^{(i)}$ represent the flattened ground truth and predicted joint i positions, respectively. $\frac{1}{3J_\varphi} \sum_{j=1}^{3J_\varphi}$ computes the mean distance across all joints, while $\|\cdot\|_2$ denotes the Euclidean distance.

2.1.4. Datasets

To train and evaluate a multi-person pose forecasting model, several datasets capturing a variety of human motions and interactions can be employed. Standard methodology, as used in prior works such as SoMoFormer [48] and MRT [52], utilizes the 3D Poses in the Wild (3DPW) [50] and the Archive of Motion Capture As Surface Shapes (AMASS) [25] datasets. The 3DPW dataset includes over 60 video sequences of human motion in real-world settings, offering accurate 3D pose annotations in various natural scenes, such as people communicating, engaging in sports, or walking, recorded with a moving hand-held camera. However, 3DPW is used in the form of the SoMoF Benchmark [2], which inverts the standard train-test split, training the model on the test set and evaluating it on the train set. The AMASS dataset provides an extensive repository of motion capture data, comprising over 40 hours of motion and 11,000 single-person sequences presented as SMPL mesh models. For training, the CMU, BMLMovi, and BMLRub subsets of AMASS are typically used, as they encompass a broad range of motions. To generate multi-person training data from the single-person sequences, data syncretization is needed by combining sampled sequences, thereby creating multi-training scenarios.

Supporting datasets for additional evaluation include the CMU-Mocap [8] and MuPoTS-3D [31] datasets. The Carnegie Mellon University Motion Capture Database (CMU-Mocap) and the Multi-person Pose Estimation Test Set (MuPoTS-3D) contain three-person scenes with diverse motions, such as communication gestures and waving. However, these two datasets often feature simpler movements with minimal interactions. For a more challenging evaluation involving complex interactions and varied human motions, the Extreme Pose Interaction (ExPI) [13] dataset is preferred. ExPI includes dynamic sequences involving two couples of dances engaging in extreme movements and interactions, such as aerial maneuvers, resulting in a collection of 115 sequences and 60,000 annotated 3D body poses.

2.2. A lightweight neural network architecture for multi-person pose forecasting

The Multi-Person Pose Forecasting with Social Interaction Recognition (MPFSIR) [43] model is an advanced neural network architecture designed to enhance multi-person pose forecasting by integrating temporal, spatial, and social interaction information. The model architecture consists of several distinct modules: two temporal modules, a temporal context module, a spatial context module, and a social interaction auxiliary module. The architecture of the MPFSIR model features a series of fully connected layers integrated with skip connections to enhance information flow and gradient propagation. Each module in the model consists of fully connected layers interspersed with Parametric Rectified Linear Unit (PReLU) activation functions, layer normalization, and dropout for regularization. Skip connections are strategically implemented to link the initial and final layers within each module, ensuring that important information is preserved and facilitating more efficient learning. This design approach allows the model to effectively capture and utilize both temporal and spatial dependencies while also improving the robustness of the training process.

Initially, pose sequences are preprocessed by padding them with the last known pose to a uniform length and applying a Discrete Cosine Transform (DCT) to convert the human motion data into the frequency domain. This transformation helps capture motion dynamics more efficiently, as demonstrated in previous studies like SoMoFormer [48] and LTD [28]. After preprocessing, the model processes two separate pose sequences from the same scene, denoted as S_1 and S_2, through the first temporal module (T_1). This module extracts temporal features from each sequence. The output sequences from T_1 are then fed into the temporal context module ($TCTX$), which captures temporal dependencies between the two sequences, enabling the model to learn complex temporal interactions. Following the temporal context processing, the spatial context module ($SCTX$) captures spatial dependencies between the pose sequences. In parallel, the social interaction auxiliary module (SCINT) evaluates the nature of social interactions between the individuals represented by the sequences. SCINT classifies the relationships into categories: social interaction, no social interaction, or only one person in the scene, which enhances the model's ability to interpret social dynamics. After extracting features from these modules, the output sequences are fed into the temporal module T_2 to refine the prediction of the sequences along the temporal dimension. Finally, the output sequences from T_2 undergo an Inverse DCT (IDCT) to revert the data from the frequency domain back to Cartesian coordinates. In cases where a scene contains more than two individuals, the model processes the data by forming all possible pairs of individuals, performing two-person forecasting for each pair independently, and subsequently combining these pairwise predictions to produce the final multi-person forecast for the entire scene. The model architecture, along with sequence processing, is visually represented in Figure 2.

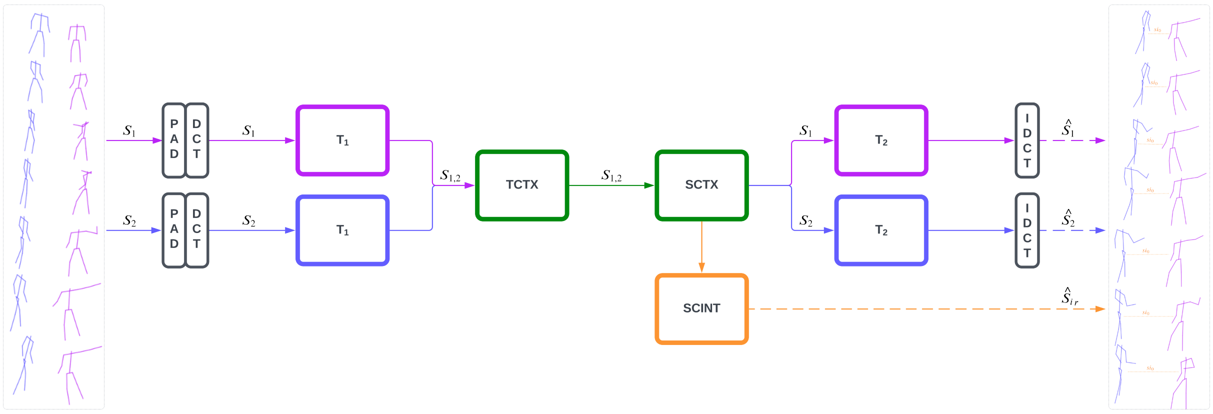


Figure 2: The figure depicts the MPFSIR model architecture. Input sequences S_1 and S_2 are padded and transformed using Discrete Cosine Transform (DCT) for frequency domain encoding. The model processes these sequences through several modules for pose forecasting and social interaction classification. Finally, the sequences are converted back to Cartesian coordinates using Inverse DCT (IDCT) to produce the predicted poses [43].

2.2.1. Training

Training the MPFSIR model involves several key steps to ensure robust and precise pose forecasting. The network learns to forecast future poses and classify the type of social interactions by minimizing a combined loss function that evaluates both pose error and type of interaction predictions. The loss is calculated as follows:

$$\mathcal{L} = \mathcal{L}_{rec} + \mathcal{L}_{scir} \times \gamma \tag{3}$$

$$\mathcal{L}_{rec} = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \tag{4}$$

$$\mathcal{L}_{scir} = -\frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{C} y_{i,j} \log(\hat{y}_{i,j}) \tag{5}$$

where $\mathcal{L}rec$ measures the reconstruction error for pose forecasting, and $\mathcal{L}scir$ represents the classification loss for predicting the type of social interaction. The hyperparameter γ controls the relative importance of the interaction classification loss in the total objective. In the L_{rec} loss term, N denotes the number of individuals in the scene, y_i represents the ground-truth pose sequence for the i-th individual, and $\hat{y}_i$ is the corresponding predicted pose sequence. This term minimizes the mean squared error (MSE) between predicted and ground-truth joint positions. In the L_{scir} loss term, n is the number of total interactions in the scene ($n = N - 1$), C is the number of interaction classes (e.g., interacting, not interacting, single person), $y_{i,j}$ denotes the ground-truth interaction type between individuals i for class j, and $\hat{y}_{i,j}$ is the corresponding predicted probability.

During training, the γ factor was set to 0.01 to ensure that pose forecasting remains the dominant learning objective. The model is trained for 500 epochs with a batch size of 256 using the Adam optimizer with an initial learning rate of 0.01, decayed by 0.1 at epochs 10, 200, and 400.

Additionally, data augmentation techniques are employed to aid the training process controlled by a probabilistic algorithm that determines whether and how to apply these techniques, ensuring diverse training examples and robust performance in real-world applications. Employed data augmentation techniques include sequence reversal, random scaling, random rotation, random positioning, and random person permutation.

2.2.2. Experimental results

The experimental evaluation of the MPFSIR model was conducted across three datasets: the SoMoF Benchmark, CMU-Mocap, and MuPoTS-3D. For the SoMoF Benchmark, results of the other models were sourced directly from the official website at `https://somof.stanford.edu`. In contrast, for CMU-Mocap and MuPoTS-3D, all presented models were re-evaluated using their respective implementations and training strategies to ensure a fair comparison, given the inherent randomness in the dataset creation process.

2.2.2.1. Results on SoMoF Benchmark

The SoMoF Benchmark [1, 2] is designed to evaluate the performance of multi-person pose forecasting methods. The benchmark involves predicting the next 14 frames (930 ms) using 16 frames (1070 ms) of input data. This input data includes joint positions for multiple people, and the results are reported as the mean VIM at multiple future time steps. Consistent with the methodology in [51] and [48], the 3DPW [50] and AMASS [25] datasets are used for training, providing both multi-person and single-person data. During training, only the 13 joints evaluated in SoMoF are used. Table 1 compares different methods on the SoMoF 3DPW test set, showing that the MPFSIR model consistently achieves competitive results with significantly fewer model parameters.

The results from the SoMoF Benchmark demonstrate the effectiveness of the MPFSIR model in multi-person pose forecasting. When comparing VIM values at various time steps, MPFSIR performs competitively against other state-of-the-art methods, such as SoMoFormer and Future Motion, while utilizing up to 30 times fewer parameters. Although SoMoFormer achieves the best overall performance with the lowest VIM values across all time steps, MPFSIR's results are close, particularly considering its significantly smaller number of parameters (0.15 million compared to SoMoFormer's 4.88 million). This highlights MPFSIR's efficiency and ability to maintain low predictive error with a

Table 1: Comparative performance analysis of different models on the SoMoF Benchmark test set using the VIM metric. Lower VIM values indicate a lower error in joint position predictions. The MPFSIR model demonstrates competitive performance relative to state-of-the-art methods, as shown by the official dataset page `https://somof.stanford.edu` results [43].

Method	3DPW Prediction in Time						Size
	100ms	240ms	500ms	640ms	900ms	Overall	#Param (M)
Mo-Att [27] + ST-GAT [16]	62.1	97.7	155.2	185.0	251.0	150.2	NA
SC-MPF [1]	46.3	73.9	130.2	160.8	208.4	123.9	15.65
Zero Velocity	29.4	53.6	94.5	112.7	143.1	86.7	0
TRiPOD [2]	30.3	51.8	85.1	104.8	146.3	83.7	NA
DViTA [33]	19.5	36.9	68.3	85.5	118.2	65.7	0.13
Future Motion [51]	9.5	22.9	50.9	66.2	97.4	49.4	2.56
SoMoFormer [48]	**9.1**	**21.3**	**47.5**	**61.6**	**91.9**	**46.3**	4.88
MPFSIR (our) [43]	11.5	25.5	54.7	70.6	101.5	52.76	**0.15**

lighter model. The compact architecture of MPFSIR makes it suitable for applications with limited computational resources, providing a practical alternative without a substantial loss in prediction precision. Notably, methods Mo-Att and SC-MPF perform so poorly that they fail to even outperform the Zero Velocity model, which only repeats the last known pose. Overall, the table indicates that MPFSIR strikes a favorable balance between model complexity and performance.

2.2.2.2. Results on CMU-Mocap and MuPoTS-3D

Additionally, MPFSIR is compared with the state-of-the-art models HRI [27], LTD [28], MRT [52], and SoMoFormer [48] on the CMU-Mocap and MuPoTS-3D datasets. Consistent with their respective protocols, models are trained using a synthesized dataset created by combining sampled motions from the CMU-Mocap database to generate three-person scenes. Evaluations are performed on both the CMU-Mocap and MuPoTS-3D datasets, while the models are trained only on the CMU-Mocap training set. For input, 15 frames (equivalent to 1000 ms) of historical data are provided, and the models are tasked with predicting the subsequent 45 frames (corresponding to 3000 ms). Performance is measured by reporting the Mean Per Joint Position Error (MPJPE) at 1, 2, and 3 seconds into the future. To ensure a fair comparison, the code and data provided by [52] are used to train and evaluate each method. The findings, presented in Table 2, show that the MPFSIR model consistently outperforms competing methods on both the CMU-Mocap and MuPoTS-3D datasets.

Table 2: Comparative analysis of model performance on the CMU-Mocap and MuPoTS-3D test sets using the MPJPE metric (in meters). Lower MPJPE values indicate a lower error in joint position predictions. The MPFSIR model outperforms other models in pose forecasting on both datasets [43].

Method	CMU-Mocap Test Set				MuPoTS-3D Test Set				Size
	1 sec	2 sec	3 sec	Overall	1 sec	2 sec	3 sec	Overall	#Param (M)
LTD [28]	4.03	7.06	9.91	7.00	1.75	2.98	4.10	2.94	2.61
MRT [52]	4.46	7.94	10.94	7.78	1.87	3.40	5.04	3.44	6.62
SoMoFormer [48]	4.50	8.15	11.27	7.79	1.69	3.02	4.15	2.95	4.88
MPFSIR (our) [43]	**3.94**	**7.04**	**9.87**	**6.95**	**1.67**	**2.87**	**3.93**	**2.82**	**0.24**

The MPFSIR model shows notable improvements over other methods in pose forecasting in multi-person scenes. On the CMU-Mocap test set, it achieves an overall MPJPE of 6.95, outperforming SoMoFormer, which achieves 7.79, and other models such as LTD and MRT. Similarly, on the MuPoTS-3D test set, the MPFSIR model achieves an overall MPJPE of 2.82, demonstrating better performance than SoMoFormer, which achieves 2.95, and LTD, which achieves 2.94, while the MRT achieves a significantly worse result of 3.44. The MPFSIR model's ability to accurately forecast future poses, especially in datasets involving multiple individuals, highlights its effectiveness in capturing complex interactions. Furthermore, it achieves these results with significantly fewer parameters (0.24M), making it more efficient than other state-of-the-art methods like SoMoFormer, which has 4.88M parameters. It should be noted that the number of parameters depends on the model's input and output sequence lengths, as well as the number of joints being predicted. For example, in configurations such as the SoMoF Benchmark, which uses shorter sequences with fewer joints, MPFSIR operates with as few as 0.15M parameters. This efficiency, combined with its superior performance, underscores the robustness and potential of the MPFSIR model in multi-person pose forecasting tasks.

2.3. Graph Convolutional Network and a Transformer for multi-person pose forecasting

The GCN-Transformer [45] for multi-person pose forecasting is a hybrid architecture that integrates Graph Convolutional Networks (GCNs) and Transformer modules to leverage their complementary strengths in short and long-term forecasting. The idea for this combination arose from experiments conducted in prior research [42], which demonstrated that Transformer-based models generally perform better in short-term pose forecasting, while GCN-based models perform better in long-term forecasting. By integrating both approaches, GCN-Transformer aims to achieve strong performance across both time horizons, combining the strengths of Transformers with the capabilities of GCNs.

The model is composed of two main modules: the Scene Module and the Spatio-Temporal Attention Forecasting Module. It begins by preprocessing input sequences of poses, padding them with the last known pose, and augmenting the data with temporal differentiation to create enriched sequences. Temporal differentiation refers to the process of computing the difference between joint positions across consecutive time steps to obtain motion velocity or first-order dynamics. Formally, for each person n, we compute $\Delta X_t^n = X_{t+1}^n - X_t^n$, and we concatenate this velocity signal with the original sequence along the joint feature's dimension. These sequences are then fed into the Scene Module, which encodes the poses into an embedding space using a Spatio-Temporal Fully-Connected module. This process prepares the input data for the next step, where the Spatial-GCN network, composed of 8 GCN blocks, captures the social features and interaction dependencies between individuals in a scene. The Spatial-GCN network processes the relationships between people in the scene, extracting spatial patterns while considering each individual's joint positions and movements relative to one another. This module incorporates techniques like batch normalization, dropout, and Tanh activation to optimize feature extraction, ensuring the preservation of the spatial structure in the data. Additionally, the model computes joint distance loss between individuals to maintain realistic spatial relationships between individuals.

Once the Scene Module extracts the social context, the enriched representation is passed to the Spatio-Temporal Attention Forecasting Module. This module is responsible for predicting future poses by processing the enriched sequence, scene context, and a positional query token that represents the position of each individual in the scene. The Spatio-Temporal Attention Forecasting Module splits the task between two submodules: the Spatio-Temporal Transformer Decoder and the Temporal-GCN. These two operate in parallel, simultaneously handling different aspects of the prediction task. The Spatio-Temporal Transformer Decoder processes the spatial and temporal dependencies in the data through two attention blocks. The first block focuses on spatial features, while the second block specializes in temporal patterns using Temporal Convolutional Network (TCN) layers to handle long-term temporal dependencies. In parallel, the Temporal-GCN submodule, which consists of 8 GCN blocks, refines the temporal dependencies of the sequences, enhancing the overall temporal representation of the data. After processing, the outputs from both the Spatio-Temporal Transformer Decoder and Temporal-GCN modules are concatenated and passed through another Spatio-Temporal Fully-Connected module to produce the final prediction of the future pose sequence.

Interestingly, the GCN-Transformer model avoids conventional preprocessing techniques, such as encoding pose data with the Discrete Cosine Transform (DCT) or predicting temporal differentiations that add to the last known pose. Instead, the model directly processes raw pose data, meaning the unaltered 3D joint Cartesian coordinates as directly obtained from pose estimation systems or datasets, allowing it to learn the inherent structures and dynamics of human motion without relying on artificially smoothed or transformed input. This decision was made to preserve the nuanced complexities of human movement that are often lost in conventional preprocessing techniques. The architecture of the GCN-Transformer model is illustrated in Figure 3.

Ultimately, the GCN-Transformer model represents a sophisticated blend of Transformer and GCN architectures. By leveraging the Transformer's strength in capturing temporal dependencies over short horizons and the GCN's effectiveness in modeling spatial dependencies and long-term temporal patterns, it offers a powerful tool for multi-person pose forecasting. Its ability to fuse features from multiple spatial, temporal, and social contexts ensures that it can make accurate and realistic predictions for future poses in complex social scenes.

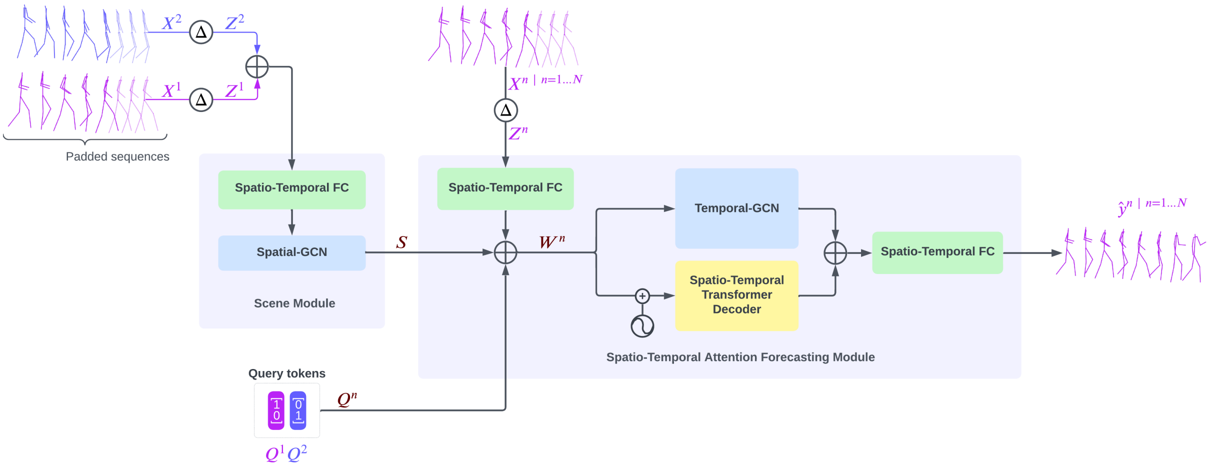


Figure 3: The figure illustrates the architecture of the GCN-Transformer model. The input sequences are first padded and enriched with temporal differentiation, forming sequences processed by the Scene Module, which extracts social features and dependencies. These features are combined with a positional query token and fed into the Spatio-Temporal Attention Forecasting Module, which uses a Spatio-Temporal Transformer Decoder and a Temporal-GCN to produce the final pose predictions for each individual in the scene [45].

2.3.1. Experimental results

The experimental evaluation of the GCN-Transformer was performed on four challenging multi-person pose forecasting datasets: the SoMoF Benchmark, the Extreme Pose Interaction (ExPI) dataset, the Carnegie Mellon University Motion Capture Database (CMU-Mocap), and the Multi-person Pose Estimation Test Set (MuPoTS-3D). In all cases, the model was trained over 512 epochs with a batch size of 256, using the Adam optimizer. The initial learning rate of 0.001 was reduced to 0.0001 after 256 epochs to ensure stable convergence. Importantly, all models presented in the experimental results, including GCN-Transformer, were retrained from scratch using their official implementations, with the exception of Future Motion, which we re-implemented based on the details provided in the original paper. All models were trained using the same formulation for the pose forecasting problem, predicting the next 14 frames based on 16 preceding frames. This differs from methods such as Future Motion, SoMoFormer, and JRTransformer, which divide the task into short-term and long-term forecasting, a strategy that typically boosts performance. Our experiments retained a unified problem formulation, allowing for a fair comparison of general model capability across forecasting horizons. In the up-coming section, we present experimental results on the CMU-Mocap and MuPoTS-3D datasets. The primary analysis will then shift to the SoMoF Benchmark and the Extreme Pose Interaction (ExPI) dataset, both of which feature two-person interactions and pose greater challenges due to their complex and realistic multi-person motion dynamics.

2.3.1.1. Results on CMU-Mocap and MuPoTS-3D

We evaluated the GCN-Transformer model against several state-of-the-art multi-person pose forecasting methods using the CMU-Mocap and MuPoTS-3D datasets, both designed to test models on three-person interaction scenarios. All models were trained using the same synthetic data based on the CMU-Mocap setup to ensure fair comparison, and performance was measured using the MPJPE metric at various time intervals.

The experimental results, presented in Table 3, show that the GCN-Transformer consistently performs best across both datasets and all forecast horizons. It outperforms all baseline methods in the short term and maintains this superiority as the prediction interval extends, demonstrating a robust capacity to model long-term motion trajectories. The model's generalization across two different datasets further underscores its adaptability to varied motion patterns and scene complexities.

Among the baseline models, MPFSIR, JRTransformer, and LTD show relatively competitive performance, although they remain behind GCN-Transformer in all metrics. LTD, in particular, performs well despite its original design for single-person forecasting tasks. Conversely, models such as MRT, SoMoFormer, and Future Motion exhibit significantly higher error rates, especially at longer forecast intervals. This suggests that while these models may handle short-term dependencies adequately, they struggle to maintain accuracy over extended predictions, likely due to limited temporal modeling capacity or weaker handling of movement dynamics dependencies.

Notably, we also observed a shift in performance rankings between the CMU-Mocap and MuPoTS-3D datasets. This change highlights the sensitivity of many models to dataset characteristics and reveals potential limitations in their generalization capabilities. In contrast, GCN-Transformer maintained top-tier performance on both datasets, indicating a stronger ability to generalize across data domains with varying levels of com-

Table 3: Comparison of model performance on the CMU-Mocap and MuPoTS-3D test sets, both of which include three-person scenes. The evaluation uses the MPJPE metric (measured in meters), where lower scores indicate better performance. The GCN-Transformer model demonstrates superior performance, outperforming all other evaluated methods across both datasets [45].

Method	CMU-Mocap Test Set				MuPoTS-3D Test Set				Average Overall
	1 s	2 s	3 s	Overall	1 s	2 s	3 s	Overall	
Zero Velocity	5.55	9.23	12.30	9.03	2.05	3.43	4.57	3.35	6.29
MRT [52]	4.46	7.94	10.94	7.78	1.87	3.40	5.04	3.44	5.61
SoMoFormer [48]	4.50	8.15	11.27	7.79	1.69	3.02	4.15	2.95	5.37
Future Motion [51]	4.08	7.24	10.21	7.18	1.98	3.40	4.57	3.31	5.25
JRTransformer [55]	4.08	7.47	10.47	7.34	1.61	2.90	4.06	2.86	5.16
LTD [28]	4.03	7.06	9.91	7.00	1.75	2.98	4.10	2.94	4.97
MPFSIR (our) [43]	3.94	7.04	9.87	6.95	1.67	2.87	3.93	2.82	4.89
GCN-Transformer (our) [45]	**3.53**	**6.58**	**9.25**	**6.46**	**1.39**	**2.41**	**3.39**	**2.40**	**4.43**

plexity and realism.

These findings validate the design of the GCN-Transformer model and confirm the effectiveness of combining graph-based spatial reasoning with Transformer-based temporal modeling. Its consistent performance across datasets and prediction ranges reflects a robust understanding of human motion dynamics in multi-person contexts. Building on this foundation, the next sections focus on evaluating the model under even more socially complex conditions using the SoMoF Benchmark and the ExPI dataset, which feature highly dynamic and interactive multi-person scenes.

2.3.1.2. Results on SoMoF Benchmark

The SoMoF Benchmark, derived from the 3DPW dataset, is a standard test for multi-person pose forecasting. The GCN-Transformer consistently achieved state-of-the-art performance across multiple evaluation metrics, including the Visibility-Ignored Metric (VIM) and the Mean Per Joint Position Error (MPJPE). As shown in Table 4, GCN-Transformer outperformed other state-of-the-art models, particularly excelling in short to mid-term forecasting (100ms to 640ms). For instance, GCN-Transformer achieved an overall VIM score of 48.02 and an MPJPE score of 61.90, surpassing strong competitors such as SoMoFormer, which obtained an overall VIM of 48.19 and MPJPE of 62.62. Additionally, the performance of the GCN-Transformer is significantly improved when incorporating the validation set from the 3DPW into the training set, resulting in an enhanced overall score of 46.21 on VIM and 59.48 on MPJPE.

Table 4: Comparison of performance on the SoMoF Benchmark test set, evaluated using the VIM and MPJPE metrics, where lower scores indicate better performance. The GCN-Transformer model achieves state-of-the-art results. The model noted with an asterisk (*) was trained with the validation dataset included and currently ranks first on the official SoMoF Benchmark leaderboard at `https://somof.stanford.edu` [45].

Method	VIM						MPJPE					
	100ms	240ms	500ms	640ms	900ms	Overall	100ms	240ms	500ms	640ms	900ms	Overall
Zero Velocity	29.35	53.56	94.52	112.68	143.10	86.65	55.28	87.98	146.10	173.30	223.16	137.16
DViTA [33]	17.40	35.62	72.06	90.87	127.27	68.65	32.09	54.48	100.03	124.07	173.01	96.74
LTD [28]	18.07	34.88	68.16	85.07	116.83	64.60	33.57	55.21	97.57	119.58	163.69	93.92
TBIformer [35]	17.62	34.67	67.50	84.01	116.38	64.03	32.26	53.65	95.61	117.22	160.99	91.94
MRT [52]	15.31	31.23	63.16	79.61	111.86	60.24	27.97	47.64	87.87	108.93	151.96	84.88
SocialTGCN [37]	12.84	27.41	58.12	74.59	107.19	56.03	23.10	40.24	76.91	96.89	139.01	75.23
JRTransformer [55]	11.17	25.73	56.50	73.19	106.87	54.69	18.44	35.38	72.26	92.42	135.12	70.73
MPFSIR (our) [43]	11.57	25.37	54.04	69.65	101.13	52.35	20.31	35.69	69.58	88.36	128.37	68.46
Future Motion [51]	10.76	24.52	54.14	69.58	100.81	51.96	18.66	34.38	69.76	88.91	129.18	68.18
SoMoFormer [48]	10.45	23.10	49.76	64.30	**93.34**	48.19	17.63	32.42	63.86	81.20	**117.97**	62.62
GCN-Transformer (our) [45]	**10.14**	**22.54**	**48.81**	**63.67**	94.94	**48.02**	**17.11**	**31.48**	**62.62**	**80.14**	118.14	**61.90**
GCN-Transformer* (our) [45]	**9.82**	**21.80**	**46.61**	**60.88**	**91.95**	**46.21**	**16.41**	**30.36**	**60.31**	**76.94**	**113.36**	**59.48**

This consistent superiority is notable across various time steps, where GCN-Transformer showed minimal degradation in performance as the forecasting horizon increased, underscoring its robustness. While models such as JRTransformer and MPFSIR exhibited reasonable short-term forecasting performance, their ability to predict accurately deteriorated significantly over long-term horizons (e.g., 900ms). In contrast, GCN-Transformer maintained competitive performance throughout both short and long-term intervals, with the SoMoFormer coming close only in the longest time frames (e.g., 900ms). JRTransformer, although a strong competitor, failed to match the performance of GCN-Transformer across most metrics, particularly struggling with long-term predictions where it exhibited significantly higher errors.

GCN-Transformer showcased its ability to generate realistic and coherent pose predictions, as illustrated in Figure 4. Unlike JRTransformer and SoMoFormer, which often generated invalid poses and unrealistic movements, GCN-Transformer's predictions adhered closely to the ground truth, reflecting a better understanding of complex human interactions in motion.

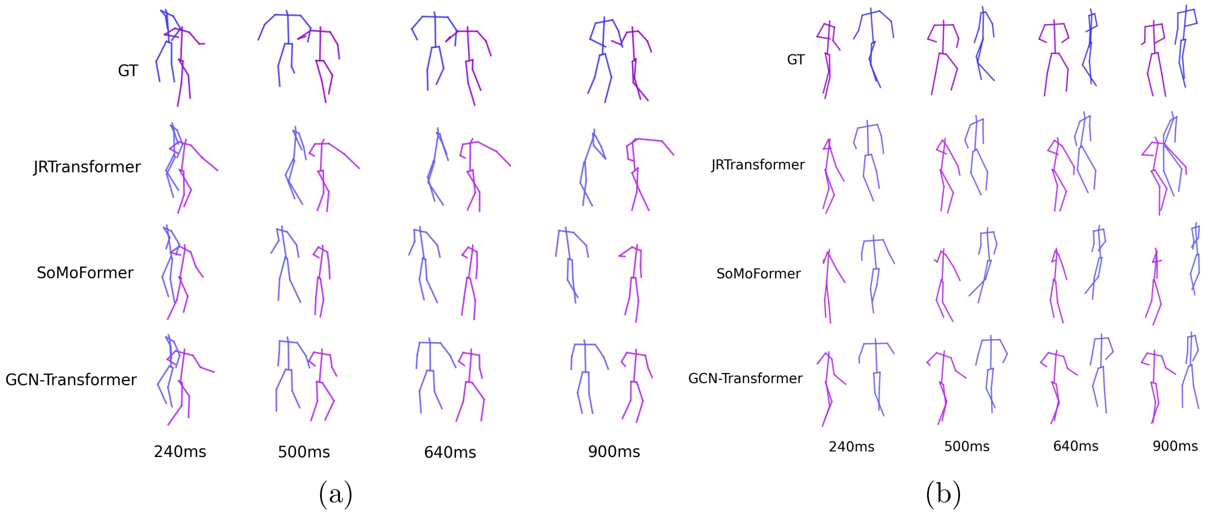


Figure 4: The figure shows predicted poses on two example sequence from the SoMoF Benchmark test set, comparing the top-performing models: JRTransformer, SoMoFormer, and GCN-Transformer, alongside the ground truth (GT). Sequence (**a**) shows two people rotating around each other, while sequence (**b**) shows two people meeting and then walking together in the same direction. The comparison highlights that while JRTransformer and SoMoFormer face challenges in producing valid poses, the GCN-Transformer effectively generates both accurate poses and realistic movements [45].

2.3.1.3. Results on ExPI Dataset

The ExPI dataset presents an even greater challenge due to its focus on extreme body poses during dance and aerial movements. On this dataset, GCN-Transformer once again demonstrated its adaptability and generalization capability, outperforming the competition across all metrics. As summarized in Table 5, GCN-Transformer achieved an overall VIM score of 40.64 and MPJPE of 53.45, significantly improving on the next best-performing models, JRTransformer (VIM of 42.06 and MPJPE of 54.87) and MPFSIR (VIM of 48.49 and MPJPE of 61.54).

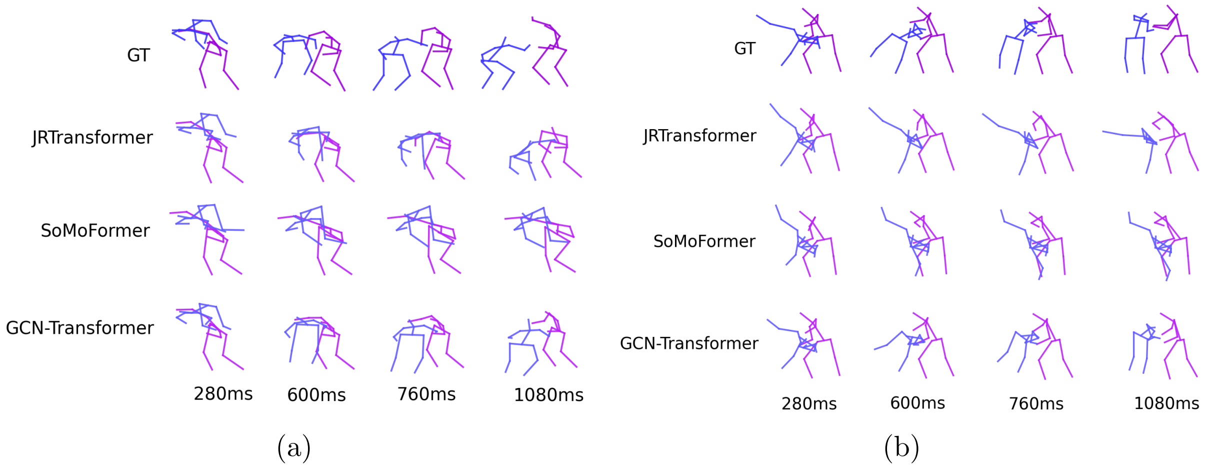
Table 5: Performance comparison on the ExPI test set based on the VIM and MPJPE metrics, where lower scores reflect better performance. The GCN-Transformer model achieves state-of-the-art results across both metrics [45].

Method	VIM						MPJPE					
	120ms	280ms	600ms	760ms	1080ms	Overall	120ms	280ms	600ms	760ms	1080ms	Overall
Zero Velocity	25.61	48.66	84.39	97.41	118.10	74.84	46.16	74.66	124.32	145.22	181.33	114.34
DViTA [33]	15.44	35.27	74.43	91.44	119.51	67.22	28.31	51.63	100.85	124.49	167.98	94.65
LTD [28]	16.22	32.94	62.73	74.60	92.84	55.87	28.83	48.73	87.37	104.82	135.61	81.07
TBIformer [35]	16.96	35.09	67.95	81.22	103.02	60.85	30.59	52.55	95.63	115.19	150.33	88.86
MRT [52]	15.32	32.07	61.84	74.04	94.59	55.57	27.79	47.91	87.01	104.80	137.22	80.95
SocialTGCN [37]	16.79	32.71	62.61	75.24	99.15	57.30	31.14	50.58	89.18	106.95	140.68	83.71
JRTransformer [55]	8.40	21.14	46.20	57.63	76.94	42.06	13.57	28.01	58.47	73.27	101.04	54.87
MPFSIR (our) [43]	9.15	23.05	52.31	65.49	92.46	48.49	15.56	30.55	64.84	81.81	114.94	61.54
Future Motion [51]	16.94	34.83	68.45	83.33	108.03	62.32	30.51	52.37	96.06	116.88	156.04	90.37
SoMoFormer [48]	9.43	23.88	54.78	68.71	92.38	49.84	15.22	31.08	67.33	85.37	119.37	63.67
GCN-Transformer (our) [45]	**8.32**	**20.84**	**44.56**	**54.81**	**74.66**	**40.64**	**13.37**	**27.63**	**57.27**	**71.25**	**97.71**	**53.45**

Interestingly, the performance ranking on ExPI shifted compared to the SoMoF Benchmark. JRTransformer, which was less competitive on SoMoF, emerged as a closer contender on ExPI, showing improved results in both short and long-term forecasting. Meanwhile, SoMoFormer, which had been a formidable competitor on SoMoF, experienced a marked decline in performance on ExPI, particularly in long-term prediction intervals. This demonstrates that the SoMoFormer model may be more sensitive to dataset characteristics, possibly overfitting to specific data types or motion patterns in the SoMoF Benchmark, while struggling to generalize to the more diverse and dynamic movements present in ExPI.

Furthermore, the Future Motion model, which performed well on SoMoF, was substantially outperformed by most models on ExPI. This highlights a critical limitation in the Future Motion model, which seems heavily reliant on the specifics of the training data and lacks the robustness necessary to handle diverse and complex motion patterns.

In terms of movement realism, GCN-Transformer once again excelled, as depicted in Figure 5. It managed to predict realistic, dynamic movements, where other models, such as JRTransformer and SoMoFormer, fell back on repeating the last known pose or generating erratic, unrealistic motion sequences.



(a)

(b)

Figure 5: The figure shows predicted poses on two example sequence from the ExPI test set, comparing the top-performing models: JRTransformer, SoMoFormer, and GCN-Transformer, alongside the ground truth (GT). Sequence (**a**) shows one person jumping off the shoulders of another, while sequence (**b**) shows one person performing a cartwheel assisted by another. While JRTransformer and SoMoFormer tend to repeat the last known pose, leading to less accurate predictions, the GCN-Transformer successfully generates more realistic and dynamic movements [45].

Across both the SoMoF Benchmark and ExPI datasets, GCN-Transformer demonstrated consistent improvements in predictive performance and generalization over prior state-of-the-art models. Its ability to handle complex multi-person and extreme motion scenarios makes it versatile for future pose forecasting tasks. Despite the varying nature of the datasets, GCN-Transformer's robust architecture allows it to excel in both short and long-term motion predictions, showcasing its adaptability and effectiveness in diverse settings.

2.4. A loss function for effective training of pose forecasting models

The training process of the GCN-Transformer model is designed to optimize its ability to forecast future poses by minimizing the difference between predicted and ground truth pose sequences. To achieve this, the model utilizes a combination of standard and novel loss function terms that effectively enhance its performance in multi-person pose forecasting. The standard approach for training pose forecasting models involves minimizing the reconstruction error (REC), which measures the difference between predicted and ground truth poses. This is typically calculated using the L_2-norm to minimize the Euclidean distance between the predicted and actual poses. While this method is effective for general pose prediction, it does not explicitly capture the interpersonal dynamics in multi-person scenarios.

To improve the model's capability to represent social interactions, in [45], we introduce the Multi-person joint distance loss (MPJD). This novel loss term encourages the model to learn the spatial relationships between different individuals within a scene by penalizing errors in the distances between corresponding joints of different people. By including this additional loss term, the model is driven to better model the social dependencies and interactions between individuals, which are crucial for realistic multi-person pose forecasting.

In addition to the MPJD loss, the novel loss function also incorporates a Velocity loss (VL) to prioritize learning smooth and coherent pose trajectories over time. Rather than focusing solely on accurately predicting discrete poses at specific time intervals, this loss encourages the generation of realistic motion sequences by ensuring that the predicted velocity of each joint is close to the ground truth velocity. This approach produces more fluid and natural pose transitions in the final predictions.

The overall loss function combines these three components: Reconstruction loss (REC), Multi-person joint distance loss (MPJD), and Velocity loss (VL), to create a comprehensive optimization objective. The MPJD and its corresponding Velocity loss are scaled by a factor γ, allowing for control over the influence of the social interaction terms in the final optimization. In practice, the γ parameter is set to 0.1, balancing the impact of the interpersonal distance losses relative to the reconstruction and velocity losses.

Mathematically, the loss function is defined as:

$$\mathcal{L}_{\text{REC}} = \frac{1}{N}\sum_{n=1}^{N}\|\hat{y}_n - y_n\|_2 \tag{6}$$

$$\mathcal{L}_{\text{MPJD}} = \frac{1}{N(N{-}1)}\sum_{n=1}^{N}\sum_{p=1}^{N}\|(\hat{y}_n - \hat{y}_p) - (y_n - y_p)\|_2 \tag{7}$$

$$\mathcal{L}_{\text{REC\_VL}} = \frac{1}{N}\sum_{n=1}^{N}\|\Delta\hat{y}_n - \Delta y_n\|_2 \tag{8}$$

$$\mathcal{L}_{\text{MPJD\_VL}} = \frac{1}{N(N-1)}\sum_{n=1}^{N}\sum_{p=1}^{N}\left\|\Delta\hat{d}_{n,p} - \Delta d_{n,p}\right\|_2 \tag{9}$$

$$\mathcal{L} = \mathcal{L}_{\text{REC}} + \mathcal{L}_{\text{REC\_VL}} + \mathcal{L}_{\text{MPJD}} \times \gamma + \mathcal{L}_{\text{MPJD\_VL}} \times \gamma \tag{10}$$

where N represents the number of individuals within the scene. The predicted pose sequences for the n-th and p-th individuals are represented by $\hat{y}_n$ and $\hat{y}_p$, respectively, while their corresponding ground truth pose sequences are denoted as y_n and y_p. The symbol $\|\cdot\|_2$ refers to the Euclidean distance (L2 norm), and the average across all individuals in the scene is calculated by $\frac{1}{N}\sum_{n=1}^{N}$. Temporal differentiation is indicated by Δ, where $\Delta y_n = y_n^t - y_n^{t+1}$ for $t = 0, 1, \ldots, T-1$ and $\Delta\hat{y}_n = \hat{y}_n^t - \hat{y}_n^{t+1}$ for $t = 0, 1, \ldots, T-1$. The predicted velocities of the joint distances between individuals are denoted by $\Delta\hat{d}_{n,p}$, while $\Delta d_{n,p}$ corresponds to the ground truth velocities of the joint distances between individuals.

2.4.1. Ablation study

To demonstrate the effectiveness of the proposed loss function, an ablation study was conducted on the GCN-Transformer model. This experiment evaluated the influence of different components and methods on the model's performance in multi-person pose forecasting. The ablation study followed a step-by-step approach, where components were progressively integrated into the baseline model, and the results were recorded after each modification. The experiment established a baseline model that included the Scene module and the Spatio-Temporal Transformer Decoder. This initial model served as the foundation for comparison. The next step was to enhance the Spatio-Temporal Attention Forecasting Module by adding the Temporal-GCN, which led to a noticeable improvement in the model's overall performance, particularly by refining the long-term forecasting performance. Following this, the Multi-person joint distance (MPJD) loss was introduced, further enhancing the model's ability to accurately predict short-term and long-term poses. The MPJD loss contributed to the model's effectiveness in capturing the spatial relationships between individuals, improving the overall prediction quality. The Velocity loss (VL) component was added to refine the model further. This addition marginally improved overall performance by encouraging the model to produce smoother and more consistent pose trajectories. Although the inclusion of the Velocity loss slightly compromised the short-term performance, it improved the long-term motion predictions and intra-sequence continuity.

Lastly, data augmentation techniques were integrated, resulting in the most substantial boost in performance across all time intervals. Data augmentation improved the model's generalization capabilities and improved predictions, significantly enhancing the short-term, mid-term, and long-term forecasting performance. The results of the ablation study, presented in Table 6, showcase the cumulative impact of each component. The integration of MPJD loss and Velocity loss, combined with data augmentation, proved to be highly effective in enhancing the model's forecasting performance, particularly in capturing realistic motion patterns across varying prediction horizons.

Table 6: The table presents the outcomes of the ablation study conducted on the SoMoF Benchmark validation set, evaluated using VIM (top) and MPJPE (bottom) metrics. The baseline configuration consists of the Scene Module and the Spatio-Temporal Transformer Decoder, with further components being added step by step to measure their impact. All models are trained exclusively on the SoMoF Benchmark training data without utilizing the AMASS dataset [45].

Metric	Method	100ms	240ms	500ms	640ms	900ms	Overall
VIM	Baseline	15.39	28.53	55.90	68.72	93.92	52.49
	+ Temporal-GCN	12.69	28.96	58.96	69.74	89.56	51.98
	+ **MPJD loss**	11.08	28.80	57.52	67.55	87.95	50.58
	+ **Velocity loss**	12.21	28.30	56.12	66.42	87.67	50.14
	+ Augmentation	**7.56**	**19.66**	**44.72**	**56.08**	**75.12**	**40.63**
MPJPE	Baseline	31.81	45.19	77.03	93.68	127.60	75.06
	+ Temporal-GCN	23.99	41.47	79.33	96.38	127.61	73.76
	+ **MPJD loss**	18.09	37.54	76.08	92.69	123.51	69.58
	+ **Velocity loss**	22.79	39.90	75.28	91.15	121.77	70.18
	+ Augmentation	**11.68**	**24.35**	**53.50**	**68.34**	**96.97**	**50.97**

The ablation study confirms the effectiveness of the proposed loss functions, where the overall VIM metric improved from 51.98 to 50.14, demonstrating a 3.5% reduction in prediction error after incorporating the MPJD and Velocity loss during training. Similarly, for the MPJPE metric, the overall score decreased from 73.76 to 70.18, reflecting a 4.8% reduction in prediction error.

2.5. An evaluation metric for pose forecasting

To evaluate the performance of pose forecasting models, it is essential to use metrics that capture how well the predicted poses match the actual ground truth in terms of error and realism. The evaluation metrics serve to assess prediction error and to provide insights into how well models can replicate human motion dynamics.

In the early stages of pose forecasting, metrics borrowed from related fields, particularly pose estimation, were often employed. One of the most widely used metrics is the Mean Per Joint Position Error (MPJPE). MPJPE calculates the average Euclidean distance (L2 norm) between the predicted and actual joint positions over all joints in the pose sequence. This metric provides an overall measure of the model's performance in predicting joints but is limited by its lack of focus on temporal dynamics, essentially treating each pose in isolation rather than evaluating how well the model predicts movement over time.

Recognizing the limitations of MPJPE, some works sought to address these shortcomings by introducing more nuanced evaluation metrics. For example, Adeli et al. proposed the Visibility-Ignored Metric (VIM) in their work [2], which evaluates pose forecasting performance by focusing solely on the final predicted pose. While VIM prioritizes the last pose of the sequence, it overlooks the quality of the model's predictions in earlier poses, potentially neglecting important aspects of motion forecasting, such as how well the predicted trajectory mirrors natural human motion.

Another attempt to improve upon MPJPE was introduced by Šajina and Ivasic-Kos in [43] with the Movement-Weighted Mean Per Joint Position Error (MW-MPJPE). This metric extends MPJPE by incorporating a weighting factor that considers the magnitude of motion in the target sequence, thus differentiating between dynamic and static poses. The goal is to emphasize joints undergoing significant movement, improving models' evaluation in predicting realistic motion dynamics. Still, the focus remains primarily on evaluating pose error without a deep temporal assessment.

Specifically for the domain of multi-person pose forecasting, Peng, Mao, and Wu in [35] employed a combination of evaluation metrics. Among these were the Joint Position Error (JPE), which is similar to MPJPE but generalizes to all individuals in a multi-person scenario; Aligned Mean Per Joint Position Error (APE), akin to Root-MPJPE, which removes global translation to focus on the relative joint positions; and Final Displacement Error (FDE), which measures the error in global movement trajectory by focusing on the difference between the final predicted position and the ground truth.

Although metrics such as MPJPE, VIM, and MW-MPJPE offer important insights into the performance of pose forecasting models, they often focus on limited aspects of the task. For instance, MPJPE treats each frame independently, while VIM concentrates solely on the final pose, and MW-MPJPE adds movement consideration but still lacks a detailed temporal evaluation. These limitations mean that the full complexity of pose forecasting, which involves predicting both accurate static poses and dynamic motion sequences, is not fully captured. Consequently, the choice of metric can significantly influence model rankings, with no single metric providing a definitive evaluation of overall performance. In recent work [45], we aimed to overcome these shortcomings by proposing a more comprehensive evaluation metric called the Final Joint Position and Trajectory Error (FJPTE), which offers a more holistic assessment of pose forecasting performance.

FJPTE evaluates model performance across four key components:

- Final global position error: This component assesses the error in the final predicted global position (e.g., the pelvis) using the Euclidean distance between the predicted and actual positions.

- Global movement trajectory error: This component evaluates the error of the model's prediction of global movement over time, measuring the Euclidean distance of the temporal differentiation of the root joint (usually the pelvis).

- Final pose position error: This component removes global movement and evaluates the error of the final pose using the Euclidean distance between the predicted and ground-truth joint positions.

- Pose trajectory error: This component assesses the error of the pose trajectory, excluding global movement, by measuring the Euclidean distance of the temporal differentiation across all joints in the sequence.

By incorporating these four components, Final Joint Position and Trajectory Error (FJPTE) metric provides a comprehensive and balanced evaluation of pose forecasting models, directly assessing human movement dynamics. This approach ensures that models are evaluated on their ability to predict individual joint positions and on their proficiency in forecasting natural and realistic human movements across time. This results in a more complete and nuanced understanding of model performance, addressing previous shortcomings in pose forecasting evaluation. An illustrative comparison of joint movement evaluation using different metrics is presented in Figure 6.

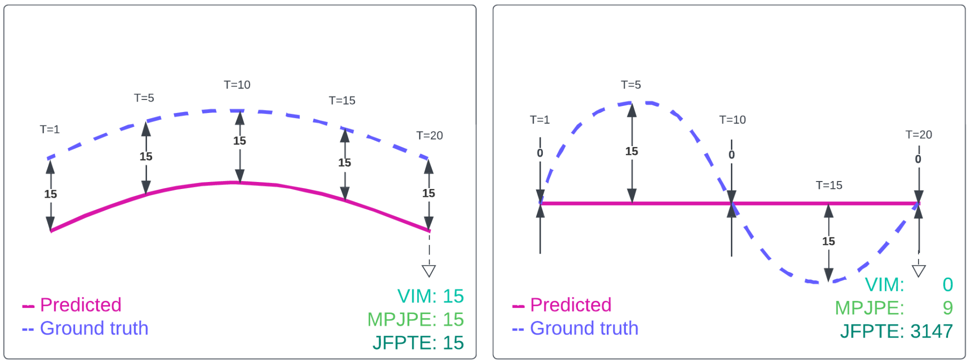


Figure 6: The figure depicts predicted (purple) and ground truth (blue) joint trajectories, with T representing the time step and the values between the trajectories indicating their distances at each time step. When the trajectories are similar but slightly shifted, FJPTE provides results equivalent to MPJPE and VIM. However, when the trajectories differ significantly, the metrics diverge in their evaluations. While MPJPE and FJPTE assess the entire joint trajectory, VIM only considers the error at the final time step, $T = 20$ [45].

Figure 7 highlights a scenario in which the FJPTE metric offers a more informative assessment of model performance compared to traditional metrics like MPJPE or VIM. In the illustrated example, the predicted sequence maintains a correct global trajectory, yet the individual pose remains unnaturally static, resembling a ghost-like motion drifting through space, a common failure mode in pose forecasting. Whereas MPJPE calculates average joint position error over time and VIM considers only the final frame's accuracy, FJPTE evaluates both the internal motion dynamics ($\text{FJPTE}_{\text{local}}$) and the global movement path ($\text{FJPTE}_{\text{global}}$). This dual perspective allows FJPTE to reveal whether a model's shortcomings lie in predicting realistic motion or in maintaining correct global displacement. By integrating both aspects into a single score, FJPTE supports a more complete and reliable evaluation of forecasting performance.

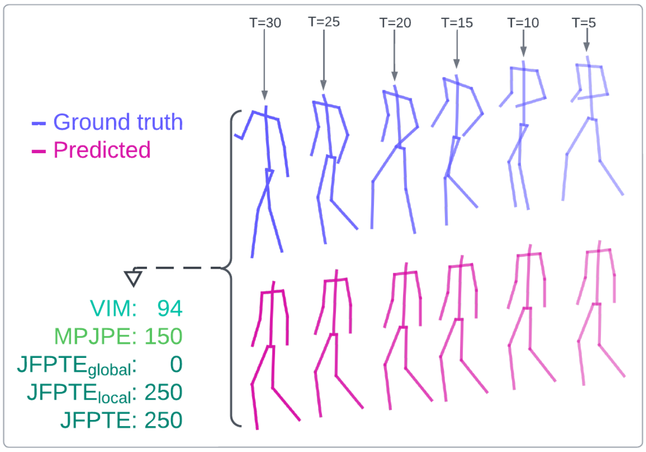


Figure 7: The figure shows an example comparing predicted (purple) and ground truth (blue) pose sequences across time, where T denotes the time interval. While the global path of the predicted sequence closely follows the ground truth, the poses remain unnaturally static, highlighting a typical challenge in pose forecasting. Unlike MPJPE and VIM, which either average joint errors over time or consider only the final frame, FJPTE provides a more detailed analysis by capturing both the quality of joint trajectories and the distinction between local motion (FJPTE$_{\text{local}}$) and overall displacement (FJPTE$_{\text{global}}$). MPJPE and FJPTE evaluate the entire sequence, whereas VIM focuses only on the final time interval at $T = 30$ [45].

FJPTE is calculated as follows:

$$\begin{gathered}
E_{position}(\hat{y}, y) = \frac{1}{J}\sum_{j=1}^{J}\|\hat{y}(j) - y(j)\|_2 \\
E_{trajectory}(\hat{Y}, Y) = \frac{1}{T-1}\sum_{t=1}^{T-1} E_{position}(\hat{Y}^t - \hat{Y}^{t+1}, Y^t - Y^{t+1}) \\
E_{global}(\hat{Y}, Y) = (E_{trajectory}(\hat{Y}_{\varphi_{pelvis}}, Y_{\varphi_{pelvis}}) + E_{position}(\hat{Y}^T_{\varphi_{pelvis}}, Y^T_{\varphi_{pelvis}})) \times 1000 \\
E_{local}(\hat{Y}, Y) = (E_{trajectory}(\hat{Y} - \hat{Y}_{\varphi_{pelvis}}, Y - Y_{\varphi_{pelvis}}) + E_{position}(\hat{Y}^T - \hat{Y}^T_{\varphi_{pelvis}}, Y^T - Y^T_{\varphi_{pelvis}})) \times 1000 \\
E_{\text{FJPTE}}(\hat{Y}, Y) = E_{global}(\hat{Y}, Y) + E_{local}(\hat{Y}, Y)
\end{gathered} \tag{11}$$

where $\hat{y}$ represents the predicted sequence, and y represents the ground truth sequence. The variable J indicates the number of joints, while T refers to the number of time steps. The term $\|\cdot\|_2$ denotes the Euclidean distance (L2 norm), and $\frac{1}{T-1}\sum_{t=1}^{T-1}$ calculates the average error across all time steps. The notation $E_{global}(\hat{Y}, Y)$ measures the global position and trajectory error between the predicted and ground truth sequences, focusing on the pelvis joint. Similarly, $E_{local}(\hat{Y}, Y)$ captures the error related to local motion dynamics, excluding the pelvis joint and overall global movement. The metric $E_{\text{FJPTE}}(\hat{Y}, Y)$ integrates both local and global errors into a single evaluation metric.

2.5.1. Experimental results using the FJPTE

In our evaluation of the SoMoF Benchmark and ExPI datasets using the proposed FJPTE metric and its respective components $\text{FJPTE}_{\text{local}}$ and $\text{FJPTE}_{\text{global}}$, we observed several key insights that highlight the advantages of using these metrics compared to standard metrics such as VIM and MPJPE.

On the SoMoF Benchmark dataset as presented in Table 7, GCN-Transformer significantly outperformed other models when evaluated using $\text{FJPTE}_{\text{local}}$, which measures movement dynamics. Its superior performance demonstrated the model's advanced capability to capture and predict the intricate dynamics of human movement and interaction over different forecasting intervals. While traditional metrics like VIM and MPJPE also placed GCN-Transformer at the top, $\text{FJPTE}_{\text{local}}$ offered a more nuanced perspective by clearly showcasing the model's edge in short-term and fine-grained movement dynamics over the alternatives.

In contrast, when using the $\text{FJPTE}_{\text{global}}$ metric, which evaluates global movement and position error, SoMoFormer exhibited a slight advantage in long-term predictions, reflecting its strength in forecasting global trajectories over extended time horizons. This distinction was not as apparent in standard metrics such as VIM and MPJPE, which typically aggregate various aspects of prediction performance without providing insight into the trade-offs between local movement dynamics and global trajectory error. By using $\text{FJPTE}_{\text{global}}$, we uncovered subtle differences in model performance, such as MPFSIR's unexpectedly strong results in global position forecasting, outperforming Future Motion by a significant margin.

On the ExPI dataset, as presented in Table 8, a similar trend emerged. GCN-Transformer again outperformed all other models across most intervals when evaluated using FJPTE$_{\text{local}}$, demonstrating its robust modeling of movement dynamics, with JR-Transformer being its closest competitor at the 120ms interval. In comparison, SoMo-Former struggled significantly, as was also seen in the results using traditional metrics, though the detailed breakdown provided by FJPTE$_{\text{local}}$ further highlighted its challenges in capturing short-term dynamics effectively.

Evaluation using FJPTE$_{\text{global}}$ provided additional insight into long-term prediction capabilities. Despite JRTransformer's strong short-term performance, GCN-Transformer

Table 7: Performance comparison on the SoMoF Benchmark test set using the proposed FJPTE metric. Lower values correspond to better performance. The table separates FJPTE$_{\text{local}}$, which captures errors in movement dynamics, from FJPTE$_{\text{global}}$, which assesses global position and trajectory errors. The model marked with an asterisk (*) used the validation set during training [45].

Method	FJPTE$_{\text{local}}$						FJPTE$_{\text{global}}$					
	100ms	240ms	500ms	640ms	900ms	Overall	100ms	240ms	500ms	640ms	900ms	Overall
Zero Velocity	65.36	97.18	142.35	158.79	178.72	128.48	91.12	146.51	241.69	284.08	363.52	225.38
DViTA [33]	55.15	91.84	147.91	168.07	194.29	131.45	47.60	81.35	162.46	212.71	319.11	164.65
LTD [28]	48.96	78.96	127.59	145.98	170.41	114.38	52.86	88.66	159.64	201.40	290.96	158.70
TBIformer [35]	55.24	88.28	138.76	156.81	178.97	123.61	51.19	84.53	150.47	190.78	283.36	152.07
MRT [52]	56.38	90.59	143.17	162.19	186.11	127.69	46.74	77.70	147.95	189.65	279.84	148.37
SocialTGCN [37]	51.50	83.54	137.45	157.54	183.19	122.64	39.76	65.92	132.28	175.90	271.09	136.99
JRTransformer [55]	41.20	72.47	124.75	145.87	174.81	111.82	26.87	54.81	122.92	166.64	264.94	127.24
MPFSIR (our) [43]	43.53	75.36	127.59	148.60	180.67	115.15	27.37	51.27	109.84	151.17	248.05	117.54
Future Motion [51]	42.74	72.22	122.18	140.77	165.83	108.75	31.04	54.72	117.86	158.93	249.45	122.40
SoMoFormer [48]	37.69	65.48	111.48	128.79	154.44	99.58	26.13	48.37	**104.01**	**139.66**	**217.92**	**107.22**
GCN-Transformer (our) [45]	**37.22**	**63.78**	**109.06**	**126.12**	**152.72**	**97.78**	**24.35**	**47.42**	107.12	146.38	234.51	111.96
GCN-Transformer* (our) [45]	**36.76**	**62.29**	**104.96**	**121.68**	**147.97**	**94.73**	**23.63**	**45.89**	**102.05**	**138.45**	228.94	107.79

Table 8: Performance comparison on the ExPI test set using the proposed FJPTE metric. Lower values correspond to better performance. The table separates FJPTE$_{\text{local}}$, which captures errors in movement dynamics, from FJPTE$_{\text{global}}$, which assesses global position and trajectory errors [45].

Method	FJPTE$_{\text{local}}$						FJPTE$_{\text{global}}$					
	120ms	280ms	600ms	760ms	1080ms	Overall	120ms	280ms	600ms	760ms	1080ms	Overall
Zero Velocity	76.63	119.52	182.09	205.19	240.31	164.75	79.80	127.56	201.88	230.77	280.05	184.01
DViTA [33]	56.91	101.25	176.21	206.20	252.27	158.57	45.58	83.58	164.19	202.36	271.01	153.34
LTD [28]	60.27	97.73	159.16	182.82	217.66	143.53	47.42	80.89	141.84	169.41	215.70	131.05
TBIformer [35]	67.38	109.04	174.85	200.29	239.29	158.17	50.23	86.97	155.57	184.96	238.15	143.18
MRT [52]	65.77	107.77	173.87	199.12	236.71	156.65	43.80	75.45	133.75	162.58	214.24	125.96
SocialTGCN [37]	72.62	110.05	174.62	201.84	247.24	161.27	52.04	83.27	149.11	178.12	237.98	140.10
JRTransformer [55]	**37.98**	71.62	130.94	155.35	197.44	118.67	**26.21**	**52.63**	102.44	126.11	168.75	95.23
MPFSIR (our) [43]	41.12	77.88	145.78	174.01	225.03	132.76	27.21	54.68	112.28	140.63	207.33	108.43
Future Motion [51]	64.87	105.26	175.12	206.69	247.48	159.88	48.70	86.51	160.21	197.70	270.41	152.71
SoMoFormer [48]	41.91	80.52	150.92	179.58	224.17	135.42	28.82	57.92	118.39	148.45	204.18	111.55
GCN-Transformer (our) [45]	38.39	**71.60**	**125.41**	**146.24**	**181.17**	**112.56**	26.67	52.74	**100.23**	**122.83**	**172.73**	**95.04**

achieved superior overall results across broader time intervals. This illustrates how FJPTE$_{\text{global}}$ helped clarify the comparative strengths of different models over varying timeframes, emphasizing long-term forecasting, which standard metrics alone might not emphasize to the same extent.

When evaluating the models using the combined FJPTE metric, as presented in Table 9, which integrates both FJPTE$_{\text{local}}$ and FJPTE$_{\text{global}}$, we gained a holistic view of each model's forecasting capabilities. On the SoMoF Benchmark dataset, SoMoFormer emerged as the best-performing model overall, except for GCN-Transformer*, which included the validation set during training. This result was consistent with observations from both the individual FJPTE$_{\text{local}}$ and FJPTE$_{\text{global}}$ metrics, where SoMoFormer excelled in global trajectory forecasting but lagged slightly in short-term movement dynamics.

Table 9: Performance comparison on the SoMoF Benchmark test set (left) and the ExPI test set (right) using the FJPTE metric, where lower scores denote better performance. The table shows combined FJPTE$_{\text{local}}$ and FJPTE$_{\text{global}}$ errors for a more holistic assessment of model performance. GCN-Transformer model demonstrates state-of-the-art performance based on the FJPTE metric. The model incorporating the validation set during training is marked with an asterisk (*) [45].

Method	SoMoF Benchmark						ExPI					
	100ms	240ms	500ms	640ms	900ms	Overall	120ms	280ms	600ms	760ms	1080ms	Overall
Zero Velocity	156.48	243.69	384.04	442.87	542.24	353.86	156.43	247.07	383.97	435.95	520.36	348.76
DViTA [33]	102.75	173.20	310.36	380.78	513.40	296.10	102.48	184.82	340.40	408.56	523.29	311.91
LTD [28]	101.82	167.62	287.23	347.38	461.37	273.08	107.69	178.62	301.01	352.23	433.36	274.58
TBIformer [35]	106.43	172.81	289.23	347.59	462.33	275.68	117.61	196.01	330.42	385.25	477.45	301.35
MRT [52]	103.11	168.29	291.12	351.84	465.95	276.06	109.58	183.22	307.63	361.70	450.95	282.62
SocialTGCN [37]	91.26	149.46	269.73	333.44	454.28	259.63	124.66	193.32	323.73	379.95	485.22	301.38
JRTransformer [55]	68.07	127.29	247.68	312.51	439.75	239.06	**64.19**	**124.25**	233.39	281.46	366.19	213.90
MPFSIR (our) [43]	70.91	126.63	237.44	299.78	428.72	232.69	68.33	132.56	258.06	314.65	432.35	241.19
Future Motion [51]	73.78	126.94	240.04	299.70	415.28	231.15	113.57	191.77	335.33	404.39	517.89	312.59
SoMoFormer [48]	63.82	113.85	**215.50**	**268.45**	**372.35**	**206.79**	70.73	138.44	269.31	328.03	428.35	246.97
GCN-Transformer (our) [45]	**61.57**	**111.21**	216.17	272.50	387.22	209.73	65.07	124.34	**225.64**	**269.07**	**353.90**	**207.60**
GCN-Transformer* (our) [45]	**60.39**	**108.19**	**207.01**	**260.13**	**376.91**	**202.53**	-	-	-	-	-	-

On the ExPI dataset, GCN-Transformer was the top-performing model overall when evaluated with the combined FJPTE metric, reaffirming its broad capability in both short-term and long-term forecasting. Although JRTransformer slightly outperformed GCN-Transformer in short-term dynamics, the combined FJPTE metric highlighted GCN-Transformer's superior performance across longer time intervals.

Using the FJPTE metric provided a richer evaluation framework than standard metrics like VIM and MPJPE. By breaking down performance into local movement dynamics and global position errors, $\text{FJPTE}_{\text{local}}$ and $\text{FJPTE}_{\text{global}}$ offered more specific and actionable insights into each model's strengths and weaknesses. This allowed for a more detailed comparison of models across different performance dimensions, revealing distinctions that would otherwise be obscured by aggregate evaluation metrics.

2.6. Pipeline for real-world application of pose forecasting

Implementing a pose forecasting pipeline for real-world applications involves several key stages, each crucial for accurately predicting future human poses from video data captured by a monocular camera. The process begins with recording a video using a monocular camera. This type of camera captures 2D images from a single viewpoint, which serves as the foundation for subsequent pose estimation and forecasting tasks. Each frame from the recorded video undergoes 2D pose estimation to detect and identify the keypoints of the human skeleton. This stage leverages deep learning models to extract the 2D coordinates of joints from each image, where the goal is to accurately map out the human body's joints and their spatial relationships in two dimensions. Once 2D keypoints are obtained, they are transformed into 3D coordinates using 2D-to-3D pose estimation models. This lifting process typically involves applying deep learning techniques that infer depth information from the 2D poses, resulting in a 3D representation of the human body. In dynamic scenes, such as sports events, it is essential to track individuals across frames to maintain the continuity of the pose sequences. Multiple Object Tracking (MOT) algorithms are used to assign consistent identifiers to each person in the video. This ensures that the sequences of poses belong to the correct individuals throughout the video, even as they move and interact. With individuals consistently tracked, sequences of poses are collected for each person. This involves grouping the 3D poses frame by frame, creating a continuous series that represents the person's movement over time. Finally, the collected sequences are then fed into the pose forecasting model to predict future poses based on the collected sequences of poses. The procedure for obtaining a sequence of poses and forecasting future poses is shown in Figure 8.

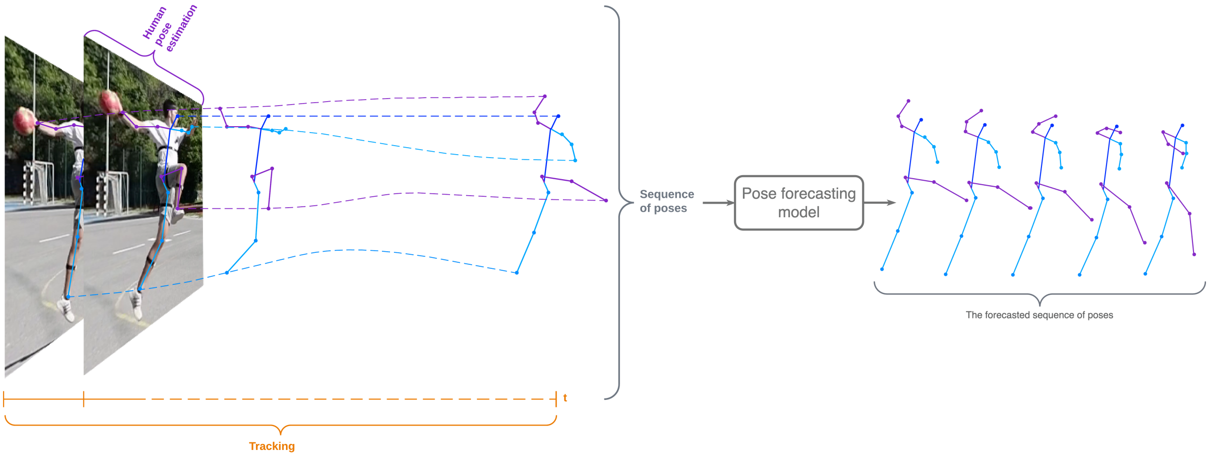


Figure 8: Creating a sequence of poses using human pose estimation to produce human skeleton keypoints and object tracking for grouping collected poses across frames (t) into a single sequence of poses. After that, the sequence of poses is then fed into the pose forecasting model to produce a forecasted sequence of poses (figure based on [44]).

2.6.1. 2D pose estimation

Human pose estimation (HPE) has developed significantly over the years, transitioning from traditional methods reliant on handcrafted features to deep learning approaches that have revolutionized the field. Initially, traditional methods for 2D pose estimation utilized low-level features such as Histogram of Oriented Gradients (HOG), contours, and color histograms, combined with machine learning algorithms like Random Forests to detect and classify body joints. These methods, however, struggled with occlusions and cases where body parts were not clearly visible, limiting their applicability in real-world scenarios. The advent of deep learning marked a significant shift in HPE. Toshev and Szegedy pioneering work, DeepPose [47], introduced the first deep convolutional neural network (CNN) for human pose estimation. The model directly regressed the coordinates of body joints from images, demonstrating that deep networks could effectively model hidden and occluded joints, significantly improving accuracy compared to traditional approaches. This work laid the foundation for subsequent research, broadly categorizing deep learning-based methods into single-person and multi-person approaches.

In the single-person approach, the problem is framed as a regression task, where the goal is to predict the keypoints of a person in an image. This approach can be further divided into direct regression-based frameworks and heatmap-based frameworks. In the direct regression framework, as seen in DeepPose, the network directly outputs the coordinates of keypoints. Carreira et al. in [9] refined this approach by introducing an iterative error feedback mechanism that significantly enhanced accuracy. Luvizon, Tabia, and Picard in [24] further improved the regression-based method by integrating a soft Argmax function to convert feature maps into keypoint coordinates, achieving results competitive with heatmap-based methods. Conversely, the heatmap-based framework, which generates heatmaps indicating the likelihood of keypoints at various locations in the image, has been widely adopted due to its robustness. Notable contributions include Newell, Yang, and Deng stacked hourglass network [32], which emphasized the importance of repeated bottom-up and top-down processing to refine pose predictions. This architecture became a cornerstone for many subsequent HPE models.

Multi-person pose estimation presents additional challenges due to the unknown number and people's positions in an image. Approaches to this problem are generally categorized into top-down and bottom-up pipelines. The top-down approach first detects individual persons within an image, typically using object detection methods, and then applies a single-person pose estimation model to each detected region. The Mask R-CNN model by He et al. in [14] is a prominent example, extending the Faster R-CNN model [40] to predict both object masks and keypoints, thereby streamlining the process of multi-person pose estimation. In contrast, the bottom-up approach first detects all keypoints in an image and then groups them into individual persons. Pishchulin et al. with DeepCut [38] introduced this paradigm by formulating the task as a joint problem of partitioning and labeling keypoints, accounting for geometric and visual constraints. Subsequent improvements, such as Part Affinity Fields (PAFs) proposed by Cao et al. in [7], refined the bottom-up approach by learning associations between detected keypoints, allowing for real-time multi-person pose estimation.

2.6.2. 3D pose estimation

Pose forecasting models typically require 3D pose data, which can be obtained either by lifting 2D poses into 3D space or by directly applying 3D pose estimation on images, bypassing the need for intermediate 2D pose estimation. Early research focused on directly predicting 3D poses from images. Li and Chan in [22] were pioneers in this area, using deep learning to regress 3D keypoints directly from images. Their approach demonstrated that convolutional neural networks (CNNs) could outperform traditional methods by learning the complex spatial relationships between different body parts without relying on handcrafted features or explicit correlation constraints. Following this, Tekin et al. in [46] extended this concept by introducing auto-encoders in the latent space to represent 3D poses. By training an auto-encoder to reconstruct 3D poses, they leveraged the latent space representation to capture the inherent constraints of the human body, which improved pose consistency and reduced errors in keypoint localization. However, direct prediction of 3D poses from images posed challenges, particularly in terms of accuracy and error propagation from earlier stages of processing. To address these issues, Martinez et al. in [29] proposed a paradigm shift by lifting 2D keypoints into 3D space using a deep feed-forward network. This approach separated the 2D pose estimation from the 3D pose reconstruction, allowing for independent optimization of each step. This not only improved accuracy but also provided a clearer pathway for debugging and enhancing each component of the pipeline.

Recent studies have increasingly focused on exploiting temporal information from sequences of images to improve 3D pose estimation. Rayat Imtiaz Hossain and Little in [39] introduced a sequence-to-sequence network using layer-normalized LSTM units, which utilized temporal context from previous frames to generate more consistent 3D pose predictions across a sequence. This method reduced errors that typically accumulate over time, providing more stable and reliable 3D reconstructions. Building on the importance of temporal data, Pavllo et al. in [34] introduced a method based on dilated temporal convolutions applied to 2D keypoint trajectories. Their approach combined the temporal context with a semi-supervised learning strategy that utilized unlabelled video data to enhance performance, especially in scenarios with limited labeled data. This method addressed issues of pose drift over long sequences, a common problem in LSTM-based models.

Occlusions and missing data due to out-of-frame targets have been another significant challenge in 3D pose estimation. Cheng et al. in [10] tackled this by integrating graph convolutional networks (GCNs) and temporal convolutional networks (TCNs). Their method modeled the relationships between bones and joints through a human-bone GCN and a human-joint GCN, which enabled the system to estimate 3D poses even in the presence of occlusions. By incorporating a joint TCN for spatial consistency and a velocity TCN for temporal smoothness, they achieved robust performance without relying on camera parameters, making the system more versatile and adaptable. Finally, the problem of dataset bias in 2D-to-3D networks was addressed by Li et al. in [21], who proposed a novel augmentation method capable of synthesizing a vast amount of training data. Their data evolution strategy generated new poses by applying mutations and crossovers to existing data, which significantly expanded the diversity and coverage of training datasets. This approach, combined with a cascaded 3D coordinate regression model, provided a scalable solution to improve the generalization of 3D pose estimation models across different datasets and scenarios.

2.6.3. Tracking

A multiple object tracking (MOT) algorithm is essential to accurately track the positions and maintain consistent identities of multiple objects across frames, ensuring reliable sequences of poses. This challenge becomes particularly pronounced in dynamic and crowded environments, where objects frequently change direction, speed, and visibility, often leading to occlusions and identity switches. Over the years, various approaches have been developed to address these challenges, focusing on improving detection, motion modeling, and feature extraction.

Motion-based tracking methods have traditionally relied on techniques such as background subtraction and frame differencing to detect moving objects. These methods are computationally efficient and often robust under controlled conditions but can struggle in more complex environments. A classical example of motion-based tracking is the Kalman filter [20], which has been widely used to estimate the position of objects by predicting their future states based on their previous movements. The Kalman filter assumes linear motion and Gaussian noise, which makes it suitable for applications like tracking vehicles on the road but less effective in scenarios involving erratic movements. To address the limitations of the Kalman filter, SORT (Simple Online and Realtime Tracking) was introduced by Bewley et al. in [5], combining the Kalman filter with the Hungarian algorithm for data association. While SORT efficiently tracks objects in real time, it does not account for visual features, making it prone to identity switches, especially in crowded scenes.

To improve the robustness of motion-based tracking, optical flow methods have been employed, which estimate the motion of objects by analyzing the changes in pixel intensities between consecutive frames. Methods like those proposed by Lucas, Kanade, et al. in [23] calculate a 2D motion field, which is particularly useful in scenarios where objects undergo non-linear motion or where frame registration is imperfect. Various works have used optical flow, such as in [3] and [19], to separate foreground objects from the background, thereby improving tracking performance in cluttered environments.

On the other hand, feature-based tracking approaches focus on objects' appearance to maintain their identities over time. These methods segment objects based on features such as color, texture, and shape and track them by matching these features across frames. Wojke, Bewley, and Paulus in [53] extended the SORT framework by introducing a deep association metric, which captures object features within the bounding box to improve tracking accuracy, particularly during occlusions. This deep feature-based approach significantly reduces identity switches, a common issue in simpler motion-based trackers. Further advancements have integrated object segmentation techniques within the tracking pipeline, as seen in [49], where objects are segmented within detected bounding boxes to enhance the accuracy of the tracking process.

Pose tracking represents a more specialized branch of tracking, focusing on tracking human body poses across video frames. Iqbal, Milan, and Gall in [18] introduced the concept of Multi-Person PoseTrack, addressing the challenges of tracking multiple people in dynamic scenes by representing body joints as a spatiotemporal graph. Subsequent methods, such as PoseFlow by Xiu et al. in [54], further refined pose tracking by introducing techniques like Pose Flow Builder (PF-builder) and Pose Flow non-maximum suppression (PF-NMS), which stabilize pose tracking by associating poses across frames more effectively. More recent approaches, like those by Bao et al. in [4], have integrated pose estimation with tracking-by-detection frameworks, utilizing temporal information to improve detection accuracy and employing graph convolutional networks to model relationships between detected persons and their poses. These advancements have made pose tracking more robust to occlusions and complex human interactions, particularly in sports and other dynamic environments.

2.6.4. Evaluation of the pipeline on a Handball Jump Shot dataset

In this section, we evaluate our 3D pose estimation and pose forecasting pipeline using a custom dataset tailored for this purpose. The Rijeka Handball Shot (RI-HBS) dataset [44, 41] was compiled from handball scenes recorded during training sessions in Rijeka. Handball, a fast-paced Olympic team sport, is widely popular in Europe but underrepresented in publicly available datasets for sports scenes, making RI-HBS a valuable resource for this domain. The dataset consists of 21 short video clips, each averaging 9 seconds, capturing two different players executing several handball jump shots. To ensure the precision of the recorded joint positions, both players were equipped with Wear-Notch motion capture sensors, which boast a static accuracy of approximately 1–2° in yaw, tilt, and roll. The recording was performed with a single stationary camera positioned 1.5 meters above the ground, and the players were located 7–10 meters away from the camera, capturing video at a resolution of 1920x1080 pixels.

2.6.4.1. Evaluation of 3D Pose Estimation

For the 3D pose estimation evaluation, the RI-HBS dataset was prepared by synchronizing the motion capture data with the video frames. The ground truth positions of the players' joints, as recorded by the Wear-Notch sensors, were used to train and evaluate the 3D pose estimation models. The synchronization process involved aligning the video footage with the sensor data, ensuring that each video frame corresponded accurately to the recorded joint positions. This setup enabled us to assess the performance of 3D pose estimation models in a real-world sports context, characterized by rapid, dynamic movements and frequent occlusions typical of handball actions.

Several well-established and high-performing models were selected to assess their effectiveness on the RI-HBS dataset of handball players performing handball shots. The primary objective of this evaluation was to identify a combination of models that delivers the best overall results in an unseen sports environment, such as handball, which is characterized by dynamic and complex motions.

Four state-of-the-art models were considered for 2D pose estimation: PoseRegression [24], ArtTrack [17], Mask R-CNN [14], and UDP-Pose [15]. These models were chosen for their proven track records in accurately estimating 2D poses in various challenging scenarios. PoseRegression and ArtTrack were trained using the MPII training dataset, while Mask R-CNN and UDP-Pose were trained using the COCO 2017 training dataset. For 3D pose estimation, three prominent models were evaluated: GnTCN [10], EvoSkeleton [21], and VideoPose3D [34]. These models predict 3D poses based on the outputs of the 2D pose estimation models, creating a pipeline where 2D pose predictions are lifted to 3D. The 3D pose estimation models, GnTCN, EvoSkeleton, and VideoPose3D, were trained using the Human3.6M training dataset, ensuring a consistent basis for fair evaluation and comparison. Each of these models has distinct strengths: GnTCN leverages graph-based neural networks for capturing temporal dependencies, EvoSkeleton utilizes evolutionary algorithms to refine skeleton predictions, and VideoPose3D applies convolutional neural networks to temporal sequences of 2D poses for predicting 3D poses.

In total, 12 combinations of 2D and 3D models were evaluated to determine the best-performing pipeline for 3D pose estimation on the RI-HBS dataset. Experiments utilizing top-down methods were provided with ground truth bounding boxes to ensure that the assessment focused purely on pose estimation accuracy, eliminating potential errors from object detectors. The performance of different 3D pose estimation model combinations on the RI-HBS dataset is summarized in Table 10, with the best results highlighted in bold. The upper portion of the table presents results based on 2D pose estimation models pre-trained on the Human3.6M dataset, while the lower portion shows results after fine-tuning the Mask R-CNN and UDP-Pose models on the RI-HBS dataset. The two models were chosen for fine-tuning based on the best performance during the pre-trained evaluation, resulting in noticeable improvements after fine-tuning. Among all combinations, the UDP-Pose and EvoSkeleton models consistently performed the best, with the lowest PA-MPJPE score of 0.073 after fine-tuning. This demonstrates that fine-tuning the models on the RI-HBS dataset enhances the accuracy of 3D pose estimation in the specific context of handball player movements.

Table 10: Evaluation results of 3D pose estimation model combinations on the RI-HBS dataset. The best-performing results are highlighted in bold. ”PA” denotes Procrustes alignment, which uses Procrustes analysis to align and compare poses across all axes before evaluation [44].

Training of 2D models	2D Model	3D Model	▼PA-MPJPE
Pre-trained on Human3.6M	PoseRegression	GnTCN	0.150
	PoseRegression	EvoSkeleton	0.144
	PoseRegression	VideoPose3D	0.154
	ArtTrack	GnTCN	0.106
	ArtTrack	EvoSkeleton	0.107
	ArtTrack	VideoPose3D	0.151
	Mask R-CNN	GnTCN	0.098
	Mask R-CNN	EvoSkeleton	0.094
	Mask R-CNN	VideoPose3D	0.124
	UDP-Pose	GnTCN	0.083
	UDP-Pose	EvoSkeleton	**0.074**
	UDP-Pose	VideoPose3D	0.117
Fine-tuned on RI-HBS	Mask R-CNN	GnTCN	0.090
	Mask R-CNN	EvoSkeleton	0.086
	Mask R-CNN	VideoPose3D	0.118
	UDP-Pose	GnTCN	0.080
	UDP-Pose	EvoSkeleton	**0.073**
	UDP-Pose	VideoPose3D	0.115

2.6.4.2. Evaluation of Multi-Person Pose Forecasting models

In the context of multi-person pose forecasting, the RI-HBS dataset was further refined to facilitate the training and evaluation of pose forecasting models. The dataset was divided into a training set, comprising 7 scenes, and a test set, comprising 3 scenes, with both sets including two players performing handball shots. The motion capture data, recorded at 40 frames per second, provided approximately 4000 unique poses across the dataset, offering a robust foundation for pose forecasting tasks. The sequences were standardized to align with the SoMoF Benchmark, using a sampling frequency of 2, resulting in input sequences covering 775 milliseconds and output sequences spanning 675 milliseconds.

The evaluation of multi-person pose forecasting models on the RI-HBS dataset was conducted in two phases: first using models pre-trained on the 3DPW and AMASS datasets, and then fine-tuning these models specifically on the RI-HBS dataset with an additional 50 epochs. Previous research [41] has demonstrated that fine-tuning pre-trained models on a custom dataset can significantly enhance performance by allowing the models to adapt to the specific characteristics of the new data. Building on this finding, we applied fine-tuning to improve the models' ability to predict poses in the RI-HBS dataset. By doing so, we aimed to leverage the generalization capability obtained from pre-training on larger, diverse datasets while refining the models for the nuances of the RI-HBS test set. The results of these evaluations are detailed in Tables 11 and 12 respectively, with clear improvements after fine-tuning, validating the effectiveness of this approach.

In the pre-trained phase, as shown in Table 11, the GCN-Transformer and MPFSIR emerged as the top performers across both VIM and MPJPE metrics, demonstrating superior performance, particularly at short-term intervals. It was closely followed by the Future Motion model, which also performed well, though marginally less accurate in longer-term predictions than the GCN-Transformer or MPFSIR. The JRTransformer and SoMoFormer models also delivered competitive results, showing strength in predicting short to mid-term future poses but lagging slightly in long-term predictions. Notably, models like SocialTGCN and DViTA, while effective in mid-term forecasting, showed more significant performance degradation as the prediction horizon extended, highlighting their

Table 11: Evaluation results of multi-person pose forecasting models on the RI-HBS test dataset when pre-trained on the 3DPW and AMASS datasets. The table presents model performance at various time intervals using VIM and MPJPE metrics. GCN-Transformer leads in the overall ranking, with MPFSIR and Future Motion close behind [41].

Method	VIM						MPJPE					
	75ms	175ms	375ms	475ms	675ms	Overall	75ms	175ms	375ms	475ms	675ms	Overall
Zero Velocity	68.91	127.67	238.84	293.51	382.84	222.35	133.27	212.36	360.59	435.76	581.38	344.67
LTD [28]	46.11	90.26	178.25	223.02	300.20	167.57	87.30	144.15	255.39	313.85	431.48	246.44
MRT [52]	40.09	82.53	165.42	205.49	292.87	157.28	74.41	128.96	237.56	292.73	407.81	228.29
TBIformer [35]	41.81	81.85	154.54	187.42	245.39	142.20	78.22	129.46	225.61	272.18	363.06	213.70
DViTA [33]	37.85	80.51	156.37	191.77	251.66	143.63	67.42	120.32	222.83	273.44	369.93	210.79
JRTransformer [55]	29.74	72.82	152.65	189.85	262.80	141.57	50.77	104.17	212.17	264.53	368.22	199.97
SocialTGCN [37]	33.20	72.15	149.28	181.05	**232.54**	133.64	60.61	109.14	209.62	257.78	345.65	196.56
SoMoFormer [48]	27.71	65.76	142.36	179.64	253.39	133.77	48.36	95.02	193.19	243.78	346.59	185.39
Future Motion [51]	28.22	65.20	136.23	169.33	239.72	**127.74**	49.60	95.59	188.73	235.01	328.81	179.55
MPFSIR (our) [43]	28.62	66.33	**135.35**	**166.32**	259.55	131.24	50.04	94.71	186.88	**231.47**	**325.19**	177.66
GCN-Transformer (our) [45]	**26.15**	**64.07**	135.94	171.18	249.45	129.36	**44.30**	**90.60**	**186.54**	234.46	332.34	**177.65**

limitations in capturing the dynamics of long-term pose evolution.

Table 12: Evaluation results of multi-person pose forecasting models on the RI-HBS test dataset after fine-tuning on the RI-HBS training set for an additional 50 epochs. The table shows performance across multiple time intervals using VIM and MPJPE metrics. Significant improvements are observed across all models after fine-tuning, with GCN-Transformer retaining its top position and LTD showing a notable rise in performance, now ranking just behind SoMoFormer [41].

Method	VIM						MPJPE					
	75ms	175ms	375ms	475ms	675ms	Overall	75ms	175ms	375ms	475ms	675ms	Overall
Zero Velocity	68.91	127.67	238.84	293.51	382.84	222.35	133.27	212.36	360.59	435.76	581.38	344.67
DViTA [33]	37.31	76.43	138.81	167.55	217.20	127.46	68.16	116.27	203.72	244.76	323.45	191.27
TBIformer [35]	37.29	72.05	126.37	148.89	185.48	114.02	70.09	114.83	194.66	229.93	293.37	180.58
SocialTGCN [37]	33.41	68.88	123.90	148.17	186.50	112.17	61.66	106.83	185.64	222.11	288.52	172.95
JRTransformer [55]	28.76	68.50	125.15	148.69	194.49	113.12	49.13	98.16	181.45	217.98	288.36	167.01
Future Motion [51]	29.52	63.61	123.63	152.41	210.23	115.88	53.61	96.93	178.56	219.05	298.66	169.36
MRT [52]	34.65	68.22	115.83	135.10	167.65	104.29	63.99	108.21	180.22	210.96	267.67	166.21
MPFSIR (our) [43]	29.02	65.79	123.73	146.99	199.59	113.02	50.43	96.28	178.51	215.49	287.81	165.70
LTD [28]	33.29	66.19	115.46	132.53	160.87	101.67	61.89	104.01	174.28	205.02	258.65	160.77
SoMoFormer [48]	25.85	58.10	100.79	120.77	157.76	92.65	44.91	85.44	152.55	182.04	238.45	140.68
GCN-Transformer (our) [45]	**25.48**	**56.76**	**98.32**	**115.32**	**142.25**	**87.63**	**44.58**	**85.28**	**150.55**	**178.58**	**226.66**	**137.13**

Fine-tuning the models on the RI-HBS dataset resulted in substantial performance improvements across all the evaluated models, highlighting the importance of domain-specific training as shown in Table 12. Every model benefited from this process, showing reduced errors in both VIM and MPJPE metrics. The GCN-Transformer continued to excel, maintaining its position as the top performer. After fine-tuning, it further reduced its errors across all time intervals, particularly strengthening its long-term prediction capabilities. This solidified its status as the most reliable model for forecasting both short-term and long-term poses. SoMoFormer, which had already been a strong contender, showed even more impressive results after fine-tuning, significantly narrowing the gap with the GCN-Transformer. Its performance in short to mid-term predictions improved considerably. The JRTransformer also made notable strides, especially in short-term predictions, but still significantly lagging behind the top-performing models. One of the most striking changes after fine-tuning was observed in the LTD model. Initially, it had lagged behind many of the other models, but after fine-tuning, it made a dramatic leap in performance, positioning itself right behind SoMoFormer. Other models, such as MPFSIR, MRT, SocialTGCN, DViTA, and SoMoFormer, also showed improvements after fine-tuning, though they remained better suited for short to mid-term predictions. While they still trailed behind the top performers, their errors were reduced, and their long-term prediction capabilities saw some enhancements, albeit not as significant as those observed in the leading models.

3. CONCLUSION

In conclusion, this doctoral thesis has presented significant advancements in the field of multi-person pose forecasting by developing new models, metrics, and loss functions that improve the performance and generalizability of forecasting poses in complex dynamic environments. The research focused on addressing core challenges in the domain, particularly regarding computational efficiency, model performance, and comprehensive evaluation.

One of the key contributions of this thesis is the development of a lightweight neural network architecture named MPFSIR, which also includes a social interaction prediction component designed to model and predict interactions between individuals. This model is particularly suited for real-time applications, as it achieves comparable results to state-of-the-art methods while using up to 30 times fewer parameters. This efficiency is especially valuable when computational resources are limited or fast and reliable processing is necessary. It was demonstrated through rigorous evaluation that the model could maintain high performance, optimizing the balance between the number of model parameters and results.

Additionally, this thesis introduced the GCN-Transformer, a novel neural network architecture that combines Graph Convolutional Networks (GCN) and Transformer components. By effectively capturing both spatial and temporal dynamics, this hybrid model proved highly adaptable, outperforming existing models on four challenging datasets. Based on the MPJPE metric, the GCN-Transformer shows an average improvement of 4.15% over the closest state-of-the-art model across these datasets. More specifically, GCN-Transformer shows a 4.7% improvement over the closest SOTA model on CMU-Mocap, 4.3% improvement over the closest SOTA model on MuPoTS-3D, 5% improvement over the closest SOTA model on the SoMoF Benchmark, and a 2.6% improvement over the closest SOTA model on the ExPI dataset. Its ability to generalize across different data domains, unlike other models whose performance fluctuates across datasets, underscores the robustness of this architecture. GCN-Transformer generalization capability is further supported by the minimal variation in its improvement over the Zero Velocity baseline, with a standard deviation of only 1.69% in two-person scenes and just

0.1% in three-person scenes. This consistent performance demonstrates that the GCN-Transformer provides a foundation for application in diverse environments and domains, solidifying its position as the current state-of-the-art in multi-person pose forecasting.

The thesis also introduced an innovative loss function composed of two specific terms: Multi-person Joint Distance Loss (MPJD) and Velocity Loss (VL). VL captures movement velocities to enhance the temporal coherence of pose sequences, while MPJD measures joint distances between individuals to improve spatial interaction dependencies. This loss function significantly enhanced model performance by improving the training process, leading to better capturing human movement dynamics. An ablation study conducted as part of the research demonstrated a clear improvement in forecasting performance when the new loss terms were applied. The enhanced loss function resulted in a 4.8% improvement in the MPJPE metric and a 3.5% improvement in the VIM metric compared to the standard loss function, clearly validating its effectiveness. This improvement highlights the value of incorporating these additional movement dynamics into the training process, leading to more accurate and realistic pose forecasts.

Moreover, a new evaluation metric named FJPTE (Final Joint Position and Trajectory Error) has been introduced to provide a more detailed framework for assessing pose forecasting models. Unlike standard metrics, FJPTE evaluates both the full movement trajectory and the final joint position, providing deep insights into the models' effectiveness in capturing human dynamics by assessing both local movements and global positions. This nuanced approach to evaluation helps distinguish the subtle performance differences among leading models, providing a clearer picture of each model's strengths and weaknesses in various forecasting aspects.

Overall, this thesis has made significant contributions to the field of multi-person pose forecasting by tackling key challenges in model design, training strategies, and evaluation methods. The research has advanced theoretical understanding and paved the way for practical applications in scenarios that demand high performance and efficiency.

3.1. Validating hypotheses and scientific contributions

This doctoral thesis is grounded in a series of scientific contributions and hypotheses outlined in section 1.3. This section will revisit these contributions and hypotheses, explaining how each was implemented and experimentally validated. Through rigorous experimentation and analysis, these contributions and the underlying hypotheses were proven effective in addressing the challenges posed by multi-person pose forecasting, demonstrating their value within the broader field context.

Scientific hypotheses were:

- **H1:** A model utilizing spatial and temporal pose features can achieve equivalent multi-person pose forecasting performance as more complex SOTA models while using significantly fewer parameters.

  Hypothesis H1, which asserts that a model utilizing spatial and temporal pose features can achieve equivalent multi-person pose forecasting performance as more complex state-of-the-art (SOTA) models while using significantly fewer parameters, was validated through the development of the MPFSIR model as presented in the paper [43]. The MPFSIR model effectively leveraged spatial and temporal features of human poses to forecast multi-person movements with performance comparable to more complex architectures. Despite its lightweight design, the model optimized the Pareto front by balancing pose forecasting error and the number of model parameters, proving that reducing the number of model parameters without sacrificing performance is possible. Specifically, MPFSIR achieved results comparable to SOTA models while using up to 30 times fewer parameters, thereby confirming the H1 hypothesis.

- **H2:** Combining the architectures of graph convolutional networks and Transformers can create a model that has a lower error in multi-person pose forecasting compared to existing SOTA model architectures.

  Hypothesis H2 asserts that integrating graph convolutional networks (GCNs) with Transformer architectures can yield a model that surpasses existing state-of-the-art (SOTA) models in terms of error rates in multi-person pose forecasting. This hypothesis was validated by developing and implementing the GCN-Transformer model, detailed in the paper [45]. The GCN-Transformer model leverages the strengths of both GCNs and Transformers, capturing spatial relationships between joints and temporal dynamics of human motion with exceptional efficacy. This hybrid model provides a comprehensive understanding of pose sequences by incorporating the spatio-temporal processing capabilities of GCNs with Transformers's spatio-temporal and contextual proficiency. It demonstrated superior performance compared to existing SOTA models across diverse datasets, establishing it as the new benchmark in the field. On average, GCN-Transformer outperformed the closest SOTA model by 4.15% based on the MPJPE metric across four evaluated datasets. Moreover, its generalization capability was confirmed by low variability in performance, showing a standard deviation of only 1.69% in two-person scenes and just 0.1% in three-person scenes when measuring improvement compared to the Zero Velocity baseline. This model architecture validates the H2 hypothesis and significantly advances developing predictive models for multi-person pose forecasting, highlighting the potent synergies between GCNs and Transformers in this complex domain.

- **H3:** A loss function that includes movement velocity error and joint distance error between individuals contributes to the effective training of the model.

  Hypothesis H3 asserts that a loss function incorporating both movement velocity and joint distance errors between individuals enhances the effectiveness of model training for multi-person pose forecasting. This hypothesis was substantiated in the paper [45], which introduced a novel loss function that includes terms for movement velocities and spatial joint distances between individuals. By integrating these elements, the loss function ensures that the model predicts accurate positions and captures the intricate dynamics of motion and the interactions among individuals in a scene. The efficacy of this enhanced loss function was rigorously tested through an ablation study, demonstrating significant improvements in model performance metrics. Specifically, including these additional loss components led to a 4.8% reduction in the MPJPE and a 3.5% decrease in the VIM, confirming the H3 hypothesis.

The realized scientific contributions were:

- **A lightweight neural network architecture and model for multi-person pose forecasting based on spatial and temporal features.**

  The paper [43] introduces a significant contribution to multi-person pose forecasting with a lightweight neural network architecture called MPFSIR. This model leverages spatial and temporal features to predict future poses efficiently. The contribution was validated by demonstrating the models' ability to optimize the Pareto front, where it achieved a balance between forecasting error and the number of model parameters. The MPFSIR model delivered comparable performance to state-of-the-art models on evaluated datasets while utilizing up to 30 times fewer parameters. This optimization showed that the model maintained high forecasting performance and offered computational advantages, proving its value for practical applications requiring efficient performance.

- **A neural network architecture and model for multi-person pose forecasting comprising a graph convolutional network and a Transformer.**

  The paper [45] presented a novel neural network architecture for multi-person pose forecasting that integrates a Graph Convolutional Network (GCN) with a Transformer called GCN-Transformer. This hybrid approach effectively captures both the spatial relationships between joints and the temporal dynamics of human motion, making it a powerful tool for pose forecasting. The contribution was validated by demonstrating GCN-Transformer's superior performance over other state-of-the-art models on four distinct datasets. On average, GCN-Transformer improved the MPJPE metric by 4.15% compared to the closest state-of-the-art model across these datasets. The model's ability to generalize across different dataset domains, where competing models struggled to maintain consistent performance, proved its robustness and versatility. This generalizability is further evidenced by the low variability in its improvements over the Zero Velocity baseline, recording a standard deviation of just 1.69% in two-person scenes and 0.1% in three-person scenes. This adaptability across varied domains highlighted the strength of the GCN-Transformer architecture, suggesting that other models might be more specialized to specific types of domains, whereas GCN-Transformer architecture showed broad applicability and consistent ranking in both datasets.

- **A loss function for effective training of pose forecasting models that includes movement velocities and joint distance between individuals.**

  The paper [45] introduced a novel loss function designed to enhance the training of pose forecasting models. This loss function incorporates both movement velocities and joint distances between individuals, implemented through two terms: Velocity Loss (VL) and Multi-person Joint Distance Loss (MPJD). These terms ensure that the model predicts the correct positions and accurately captures motion dynamics and the interactions between multiple individuals. The effectiveness of this contribution was validated through an ablation study, where the inclusion of VL and MPJD resulted in a significant improvement in the model's performance. Specifically, the enhanced loss function resulted in a 4.8% improvement in the MPJPE metric and a 3.5% improvement in the VIM metric, demonstrating its superiority over the standard loss function. These results underscore the value of incorporating movement and interaction dynamics into the loss, leading to more accurate and realistic pose predictions.

- **An evaluation metric for pose forecasting that considers the movement trajectory and the final position.**

  The paper [45] proposed a novel evaluation metric for pose forecasting that considers both the movement trajectory and the final position of the predicted poses. This metric, referred to as Final Joint Position and Trajectory Error (FJPTE), provides a more nuanced and comprehensive assessment of model performance than standard metrics such as VIM and MPJPE. The value of this contribution was demonstrated by evaluating various models using the new metric. The FJPTE metric offers a breakdown of performance into two main components: $\text{FJPTE}_{\text{local}}$, which measures errors related to local movement dynamics, and $\text{FJPTE}_{\text{global}}$, which quantifies errors in global positioning and trajectory. This separation enables a more detailed comparison of models, providing specific and actionable insights into their strengths and weaknesses. Utilizing this metric revealed important distinctions between models that would have been obscured by standard aggregate metrics, proving it to be a valuable tool for evaluating pose forecasting models in a more fine-grained and meaningful manner.

3.2. Future research directions

As we look toward the future of multi-person pose forecasting, several avenues for research emerge from the findings and limitations of this thesis. One critical area involves modeling social interactions between individuals within a scene. Understanding and accurately predicting how people move in response to others requires a deeper integration of social dynamics into forecasting models. This could be achieved through advanced graph-based techniques or by incorporating social interaction knowledge directly into the neural network architectures.

Further advancements should enhance the models' capabilities to learn human movement dynamics. This could involve the development of more sophisticated loss functions that better capture the intricacies of human motion, improvements in data preprocessing techniques to enrich model inputs or the exploration of new model architectures that more accurately reflect human biomechanics. Each of these areas offers the potential for significant impacts on the realism and accuracy of pose forecasts.

Additionally, embedding knowledge about valid human poses directly into the models is crucial to avoid generating physically impossible poses. This could be addressed by integrating constraints into the learning process, possibly through generative models or by applying post-processing constraints that adjust the outputs to fall within humanly possible ranges. Such developments would improve the visual credibility of the models' outputs and enhance their overall performance.

Exploration into hybrid architectures that combine the strengths of various modeling approaches, such as those blending graph convolutional networks with Transformers, should continue. These architectures have shown promise in capturing both spatial and temporal relationships. Still, they may be further enhanced by integrating additional components like recurrent neural networks or capsule networks to better handle sequences and hierarchical relationships.

Lastly, improving the generalization capabilities of pose forecasting models across different types of data domains remains a significant challenge. Future research should focus on developing models that maintain high performance regardless of the dataset characteristics. This might involve training strategies that encourage robustness, such as domain adaptation techniques or multi-task learning frameworks that can handle a variety of human activities and interaction scenarios.

By addressing these future research directions, multi-person pose forecasting can continue to advance toward more accurate, efficient, and universally applicable models. These efforts will extend the applicability of pose forecasting technology to a broader range of real-world scenarios, further bridging the gap between theoretical research and practical applications.

4. ABSTRACTS OF ARTICLES

4.1. MPFSIR: An Effective Multi-Person Pose Forecasting Model With Social Interaction Recognition

In recent years, multi-person pose forecasting has gained significant attention due to its potential applications in various fields such as computer vision, robotics, sports analysis, and human-robot interaction. In this paper, we propose a novel deep learning model for multi-person pose forecasting called MPFSIR (multi-person pose forecasting and social interaction recognition) that achieves comparable results with state-of-the-art models, but with up to 30 times fewer parameters. In addition, the model includes a social interaction prediction component to model and predict interactions between individuals. We evaluate our model on three benchmark datasets: 3DPW, CMU-Mocap, and MuPoTS-3D, compare it with state-of-the-art methods, and provide an ablation study to analyze the impact of the different model components. Experimental results show the effectiveness of MPFSIR in accurately predicting future poses and capturing social interactions. Furthermore, we introduce the metric MW-MPJPE to evaluate the performance of pose forecasting, which focuses on motion dynamics. Overall, our results highlight the potential of MPFSIR for predicting the poses of multiple people and understanding social dynamics in complex scenes and in various practical applications, especially where computational resources are limited. The code is available at `https://github.com/RomeoSajina/MPFSIR`.

Available at:
`https://ieeexplore.ieee.org/document/10210381`

Romeo Šajina, Marina Ivašić-Kos, 2023. MPFSIR: An Effective Multi-Person Pose Forecasting Model With Social Interaction Recognition. IEEE Access, Volume 11, ISSN 2169-3536,
DOI: 10.1109/ACCESS.2023.3303018

4.2. GCN-Transformer: Graph Convolutional Network and Transformer for Multi-Person Pose Forecasting Using Sensor-Based Motion Data

Multi-person pose forecasting involves predicting the future body poses of multiple individuals over time, involving complex movement dynamics and interaction dependencies. Its relevance spans various fields, including computer vision, robotics, human–computer interaction, and surveillance. This task is particularly important in sensor-driven applications, where motion capture systems, including vision-based sensors and IMUs, provide crucial data for analyzing human movement. This paper introduces GCN-Transformer, a novel model for multi-person pose forecasting that leverages the integration of Graph Convolutional Network and Transformer architectures. We integrated novel loss terms during the training phase to enable the model to learn both interaction dependencies and the trajectories of multiple joints simultaneously. Additionally, we propose a novel pose forecasting evaluation metric called Final Joint Position and Trajectory Error (FJPTE), which assesses both local movement dynamics and global movement errors by considering the final position and the trajectory leading up to it, providing a more comprehensive assessment of movement dynamics. Our model uniquely integrates scene-level graph-based encoding and personalized attention-based decoding, introducing a novel architecture for multi-person pose forecasting that achieves state-of-the-art results across four datasets. The model is trained and evaluated on the CMU-Mocap, MuPoTS-3D, SoMoF Benchmark, and ExPI datasets, which are collected using sensor-based motion capture systems, ensuring its applicability in real-world scenarios. Comprehensive evaluations on the CMU-Mocap, MuPoTS-3D, SoMoF Benchmark, and ExPI datasets demonstrate that the proposed GCN-Transformer model consistently outperforms existing state-of-the-art (SOTA) models according to the VIM and MPJPE metrics. Specifically, based on the MPJPE metric, GCN-Transformer shows a 4.7% improvement over the closest SOTA model on CMU-Mocap, 4.3% improvement over the closest SOTA model on MuPoTS-3D, 5% improvement over the closest SOTA model on the SoMoF Benchmark, and a 2.6% improvement over the closest SOTA model on the ExPI dataset. Unlike other models with performances that fluctuate across datasets, GCN-Transformer performs consistently, proving its robustness in multi-person pose forecasting and providing an excellent foundation for the application of GCN-Transformer in different domains.

Available at:
`https://www.mdpi.com/1424-8220/25/10/3136`

Romeo Šajina, Goran Oreški, Marina Ivašić-Kos, 2025. GCN-Transformer: Graph Convolutional Network and Transformer for Multi-Person Pose Forecasting Using Sensor-Based Motion Data. Sensors, Volume 25, ISSN 1424-8220,
DOI: 10.3390/s25103136

4.3. 3D Pose Estimation and Tracking in Handball Actions Using a Monocular Camera

Player pose estimation is particularly important for sports because it provides more accurate monitoring of athlete movements and performance, recognition of player actions, analysis of techniques, and evaluation of action execution accuracy. All of these tasks are extremely demanding and challenging in sports that involve rapid movements of athletes with inconsistent speed and position changes, at varying distances from the camera with frequent occlusions, especially in team sports when there are more players on the field. A prerequisite for recognizing the player's actions on the video footage and comparing their poses during the execution of an action is the detection of the player's pose in each element of an action or technique. First, a 2D pose of the player is determined in each video frame, and converted into a 3D pose, then using the tracking method all the player poses are grouped into a sequence to construct a series of elements of a particular action. Considering that action recognition and comparison depend significantly on the accuracy of the methods used to estimate and track player pose in real-world conditions, the paper provides an overview and analysis of the methods that can be used for player pose estimation and tracking using a monocular camera, along with evaluation metrics on the example of handball scenarios. We have evaluated the applicability and robustness of 12 selected 2-stage deep learning methods for 3D pose estimation on a public and a custom dataset of handball jump shots for which they have not been trained and where never-before-seen poses may occur. Furthermore, this paper proposes methods for retargeting and smoothing the 3D sequence of poses that have experimentally shown a performance improvement for all tested models. Additionally, we evaluated the applicability and robustness of five state-of-the-art tracking methods on a public and a custom dataset of a handball training recorded with a monocular camera. The paper ends with a discussion apostrophizing the shortcomings of the pose estimation and tracking methods, reflected in the problems of locating key skeletal points and generating poses that do not follow possible human structures, which consequently reduces the overall accuracy of action recognition.

Available at:

https://www.mdpi.com/2313-433X/8/11/308

Romeo Šajina, Marina Ivašić-Kos, 2022. 3D Pose Estimation and Tracking in Handball Actions Using a Monocular Camera. Journal of Imaging, Volume 8, ISSN 2313-433X, DOI: 10.3390/jimaging8110308

4.4. Other research papers that are the result of the doctorate research

4.4.1. Analysis of Multi-Person Pose Forecasting Models on Handball Actions

Multi-person pose forecasting involves predicting the future poses of multiple individuals within a scene, which is crucial for various applications in sports analytics, surveillance, human-computer interaction, etc. This paper investigates multi-person pose forecasting models in the context of handball actions, presenting a comprehensive analysis through two main experiments. Firstly, we evaluate pre-trained models on a custom dataset of handball shots to assess their applicability in real-world scenarios. Secondly, we analyze model performance on the same handball shot dataset after fine-tuning, emphasizing domain adaptation effects. Additionally, we introduce a novel dataset for multi-person pose forecasting, featuring scenarios where two players execute handball shots. This dataset fills a critical gap by providing a specialized and dynamic environment for evaluating pose forecasting models. The experiments highlight the effectiveness of transfer learning and domain adaptation in enhancing model accuracy and robustness for real-world applications involving complex human interactions and movements.

The code and dataset are available at `https://github.com/RomeoSajina/MPF-HBS`.

Available at:
`https://ieeexplore.ieee.org/document/10638569/`

Romeo Šajina, Marina Ivašić-Kos, 2024. Analysis of Multi-Person Pose Forecasting Models on Handball Actions. 2024 8th International Conference on Computer, Software and Modeling (ICCSM), ISBN:979-8-3503-6714-0,
DOI: 10.1109/ICCSM63823.2024.00018

4.4.2. Evaluacija i analiza modela dubokih neuronskih mreža za predviđanje kretanja više osoba na sceni

Predviđanje kretanja više osoba na sceni predstavlja izazovan zadatak sa širokim potencijalom primjene. Ovaj problematičan aspekt pronalazi svoju primjenu u kontekstu autonomnih vozila, gdje se koristi za predviđanje kretanja pješaka i na temelju tih predviđanja se poduzimaju odgovarajuće akcije. Također, u sportskoj analizi, ovakvi modeli se primjenjuju za predviđanje kretanja igrača, dok se u području robotske mobilnosti koriste za anticipiranje ponašanja okoline i prilagođavanje ponašanja robota prema tim predviđanjima. Ovaj znanstveni rad temelji se na istraživanju modela za predviđanje kretanja više osoba na sceni, s posebnim fokusom na analizi njihovih performansi na novom, do sada neistraženom skupu podataka. U radu će se analizirati najnoviji modeli koji se većinski oslanjaju na arhitekturu Transformera, ali će se također obuhvatiti i pristupi temeljeni na jednostavnijim arhitekturama. Kroz ovu analizu, rad pridonosi dubljem razumijevanju raznolikosti modela za predviđanje kretanja više osoba, pružajući uvid u to kako najnovije arhitekture, poput Transformera, odgovaraju na ovaj problem u usporedbi s prethodnim pristupima. Istraživanje donosi doprinos razvoju tehnika predviđanja kretanja u realnom vremenu, s potencijalom za unapređenje autonomnih sustava u različitim okolinama i scenarijima primjene.

Available at:
`http://mipro-proceedings.com`

Romeo Šajina, 2024. Evaluacija i analiza modela dubokih neuronskih mreža za predviđanje kretanja više osoba na sceni. 2024 47th MIPRO ICT and Electronics Convention (MIPRO), The paper is not indexed because it was written in Croatian.

REFERENCES

[1] Vida Adeli et al. "Socially and contextually aware human motion and pose forecasting". In: *IEEE Robotics and Automation Letters* 5.4 (2020), pp. 6033–6040.

[2] Vida Adeli et al. "Tripod: Human trajectory and pose dynamics forecasting in the wild". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 13390–13400.

[3] A Balasundaram, S Ashok Kumar, and S Magesh Kumar. "Optical flow based object movement tracking". In: *Int. J. Eng. Adv. Technol.(IJERT)* 9 (2019), pp. 3913–3916.

[4] Qian Bao et al. "Pose-guided tracking-by-detection: Robust multi-person pose tracking". In: *IEEE Transactions on Multimedia* 23 (2020), pp. 161–175.

[5] Alex Bewley et al. "Simple online and realtime tracking". In: *2016 IEEE international conference on image processing (ICIP)*. IEEE. 2016, pp. 3464–3468.

[6] Arij Bouazizi et al. "MotionMixer: MLP-based 3D Human Body Pose Forecasting". In: *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22, Vienna, Austria, 23–29 July 2022*. July 2022, pp. 791–798.

[7] Zhe Cao et al. "Realtime multi-person 2d pose estimation using part affinity fields". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 7291–7299.

[8] *Carnegie Mellon University Motion Capture Database*. (accessed on 2 February 2025). URL: `https://paperswithcode.com/dataset/cmu-motion-capture`.

[9] Joao Carreira et al. "Human pose estimation with iterative error feedback". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 4733–4742.

[10] Yu Cheng et al. "Graph and Temporal Convolutional Networks for 3D Multi-person Pose Estimation in Monocular Videos". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 35.2 (May 2021), pp. 1157–1165.

[11] Hsu-kuang Chiu et al. "Action-agnostic human pose forecasting". In: *2019 IEEE winter conference on applications of computer vision (WACV)*. IEEE. 2019, pp. 1423–1432.

[12] Wen Guo et al. "Back to mlp: A simple baseline for human motion prediction". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2023, pp. 4809–4819.

[13] Wen Guo et al. "Multi-person extreme motion prediction". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 13053–13064.

[14] Kaiming He et al. "Mask r-cnn". In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2961–2969.

[15] Junjie Huang et al. "The devil is in the details: Delving into unbiased data processing for human pose estimation". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 5700–5709.

[16] Yingfan Huang et al. "Stgat: Modeling spatial-temporal interactions for human trajectory prediction". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 6272–6281.

[17] Eldar Insafutdinov et al. "Arttrack: Articulated multi-person tracking in the wild". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 6457–6465.

[18] Umar Iqbal, Anton Milan, and Juergen Gall. "Posetrack: Joint multi-person pose estimation and tracking". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 2011–2020.

[19] Kiran Kale, Sushant Pawar, and Pravin Dhulekar. "Moving object tracking using optical flow and motion vector estimation". In: *2015 4th international conference on reliability, infocom technologies and optimization (ICRITO)(trends and future directions)*. IEEE. 2015, pp. 1–6.

[20] Rudolph Emil Kalman. "A new approach to linear filtering and prediction problems". In: (1960).

[21] Shichao Li et al. "Cascaded Deep Monocular 3D Human Pose Estimation With Evolutionary Training Data". In: *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* June 2020.

[22] Sijin Li and Antoni B Chan. "3d human pose estimation from monocular images with deep convolutional neural network". In: *Asian Conference on Computer Vision.* Springer. 2014, pp. 332–347.

[23] Bruce D Lucas, Takeo Kanade, et al. "An iterative image registration technique with an application to stereo vision". In: Vancouver. 1981.

[24] Diogo C Luvizon, Hedi Tabia, and David Picard. "Human pose regression by combining indirect part detection and contextual information". In: *Computers & Graphics* 85 (2019), pp. 15–22.

[25] Naureen Mahmood et al. "AMASS: Archive of motion capture as surface shapes". In: *Proceedings of the IEEE/CVF international conference on computer vision.* 2019, pp. 5442–5451.

[26] Wei Mao, Miaomiao Liu, and Mathieu Salzmann. "History repeats itself: Human motion prediction via motion attention". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16.* Springer. 2020, pp. 474–489.

[27] Wei Mao, Miaomiao Liu, and Mathieu Salzmann. "History repeats itself: Human motion prediction via motion attention". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16.* Springer. 2020, pp. 474–489.

[28] Wei Mao et al. "Learning trajectory dependencies for human motion prediction". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision.* 2019, pp. 9489–9497.

[29] Julieta Martinez et al. "A simple yet effective baseline for 3d human pose estimation". In: *Proceedings of the IEEE International Conference on Computer Vision.* 2017, pp. 2640–2649.

[30] Omar Medjaouri and Kevin Desai. "Hr-stan: High-resolution spatio-temporal attention network for 3d human motion prediction". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 2540–2549.

[31] Dushyant Mehta et al. "Single-shot multi-person 3d pose estimation from monocular rgb". In: *2018 International Conference on 3D Vision (3DV)*. IEEE. 2018, pp. 120–130.

[32] Alejandro Newell, Kaiyu Yang, and Jia Deng. "Stacked hourglass networks for human pose estimation". In: *European conference on computer vision*. Springer. 2016, pp. 483–499.

[33] Behnam Parsaeifard et al. "Learning decoupled representations for human pose forecasting". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 2294–2303.

[34] Dario Pavllo et al. "3d human pose estimation in video with temporal convolutions and semi-supervised training". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 7753–7762.

[35] Xiaogang Peng, Siyuan Mao, and Zizhao Wu. "Trajectory-aware body interaction transformer for multi-person pose forecasting". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 17121–17130.

[36] Xiaogang Peng et al. "SoMoFormer: Social-Aware Motion Transformer for Multi-Person Motion Prediction". In: *arXiv preprint arXiv:2208.09224* (2022).

[37] Xiaogang Peng et al. "The MI-Motion Dataset and Benchmark for 3D Multi-Person Motion Prediction". In: *arXiv preprint arXiv:2306.13566* (2023).

[38] Leonid Pishchulin et al. "Deepcut: Joint subset partition and labeling for multi person pose estimation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 4929–4937.

[39] Mir Rayat Imtiaz Hossain and James J Little. "Exploiting temporal information for 3D pose estimation". In: *arXiv e-prints* (2017), arXiv–1711.

[40] Shaoqing Ren et al. "Faster r-cnn: Towards real-time object detection with region proposal networks". In: *arXiv preprint arXiv:1506.01497* (2015).

[41] Romeo Sajina and Marina Ivasic-Kos. “Analysis of Multi-Person Pose Forecasting Models on Handball Actions”. In: *2024 8th International Conference on Computer, Software and Modeling (ICCSM)*. 2024, pp. 57–62. DOI: 10.1109/ICCSM63823.2024.00018.

[42] Romeo Šajina. “Evaluacija i analiza modela dubokih neuronskih mreža za predviđanje kretanja više osoba na sceni”. In: *Proceedings of 47th ICT and Electronics Convention (MIPRO)*. 2024, pp. 212–217.

[43] Romeo Šajina and Marina Ivasic-Kos. “MPFSIR: An Effective Multi-Person Pose Forecasting Model With Social Interaction Recognition”. In: *IEEE Access* 11 (2023), pp. 84822–84833. DOI: 10.1109/ACCESS.2023.3303018.

[44] Romeo Šajina and Marina Ivašić-Kos. “3D Pose Estimation and Tracking in Handball Actions Using a Monocular Camera”. In: *Journal of Imaging* 8.11 (2022), p. 308. DOI: 10.3390/jimaging8110308.

[45] Romeo Šajina, Goran Oreški, and Marina Ivašić-Kos. “GCN-Transformer: Graph Convolutional Network and Transformer for Multi-Person Pose Forecasting Using Sensor-Based Motion Data”. In: *Sensors* 25.10 (2025). ISSN: 1424-8220. DOI: 10.3390/s25103136.

[46] Bugra Tekin et al. “Structured prediction of 3d human pose with deep neural networks”. In: *arXiv preprint arXiv:1605.05180* (2016).

[47] Alexander Toshev and Christian Szegedy. “Deeppose: Human pose estimation via deep neural networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, pp. 1653–1660.

[48] Edward Vendrow et al. “SoMoFormer: Multi-Person Pose Forecasting with Transformers”. In: *arXiv preprint arXiv:2208.14023* (2022).

[49] Paul Voigtlaender et al. “Mots: Multi-object tracking and segmentation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 7942–7951.

[50] Timo Von Marcard et al. “Recovering accurate 3d human pose in the wild using imus and a moving camera”. In: *Proceedings of the European conference on computer vision (ECCV)*. 2018, pp. 601–617.

[51] Chenxi Wang et al. “Simple baseline for single human motion forecasting”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 2260–2265.

[52] Jiashun Wang et al. “Multi-person 3D motion prediction with multi-range transformers”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 6036–6049.

[53] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. “Simple online and realtime tracking with a deep association metric”. In: *2017 IEEE international conference on image processing (ICIP)*. IEEE. 2017, pp. 3645–3649.

[54] Yuliang Xiu et al. “Pose flow: Efficient online pose tracking”. In: *arXiv preprint arXiv:1802.00977* (2018).

[55] Qingyao Xu et al. “Joint-Relation Transformer for Multi-Person Motion Prediction”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 9816–9826.

[56] Chongyang Zhong et al. “Spatio-temporal gating-adjacency GCN for human motion prediction”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 6447–6456.

LIST OF FIGURES

LIST OF TABLES

APPENDIX

Appendix A, B, and C comprise the three main published papers, while appendix D and E comprise other research papers that are the result of the doctorate research.

A. MPFSIR: An Effective Multi-Person Pose Forecasting Model With Social Interaction Recognition

This work was published as: Romeo Šajina, Marina Ivašić-Kos, 2023. MPFSIR: An Effective Multi-Person Pose Forecasting Model With Social Interaction Recognition. IEEE Access, Volume 11 ISSN 2169-3536, DOI: 10.1109/ACCESS.2023.3303018

https://ieeexplore.ieee.org/document/10210381

1. Introduction

Pose forecasting is a subfield of computer vision that aims to predict future joint positions of the human body based on a set of previous poses. It includes the prediction of joint positions as well as the orientation and movement of the body.

Using different deep learning methods such as methods using RNN, graph convolutional network, and attention, significant progress has been made in predicting human poses, which are usually associated with predicting human movement. It is mainly a task of predicting 3D human poses, which is defined in fixed time intervals that mimic a fixed camera recording speed. Most often, from the initial observation of a person, the 3D behavior of that person is predicted up to ≈1 second in the future or the long-term behavior of n (several) seconds in the future, which increases the complexity of the prediction because the movement can include changes in the speed and direction of movement as well as changes in the type of movement due to execution of some other action. Pose forecasting can be performed for one person or multiple people in the scene. Single-person pose forecasting aims to predict the future pose of one person, while multi-person pose forecasting aims to predict the future poses of more than one person in a scene while considering the social interaction between those people. Understanding the social interaction between people on the scene requires determining the relationship between individuals that is not necessarily determined only by their physical location on the scene, although this is often the case when people know each other and interact or talk to each other or walk in parallel. Likewise, there are not rare examples when people stand next to each other, for example, at a bus stop or walk next to each other on a promenade at some point, and they do not know each other and have no interaction. The difference between single-person and multi-person forecasting lies in the complexity of the problem since multi-person forecasting requires modeling the interactions and dependencies between people, their movements, and poses that affect the range of what social interaction is involved and whether there is social contact.

Figure 1 exemplifies the main objective of our research, showcasing the capabilities of our proposed model, named MPFSIR, which effectively leverages the historical information of previous poses to accurately forecast future poses, while also demonstrating its

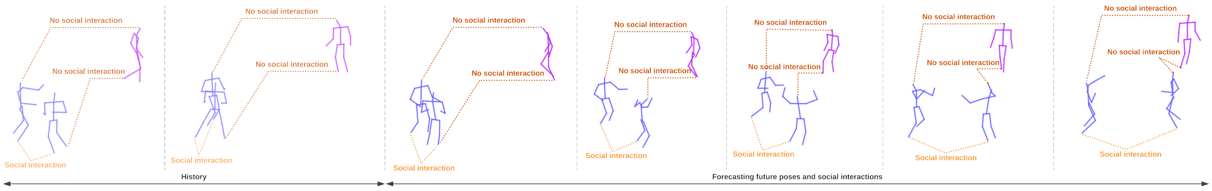


Figure 1: Illustration of the paper's objective to forecast future poses and predict the type of social interaction using historical pose information. The figure demonstrates the model's ability to leverage past poses to accurately predict future poses while considering the social interactions among individuals in the scene.

ability to predict the type of social interaction between individuals in the scene, thereby highlighting the comprehensive nature of our approach.

MPFSIR model leverages fully-connected layers with skip-connections to achieve state-of-the-art accuracy of pose forecasting with a small number of model parameters. Our model primarily focuses on utilizing the temporal information present in the data, allowing it to effectively capture the dynamics and temporal dependencies of human poses over time.

Predicting the poses of multiple individuals has applications in many fields, including robotics, human-computer interaction, and sports analysis. Several approaches have been proposed to solve this problem, including deep learning methods, graph-based models, and physics-based models. Despite significant progress, predicting the poses of multiple people remains a challenge with room for improvement, especially in scenarios with crowded scenes and complex social interactions. The paper presents our model called MPFSIR, which effectively uses historical information about past poses of people in a scene and prediction of the social interaction type to predict future poses of two or more people in a scene as accurately as possible, highlighting the comprehensive nature of our approach. Figure 1 shows the main goal of our model, which goes beyond traditional pose forecasting by considering the social aspect of human interactions and focuses on using the temporal information available in the data to effectively model the dynamics and temporal dependencies of human poses over time.

The MPFSIR model uses fully connected layers with skip-connections and achieves prediction accuracy in the range of the most modern models but with a significantly smaller number of parameters, Figure 2. Our model provides additional information regarding the type of social interaction between people in the scene and thus, especially

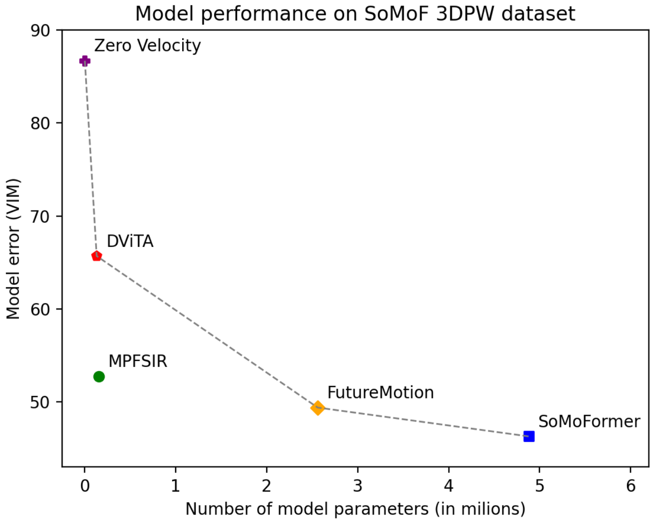


Figure 2: Comparison of model performance and parameter efficiency. The figure showcases the Pareto front, highlighting how our MPFSIR model achieves competitive results with a significantly reduced number of model parameters.

in the case of two or multiple rooms on the scene, gives an additional possibility of a deeper understanding of the context and dynamics of the observed human poses and offers valuable insights into social relations and interactions between people. In short, our contributions are:

- a new model for multi-person pose forecasting, with only 0.15 million model parameters
- a module for recognizing the type of social interaction between people in the scene and improving the pose forecasting regarding the interaction
- a new evaluation metric for pose forecasting that considers the person's overall dynamics and movements.

This paper is organized into several sections starting with the *Related work* section that describes previous research efforts related to pose forecasting and social interaction modeling. Then the *Pose Forecasting* section provides the basic concepts of predicting future poses from historical data and discusses the importance of temporal dependencies. In the *Proposed Model* section, our new pose forecasting approach is presented, highlighting the special features and fundamental principles of the proposed MPFSIR model. In the *Experimental Results*, the performance of the proposed model is compared with state-of-the-art methods on multiple datasets (SoMoF 3DPW, CMU-Mocap, MuPoTS-3D) along with the ablation study to examine different model components and the key factors contributing to model success. Additionally, a novel pose forecasting evaluation metric *MW-MPJPE* is described in the dedicated section with advantages over existing metrics in assessing the accuracy of pose forecasting. The paper ends with *Conclusion* that addresses the limitations of the proposed model and highlights areas for future improvement and potential directions for further research.

2. Related work

2.1. Pose forecasting

In single-person pose forecasting, models predict future joint coordinates without global translation. Recent models utilize RNN backbones, with some utilizing graph attention networks or GANs to extend prediction to multiple entities or provide plausible outputs. For example, Chiu et al. [4] use a hierarchical RNN to predict human motion, and Mao, Liu, and Salzmann in [9] models human motion as a graph of encoded motions for each joint coordinate with a GNN architecture to pass information between nodes. Attention-based models, such as the one introduced in [11] by Mao et al., capture similarities between current and historical motion sub-sequences to aggregate past motions for long-term forecasting. Guo et al. in [5] simplified the task of pose forecasting and proposed a simple MLP network with skip connections that models future poses using temporal information to predict the residual displacement of joints. They showed that using only 0.14 million parameters with this arrangement can perform better than the state-of-the-art models, which contain 20x to 30x more parameters. Recent papers in the field of pose forecasting have placed emphasis on separately modeling the spatial and temporal information of poses. Medjaouri and Desai in [13] proposed a model called HR-STAN (High-Resolution Spatio-Temporal Attention Network), that adopts a unique architecture that directly maps a fixed-length pose history to a fixed-length pose forecasting sequence, eliminating the need for a separate pose encoding and decoding step. HR-STAN decomposes convolutions into spatial and temporal components, allowing for more efficient modeling of spatio-temporal relationships within the pose history. Instead of using strided convolutions, dilated convolutions are employed in subsequent branches of the network to increase the receptive field without feature compression. Additionally, split spatial and temporal attention mechanisms are introduced to encourage the network to focus on the spatio-temporal relationships of specific motions. Zhong et al. in [22] proposed Gating-Adjacency GCN (GAGCN), a model that consists of an encoder-decoder structure, where GAGCN serves as the encoder and Temporal Convolutional Networks (TCN) act as the decoder. The encoder focuses on learning the spatio-temporal dependencies of the historical motion sequence. It utilizes a spatial and temporal gating network to derive blending coefficients that determine the importance of spatial and temporal information. These coefficients are then used to blend the spatial and temporal adjacency matrices, resulting in an adaptive adjacency matrix that captures the cross spatio-temporal dependencies. The fused spatial and temporal dependencies are obtained through the Kronecker product and are passed to the next layer. The decoder, utilizing TCN, takes the latent motion representation obtained from the encoder and predicts the future pose sequence.

For multi-person settings, such as those encountered in sports and social gathering environments, global interactions between people must be considered. Recent works in multi-person pose forecasting have therefore focused on predicting the future for an entire scene, using attention-based methods similar to those employed in pose and trajectory forecasting. However, current methods still have limitations in making predictions for multiple individuals. For instance, Adeli et al. [2] use graph attentional networks to model interactions between humans and objects but only use an RNN to predict future motion. Mart'inez-Gonz'alez, Villamizar, and Odobez in [12] use a transformer to predict an entire future sequence without recurrence but only consider one person at a time. Wang et al. [21] uses a transformer-based architecture to model global interactions between multiple individuals but can only make inferences for one person at a time. Vendrow et al. in [18] proposed a model called SoMoFormer, that solves the problem of inferencing only one person at a time by proposing a transformer architecture that models human motion input as a joint sequence rather than a time sequence, allowing them to perform attention over joints while predicting an entire future motion sequence for each joint in parallel.

Guo et al. in [6] proposed a multi-person pose forecasting model that consists of two parallel pipelines for the leader and the follower individuals. Each pipeline includes an attention model for temporal attention and a Graph Convolutional Network (GCN)-based predictor for spatial attention. These single-person motion forecasting mappings aim to learn representations for motion forecasting based on past motions and joint relationships. The model normalizes the raw poses by removing global displacement, and by normalizing the poses and considering the relative positions of the two individuals, the model aims to predict both distinct poses and their relative positions.

Peng et al. in [16] proposed a model called SoMoFormer, that aims to forecast the poses of multiple individuals by effectively capturing both local and global pose dynamics. It consists of three main components: the displacement sub-sequence encoder (DSE), the social interaction encoder (SIE), and the Transformer predictor. The DSE utilizes multiple Graph Convolutional Network (GCN) units to extract features from sub-sequences in the displacement trajectory space. They divide sequences into subsequences because humans tend to repeat their motion across a period of time, and dividing displacement sequences into sub-sequences can boost performance. Finally, the Transformer predictor employs multi-head attention to consider the relations between current and historical context across individuals and generates future motion trajectories for each individual through fully connected layers.

2.2. Social interaction

The SocialPool [1] layer models social interactions between people in the scene based on the distance between them, not taking into account that sometimes people can be spatially close without any social interaction. The SocialPool layer aggregates information from neighboring individuals in a scene and passes it through a pooling operation. This pooling operation can be average, max, or sum pooling, and the resulting pooled feature maps are concatenated with the individual features before being passed to the next layer. The SoMoFormer [18] models social connections between individuals by adopting a grid positioning method. This involves dividing the overall scene into a grid of cells and assigning each cell a learnable positional embedding. To associate individuals with specific cells, the neck joint position of each person at the last known frame is used to determine their corresponding cell. By incorporating the grid embedding and leveraging the distance between people in the scene, the model learns to capture and represent social relationships. This approach enables SoMoFormer to effectively model social connections and encode the dependencies between individuals. However, similarly to SocialPool [1], it does not consider that sometimes people can be spatially close without any social interaction.

Guo et al. in [6] propose a module called Cross-Interaction Attention (XIA) to model social interaction between dancers. XIA aims to share motion information between two predictors, a follower, and a leader. They introduce a cross-interaction attention module that takes one person's pose information (key-value pairs) and uses multi-head self-attention to refine the pose information for better motion forecasting. The module updates the keys and values using the Multi-Head Self-Attention module, followed by fully connected layers, and it is integrated at multiple stages of the computing flow. The refined keys and values are used in the collaborative human motion forecasting task to exploit information and jointly predict each person's motion.

Peng et al. in [16] propose a social interaction encoder (SIE) based on the Transformer model. The SIE consists of three components: a time encoder that calculates timestamp features, a spatial encoder that encodes multi-person displacement sequences, and social-aware motion attention. It simultaneously models individual motion and social interactions by capturing past displacements, preserving temporal information, representing spatial relations, and utilizing a social-aware attention mechanism. The SIE aims to improve multi-person motion forecasting by effectively incorporating social dynamics into the model.

On the other hand, our approach is to concatenate the pose sequences of two individuals, pass them through an auxiliary social interaction module $SCINT$ and classify the type of social interaction. This approach does not involve any pooling operation and focuses solely on modeling the social interactions between two individuals. This provides a more fine-grained analysis of the social interactions between individuals by specifically classifying the type of social interaction. However, it may not capture the social interactions between multiple individuals in the scene in a single step, which the SocialPool [1] layer, and SIE [16] encoder are designed to handle.

3. Proposed multi-person pose forecasting model with social interaction recognition

The goal is to predict the future motion of N individuals in a given scene. Each individual is represented by J joints, which are anatomical points on the body, such as elbows, knees, and shoulders. The model needs to predict the motion of these joints for T timesteps into the future. To do so, the model is given a sequence of historical poses for each individual in the scene. The historical poses are represented by the three-dimensional Cartesian coordinates of each joint in global coordinates. Each historical pose for individual n is represented by a J-dimensional vector x_k^n, where k is the time step, with the sequence of historical poses for individual n given by $X_{1:t}^n$.

The input pose sequence length, denoted as t, is the number of historical poses that the model receives as input. The range of values for n is from 1 to N, where N is the total number of individuals in the scene. The model's objective is to predict the future pose sequence for each individual, denoted as $S_{t+1:T}^n$. Here, T is the total number of timesteps into the future that the model is required to predict. The pose sequence $S_{t+1:T}^n$ represents the predicted poses of individual n for $T - t$ timesteps into the future.

3.1. Proposed model architecture

Our model is built with two temporal modules, one temporal context module, one spacial context module, and a social interaction auxiliary module as shown in Figure 3. Two sequences s_1 and s_2 are separately run through a preprocessing step that involves padding pose sequences with the last pose to the full sequence length (i.e. input size + output size) and applying a Discrete Cosine Transform (DCT) to encode human motion into the frequency domain, following the approach adopted by models such as SoMoFormer [18] and LTD [11]. Then, the two sequences s_1 and s_2 are separately fed into temporal module T_1 to capture temporal information of the sequences s_1 and s_2. Output sequences of T_1 are fed into the temporal context module $TCTX$ to capture temporal information between the sequences $T_1(s_1)$ and $T_1(s_2)$. Output sequences of $TCTX$ are then fed

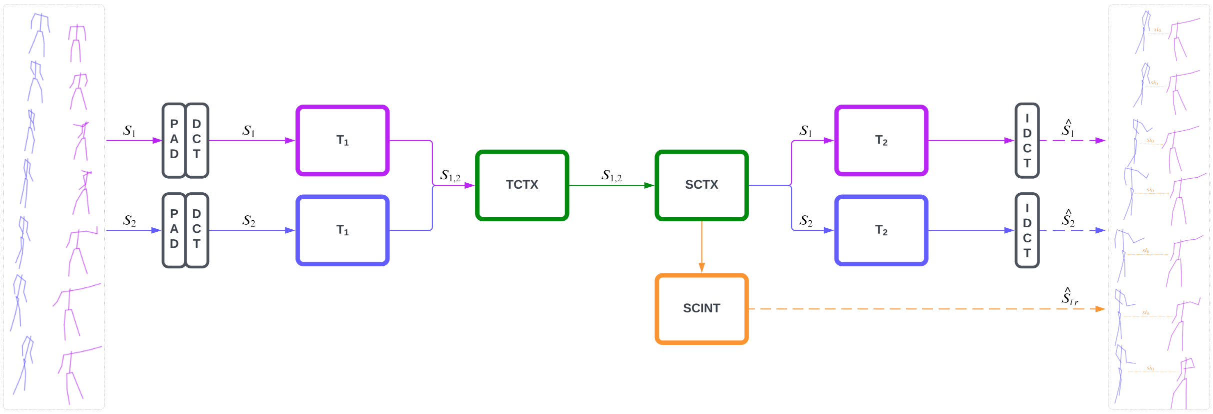


Figure 3: The figure illustrates the architectural components of the MPFSIR model. In the preprocessing step, the input sequences S_1 and S_2 are padded with the last pose to match the full length of the sequence. The padded sequences undergo a Discrete Cosine Transform (DCT) to convert them into the frequency domain representation. The transformed sequences are then fed into the model, which consists of various modules for pose forecasting and social interaction prediction. After passing through the model, the sequences are transformed back to Cartesian coordinates using the Inverse DCT (IDCT) to obtain the predicted poses. The model also classifies the type of social interaction between individuals in the scene.

into spacial context module $SPCTX$ to capture spatial information between sequences $TCTX_{1,2}(T_1(s_1), T_1(s_2))$. Output sequences of $SPCTX$ are parallelly fed into social interaction auxiliary module $SCINT$ to predict the type of social interaction between the sequences $SPCTX_{1,2}(TCTX_{1,2}(T_1(s_1), T_1(s_2)))$, and temporal module T_2 to refine the prediction of the sequences along the temporal dimension. Finally, an Inverse DCT is applied to the output sequences to transform the frequency domain back to Cartesian coordinates.

3.2. Modules of the MPFSIR model

In order to improve pose forecasting when there is social interaction between people, we propose a network that uses the temporal information of two pose sequences and calculates the temporal context together with the spatial context. We followed up on the research by Guo et al. in [5] on exploring the temporal dimension of the pose sequence and built a network consisting of the following modules: Temporal, Temporal context, Spatial context, and Social interaction auxiliary module. Each module of the MPFSIR model architecture consists of a fully connected layer followed by layer normalization and regularization dropout, including skip-connections connecting the outputs of different layers, allowing information to flow through the network and facilitating improved gradient propagation during training. An example module is shown in Figure 4.

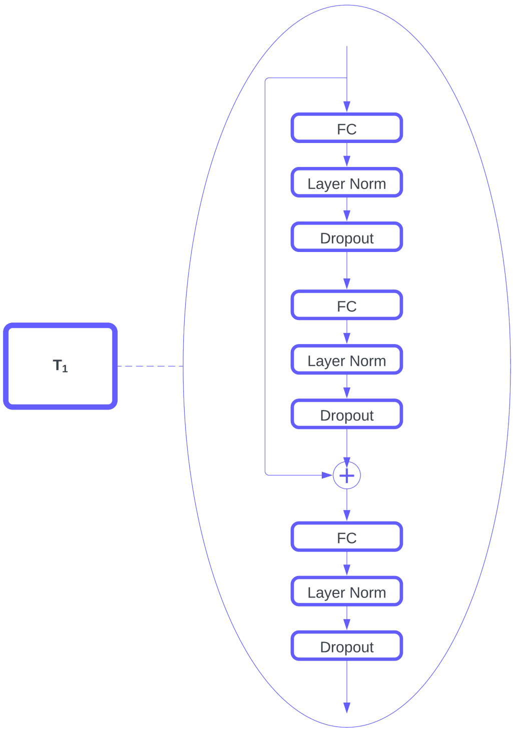


Figure 4: An example of the module architecture in the MPFSIR model. The module incorporates fully-connected layers, layer normalization, dropout, and skip connections, enabling effective information flow and facilitating enhanced training dynamics.

The temporal module (T) is designed as a sequence of three fully connected layers, where each layer is complemented with a Parametric Rectified Linear Unit (PReLU) activation function, layer normalization, and dropout (with a rate of 0.1). The first layer takes the input with a sequence length SL and expands it to a higher dimensionality of 512. Subsequently, the middle layer compresses the dimension back to the original sequence length SL. The last layer maintains the input and output dimensions. Finally, a skip connection is established, connecting the initial part of the module to the last layer to ensure better information flow and preserve crucial details.

The temporal context module (TCTX) captures temporal social information from the two pose sequences by concatenating the sequences along the temporal dimensions and processing the resulting joined sequences throughout the module. The module consists of two fully connected layers, where each layer is complemented with a PReLU activation function, layer normalization, and dropout (with a rate of 0.1). Specifically, the first fully connected layer takes the input with a dimensionality of joined sequences length $2SL$ and expands it to a higher dimensionality of $2.5 \times 2SL$. Subsequently, the second layer reduces the dimensionality to the dimension of $2SL$ while preserving important information. Notably, a skip connection is established within the module, connecting the initial part of the module to the last layer. This connection helps in better information propagation and enables the model to effectively capture relevant temporal dependencies between the two pose sequences.

The spacial context module (SCTX) is designed to capture spacial social information from the two pose sequences by concatenating the sequences along the spacial dimensions and processing the resulting joined sequences throughout the module. The module comprises two fully connected layers, each accompanied by a PReLU activation function, layer normalization, and dropout (with a rate of 0.1). Furthermore, the first fully connected layer takes the input with a dimensionality of joined sequences keypoints size $2KPS$ and expands it to a higher dimensionality of $2.5 \times 2KPS$. This expansion allows the module to capture more intricate spacial dependencies between the two pose sequences. Subsequently, the second layer reduces the dimensionality to the dimension of $2KPS$ while preserving relevant spacial information. Notably, a skip connection is established within the module, connecting the initial part of the module to the last layer. This connection facilitates better information propagation and enables the model to effectively capture spacial interactions between individuals in the scene.

The social interaction auxiliary module (SCINT) serves to classify the relationships between the two pose sequences. Initially, the sequences are joined along a new axis before being processed within the module. The first fully connected layer takes the joined sequences as input, where the dimensionality is calculated as the product of sequence length SL and sequence keypoints size KPS. The output dimensionality of this layer is set to sequence length SL. A PReLU activation function is then applied, followed by a Dropout (with a rate of 0.1). Subsequently, the sequences are flattened into a single vector and fed into the final fully connected layer, which performs the classification task. The classification outcome can assume one of three possibilities: socially dependent, socially independent, or sequences referring to the same person. This auxiliary module significantly facilitates the model's understanding of social interactions between individuals in the scene, enabling it to make more informed and accurate predictions.

3.3. Data transformation

We use a Discrete Cosine Transform (DCT) to encode human motion into the frequency domain as a collection of coefficients rather than directly predicting Cartesian coordinates. Similar approaches were reported in [18, 10, 21] where DCT transformations showed significant improvements in model performance and enabled simultaneous prediction of the entire trajectory for all future poses. The output of the model is transformed back to Cartesian coordinates using an inverse DCT (IDCT).

For a motion sequence of a single coordinate $(x_1, ..., x_T)$, the l-th DCT coefficient is computed by:

$$C_l = \sqrt{\frac{2}{T}} \sum_{t=1}^{T} \frac{x_t}{\sqrt{1+\delta_{l1}}} \cos \frac{\pi}{2T}(2t-1)(l-1) \tag{1}$$

where δ_{lj} denotes the *Kronecker* delta function with:

$$\delta_{lj} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j. \end{cases} \tag{2}$$

resulting in a coefficient sequence $(C_1, ..., C_T)$ with T coefficients, where $l = 1, 2, ..., T$. These coefficients serve as input tokens for the model to predict the completed trajectory $(\tilde{C}_1, ..., \tilde{C}_T)$. Using inverse DCT we can recover the coordinates:

$$\tilde{x}_t = \sqrt{\frac{2}{T}} \sum_{l=1}^{T} \frac{\tilde{C}_l}{\sqrt{1+\delta_{l1}}} \cos \frac{\pi}{2T}(2t-1)(l-1) \tag{3}$$

3.4. Data augmentation

Data augmentation is a crucial technique in the field of multi-person pose forecasting aimed at enhancing the performance and generalization capabilities of the models. With the limited availability of labeled training data, data augmentation methods provide a means to artificially expand the dataset by introducing diverse variations. In the context of multi-person pose forecasting, several effective methods for data augmentation have been devised. One commonly used technique is sequence reversal, which involves reversing the temporal order of the input sequences. This helps the model capture the temporal dynamics from both forward and backward perspectives, enabling it to better understand the progression of poses over time which can improve the accuracy and robustness of the predictions. Another method is random orientation, where the poses are randomly rotated to account for different camera viewpoints or human orientations. Random positioning introduces spatial variability by randomly shifting the positions of individuals within the scene, which can help the model learn to handle variations in the position of individuals in the scene. Random scaling is also applied to introduce variations in the scale of the poses, simulating different distances between the individuals and the camera. Additionally, random person permutation is employed, shuffling the order of individuals in a scene to account for different person arrangements. It can also help the model learn to handle different social interactions or group dynamics among individuals. These augmentation methods allow the model to learn from a more diverse range of scenarios, helping it to generalize better to unseen data and handle various challenges such as occlusions, varying body shapes, and complex interactions between individuals. By incorporating these data augmentation techniques, multi-person pose forecasting models can improve their performance and robustness in real-world settings.

3.4.1. Augmentation algorithm

This section describes the augmentation algorithm used during the model's training. The algorithm first checks whether data augmentation will be performed based on a random probability with a threshold of 0.5. If the probability is higher than 0.5, the algorithm returns the input sequences without performing any augmentation. Otherwise, the algorithm randomly performs the following augmentations:

- Backward movement: the function randomly chooses whether to flip the input sequences along the time dimension with a probability of 0.5.
- Reversing the order of people: the function randomly chooses whether to swap the input sequences with a probability of 0.5.
- Random scaling: the function randomly scales the input sequences by a random factor sampled from a uniform distribution between 0.1 and 5, with a probability of 0.5.
- Random rotation: the function randomly rotates the input sequences along the y-axis, x-axis, and z-axis, with a probability of 0.25 for each axis.
- Random repositioning: the function randomly repositions the input sequences in the space with a radius of 3 (radius is determined as the person's torso size), with a probability of 0.5.

The algorithm returns the augmented sequences, as shown in algorithm 1.

Algorithm 1: An algorithm for Data Augmentation

```
Require: $seq_0, seq_1$
Ensure: Augmented sequences $seq_0, seq_1$
if $rand() > 0.5$ then
    return $seq_0, seq_1$;
if $rand() > 0.5$ then
    $seq_0 \leftarrow$ flipBackwards($seq_0$);
    $seq_1 \leftarrow$ flipBackwards($seq_1$);
if $rand() > 0.5$ then
    $seq_0, seq_1 \leftarrow seq_1, seq_0$;
if $rand() > 0.5$ then
    $seq_0 \leftarrow$ RandomlyScale($seq_0, r1 = 0.1, r2 = 5$);
    $seq_1 \leftarrow$ RandomlyScale($seq_1, r1 = 0.1, r2 = 5$);
if $rand() > 0.75$ then
    $seq_0, seq_1 \leftarrow$ RandRotSeqs($seq_0, seq_1, \text{axis} = x$);
if $rand() > 0.75$ then
    $seq_0, seq_1 \leftarrow$ RandRotSeqs($seq_0, seq_1, \text{axis} = y$);
if $rand() > 0.75$ then
    $seq_0, seq_1 \leftarrow$ RandRotSeqs($seq_0, seq_1, \text{axis} = z$);
if $rand() > 0.5$ then
    $seq_0, seq_1 \leftarrow$ RandReposSeqs($seq_0, seq_1, rs = 3$);
return $seq_0, seq_1$
```

3.5. Data

We used different datasets for training and evaluation of our model to keep the same conditions as SoMoFormer [18] and MRT [21] models had: 3DPW [19] and AMASS [8] datasets for training the model, and SoMoF, CMU-Mocap [3], and MuPoTS-3D [14] dataset for evaluation.

The 3D Poses in the Wild (3DPW) [19] dataset includes over 60 video sequences of human motion in real-world settings. To evaluate our model using the SoMoF benchmark, we utilized the SoMoF benchmark splits for 3DPW, in which the 3DPW train and test set are flipped. Thus, we trained our model using the 3DPW test set and evaluated it on the 3DPW train set.

The Archive of Motion Capture As Surface Shapes (AMASS) [8] dataset provides a large dataset of human motion capture sequences, with over 40 hours of motion and 11,000 motions provided as SMPL mesh models. During training, we utilized the CMU, BMLMovi, and BMLRub subsets of this dataset, which provided a large-scale and varied set of motions. As many of these sequences are single-person, we synthesized additional training data by mixing sampled sequences to create multi-person training data.

For Carnegie Mellon University Motion Capture Database (CMU-Mocap) [3], we used the training and testing sets derived by Wang et al. in [21] to train and evaluate our model. Finally, the Multi-person Pose estimation Test Set in 3D (MuPoTS-3D) [14] dataset provides 8,000 annotated frames of poses from 20 real-world scenes. We used this dataset to evaluate the performance of our model.

3.6. Training

We train our model by taking the loss between the output and the ground truth and adding a social interaction loss multiplied by a γ factor to jointly learn future poses and the type of social interaction rather than focusing too much on each task.

We use L_2-norm loss to minimize the error between the ground truth and predicted coordinates and cross-entropy loss to minimize social interaction classification error.

$$\mathcal{L} = \mathcal{L}_{rec} + \mathcal{L}_{scir} \times \gamma \tag{4}$$

$$\mathcal{L}_{rec} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{5}$$

$$\mathcal{L}_{scir} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{C} y_{i,j}\log(\hat{y}_{i,j}) \tag{6}$$

We train our model for 500 epochs with a batch size of 256. We use the Adam optimizer with an initial learning rate of 0.01, decayed by 0.1 at epochs 10, 200, and 400. Finally, we set the γ parameter to value 0.01 while calculating the loss.

4. Experimental results

We evaluated the performance of our MPFSIR model on benchmark datasets, including SoMoF, CMU-Mocap, and MuPoTS-3D. To assess the accuracy of our predictions, we defined evaluation metrics that measure spatial and temporal alignment with the ground truth. We gain insights into our model's strengths, limitations, and generalization capabilities by analyzing the experimental results. This evaluation validates the effectiveness of our model for real-world applications in multi-person pose forecasting.

4.1. Metrics

MPJPE (Mean Per Joint Position Error) is a commonly used metric for evaluating the accuracy of pose forecasting methods [10, 12, 21, 18]. It measures the average Euclidean distance between the predicted joint positions and the corresponding ground truth positions across all joints. The lower the MPJPE value, the closer the predicted poses align with the ground truth. This metric provides a joint-level assessment of pose estimation performance. The MPJPE metric is calculated as follows:

$$E_{MPJPE}(y,\varphi) = \frac{1}{N_\varphi} \sum_{i=1}^{N_\varphi} \left\| P_{y,\varphi}^{(f)}(i) - P_{gt,\varphi}^{(f)}(i) \right\|_2 \tag{7}$$

where f denotes a time step and φ denotes the corresponding skeleton. $P_{y,\varphi}^{(f)}(i)$ is the estimated position of joint i and $P_{gt,\varphi}^{(f)}(i)$ is the corresponding ground truth position. N_φ represents the number of joints.

Another popular metric is VIM (Visibility-Ignored Metric), first introduced in [2], which is calculated by taking the mean distance between the ground truth and predicted joint positions. To compute this distance, the joint and coordinate dimensions are flattened together, resulting in a single vector representation for both the ground truth and predicted joint positions. The dimensionality of this vector would be 3J, where J represents the number of joints. Once the joint positions are flattened, the Euclidean distance (L2 norm) is computed between each corresponding pair of ground-truth and predicted joint positions. The distances are averaged across all joints to obtain the final VIM score. The SoMoF Benchmark uses this metric for evaluation. The VIM metric is calculated as follows:

$$E_{VIM}(y,\varphi) = \frac{1}{3\mathrm{J}_\varphi} \sum_{i=1}^{3\mathrm{J}_\varphi} \left\| P_{\mathrm{gt},\varphi}^{(i)} - P_{\mathrm{y},\varphi}^{(i)} \right\|_2 \tag{8}$$

where J represents the number of joints, $P_{\mathrm{gt},\varphi}^{(i)}$ is the ground-truth position of the i-th joint (flattened), $P_{\mathrm{y},\varphi}^{(i)}$ is the predicted position of the i-th joint (flattened), $\|\cdot\|_2$ denotes the Euclidean distance (L2 norm), and $\frac{1}{3\mathrm{J}_\varphi}\sum_{i=1}^{3\mathrm{J}_\varphi}$ represents the mean across all joints.

4.2. Results on SoMoF Benchmark

The benchmark provided by SoMoF [1, 2] is aimed at evaluating the performance of multi-person human pose forecasting methods. The benchmark involves predicting the next 14 frames (930 ms) using 16 frames (1070 ms) of input data. The input data includes joint positions for multiple people, and the results are reported as the mean VIM at multiple future time steps. Just as [20] and [18], we use the 3DPW [19] and AMASS [8] datasets for training, as they provide both multi-person and single-person data. During training, we only use the 13 joints evaluated in SoMoF.

Furthermore, we annotated the examples with two people in the scene with a label si_0 describing the existence of social interaction between the two people in the scene. Additionally, we use the examples from the 3DPW dataset with a single person in the scene and scenes from AMASS dataset for sampling two-person scenes with people without the social interaction si_1, and also to sample scenes where only one person is present si_2.

We compare different methods on the SoMoF 3DPW test set in Table 1 and show that our models consistently achieve comparative results with the competing methods with significantly fewer model parameters.

Table 1: Comparative analysis of model performance on the SoMoF Benchmark test set using the VIM metric. Our proposed model, MPFSIR, achieves comparable results to the state-of-the-art methods. The table presents results from the official dataset page somof.stanford.edu, where lower VIM values indicate higher accuracy in joint position predictions.

Method	3DPW Prediction in Time						Size
	100ms	240ms	500ms	640ms	900ms	Overall	#Param (M)
Mo-Att [10] + ST-GAT [7]	62.1	97.7	155.2	185.0	251.0	150.2	NA
SC-MPF [1]	46.3	73.9	130.2	160.8	208.4	123.9	15.65
Zero Velocity	29.4	53.6	94.5	112.7	143.1	86.7	0
TRiPOD [2]	30.3	51.8	85.1	104.8	146.3	83.7	NA
DViTA [15]	19.5	36.9	68.3	85.5	118.2	65.7	0.13
FutureMotion [20]	9.5	22.9	50.9	66.2	97.4	49.4	2.56
SoMoFormer [18]	**9.1**	**21.3**	**47.5**	**61.6**	**91.9**	**46.3**	4.88
MPFSIR	11.5	25.5	54.7	70.6	101.5	52.76	**0.15**

4.3. Results on CMU-Mocap and MuPoTS-3D

In our study, we also compare our proposed method with the recent multi-person pose forecasting approach by Vendrow et al. [18], which has achieved state-of-the-art results on several datasets. Additionally, we compare our method to other contemporary techniques HRI [10], LTD [11], and MRT [21]. In line with their protocols, we train the models using a synthesized dataset created by combining sampled motions from the CMU-Mocap database to generate 3-person scenes. The evaluation of these models is performed on both CMU-Mocap and MuPoTS-3D datasets.

For training the prediction of the social interaction type in the 3-people scenes, we annotated the first person $p1$ to not have any social interaction si_1 with the other two persons $p2, p3$, while we labeled the two other persons $p2, p3$ to have social interaction si_0. This annotation is in line with the way [21] prepared the dataset, where an additional person was added to the 2-person scenes in the mixing process.

For the input, we provide 15 frames (equivalent to 1000 ms) of historical data, and the models are tasked with predicting the subsequent 45 frames (corresponding to 3000 ms). We measure the performance by reporting the Mean Per Joint Position Error (MPJPE) at 1, 2, and 3 seconds into the future. To ensure a fair comparison, we utilize the code and data provided by [21] to train and evaluate each method.

Our findings, as presented in Table 2, show that our model consistently outperforms competing methods on both CMU-Mocap and MuPoTS-3D datasets. A visual comparison of the evaluated models is shown in Figure 5

When the dataset involves interactions among multiple individuals, as seen in the CMU-Mocap and MuPoTS-3D datasets, our approach is better at capturing and modeling these interactions, leading to better predictions of future poses.

Note: The architecture of our MPFSIR model requires a fixed number of people to be predicted in one step. While SoMoFormer [18] uses attention mechanisms to predict the poses of all people in the scene simultaneously, our model predicts the poses of two people simultaneously, so for datasets like CMU-Mocap and MuPoTS-3D, which contain three-person scenes, we perform two-person forecasting and combine the results afterward.

Table 2: Performance comparison of different models on the CMU-Mocap test set and MuPoTS-3D dataset using the MPJPE metric (expressed in meters), where lower MPJPE values indicate higher accuracy in joint position predictions. Our MPFSIR model exhibits superior accuracy, outperforming other models in accurately predicting human poses on both datasets.

Method	CMU-Mocap Test Set				MuPoTS-3D Test Set				Size
	1 sec	2 sec	3 sec	Overall	1 sec	2 sec	3 sec	Overall	#Param (M)
LTD [11]	4.03	7.06	9.91	7.00	1.75	2.98	4.10	2.94	2.61
MRT [21]	4.46	7.94	10.94	7.78	1.87	3.40	5.04	3.44	6.62
SoMoFormer [18]	4.50	8.15	11.27	7.79	1.69	3.02	4.15	2.95	4.88
MPFSIR	**3.94**	**7.04**	**9.87**	**6.95**	**1.67**	**2.87**	**3.93**	**2.82**	**0.24**

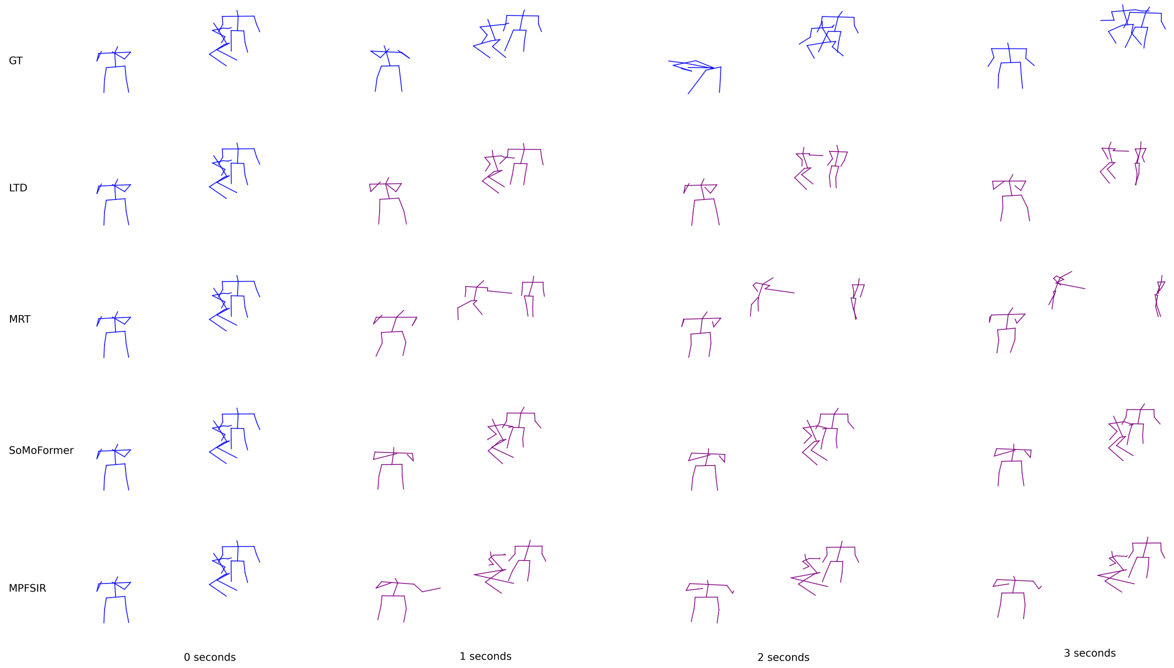


Figure 5: An example from the CMU-Mocap test set with forecasted poses from the evaluated models and ground truth (GT) poses.

4.4. Results of social interaction recognition

We conducted experiments to evaluate the prediction of the social interaction type between people in a scene using the 3DPW test dataset, from which we uniformly sampled data for each of the three classes: si_0, si_1, and si_2.

Figure 6 shows types of interactions between individuals in the scene, where si_0 denotes there is a social interaction between individuals, si_1 denotes there is no social interaction between individuals. In contrast, si_2 denotes that the two sequences are referring to the same individual. Each class contained 5039 samples with 2-person scenes. For evaluation, we used the model trained on the 3DPW train and AMASS dataset as described in section 4.2.

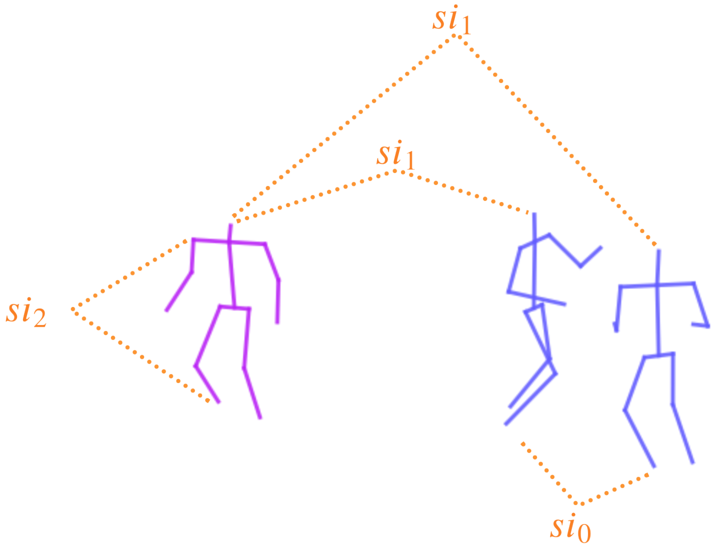


Figure 6: Example scene depicting social interactions between two individuals (blue) and an independent individual (purple). The figure visually illustrates the dynamic relationships and engagements among the individuals, showcasing the scene's complexity and diversity of social interactions. si_0 denotes a social interaction between individuals, si_1 denotes there is no social interaction between individuals, and si_2 denotes that the two sequences refer to the same individual.

As presented in Table 3, our model was evaluated using the mean F1 score, Accuracy, Precision, and Recall, a standard metric for multi-class classification. The results show that our model accurately predicted the type of social interaction between people, with an overall mean F1 score of 87.4. Specifically, our model performed best on the si_2 class, achieving an F1 score of 99.2. While evaluating class si_1, our model achieved an F1 score of 80.1 while achieving an F1 score of 82.1 on si_0 class which seems to suggest that recognizing social interaction between people is challenging. These results suggest that our model can effectively capture the type of social interaction between people in a scene.

Figure 7 provides a visual representation of the relationship between the probability of correctly classifying social interactions and the distance between individuals in the scene, which highlights the challenges of accurately predicting social interaction with individuals in close spatial proximity.

Table 3: Results on 3DPW train (which is used as a test dataset in SoMoF benchmark) for social interaction prediction.

Class	Precision ↑	Recall ↑	F1 score ↑	Accuracy ↑
si_0	78.8	85.6	82.1	85.6
si_1	84.6	77.6	80.1	77.6
si_2	99.3	99.0	99.2	99.0
Average	87.6	87.4	87.4	87.4

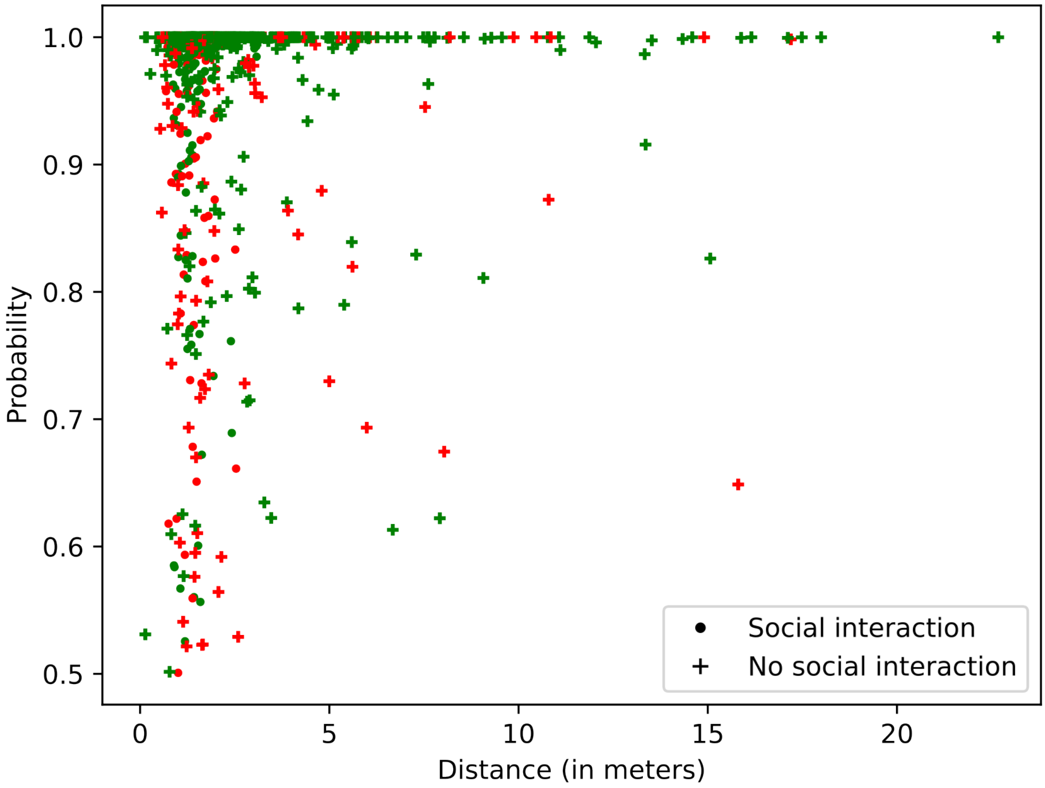


Figure 7: This figure illustrates the relationship between the probability of correctly classifying the interaction between individuals in the scene and the distance between them. The green color represents the correct classification, and the red represents the incorrect classification. The graph highlights how the probability of accurately identifying social interactions varies with the distance between individuals, where the most challenging interactions are with individuals in close proximity.

5. Ablation study

An ablation study was conducted on MPFSIR to understand the impact of its individual components on performance. The study involved adding each component individually to the model and evaluating the performance after each addition. The components evaluated in the study included the temporal and spatial context between sequences, the DCT transformation, data augmentation, and social interaction prediction. Results of the study showed that each of these components had a significant impact on the final performance of the model. Specifically, the addition of temporal and spatial context between sequences and data augmentation led to the most significant improvements in performance, while social interaction prediction had a relatively smaller impact. The study provided valuable insights into the role of each component in the model and helped identify the key factors responsible for the model's success. Table 4 displays the results of evaluating each model on VIM and MPJPE metrics after being trained on the SoMoF 3DPW training set and tested on the SoMoF 3DPW validation set.

Table 4: The presented results for the ablation study are based on the SoMoF 3DPW validation set and reported in VIM (top) and MPJPE (bottom). The baseline model is created with the same number of parameters as other models, while sequences are passed through the network independently (i.e. without joining them in TCTX and SCTX modules).

Method	100ms	240ms	500ms	640ms	900ms	Overall
Baseline	15.5	27.0	53.5	64.9	84.8	49.14
+ Temporal&Spacial CTX	11.7	23.0	47.1	57.7	77.4	43.38
+ DCT	9.6	22.2	48.1	59.8	82.7	44.48
+ Data augmentation	8.8	20.8	45.5	56.3	**76.5**	41.58
+ Social interaction	**8.6**	**20.5**	**45.3**	**56.3**	76.8	**41.50**
Baseline	3.6	6.3	12.8	15.8	21.1	11.92
+ Temporal&Spacial CTX	2.7	5.2	11.1	13.8	19.0	10.36
+ DCT	2.2	4.9	11.2	14.3	20.5	10.62
+ Data augmentation	2.0	4.6	10.4	**13.1**	**18.4**	9.70
+ Social interaction	**1.9**	**4.5**	**10.3**	13.2	18.5	**9.68**

6. MW-MPJPE: a pose forecasting evaluation metric

In this paper, we propose a new evaluation metric called Movement-Weighted Mean Per Joint Position Error (MW-MPJPE) to enhance the existing MPJPE metric for pose forecasting evaluation. While effective in assessing pose accuracy in the task of pose estimation as described by Šajina and Ivašić-Kos in [17], the commonly used Mean Per Joint Position Error (MPJPE) metric has a limitation in capturing the quality of learned movement patterns. Models optimized solely based on MPJPE often tend to reproduce the last observed pose, resulting in inflated performance scores without properly understanding and predicting the underlying skeletal movement, as shown in Figure 8.

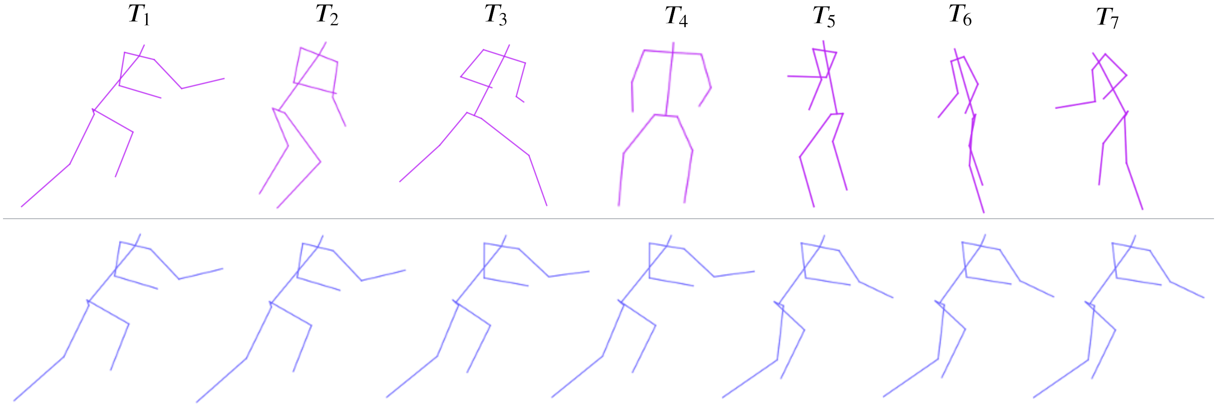


Figure 8: The figure illustrates a comparison between the ground truth poses (depicted in purple) and the predicted poses (depicted in blue) by various models. It demonstrates a recurring pattern where the models tend to accurately position the poses in the correct global location but exhibit limited skeletal movement. This discrepancy between the ground truth and predicted poses is not captured well by the regular MPJPE metrics.

To address this issue, MW-MPJPE introduces a crucial weighting factor that incorporates the overall movement exhibited by the individual throughout the target pose sequence. By scaling the MPJPE error with this movement information, MW-MPJPE provides a more comprehensive and nuanced assessment of the forecasted poses. This weighting scheme ensures that the metric accounts for the complexity and dynamics of the pose sequences, encouraging models to learn and predict meaningful movement patterns rather than solely optimizing for pose accuracy. By incorporating the concept of movement into the evaluation process, MW-MPJPE incentivizes models to capture and reproduce the natural dynamics of human motion, making it a more reliable and informative metric for pose forecasting tasks.

MW-MPJPE is calculated as follows:

$$P_{fixed} = P_{gt,\varphi} - P^{(0)}_{gt,\varphi}(pelvis)$$

$$\begin{gathered} E_{MW\text{-}MPJPE}(Y, \varphi) = \\ \sum_{f=1}^{F_{gt}} |P^{(f-1)}_{fixed} - P^{(f)}_{fixed}| \times \frac{1}{F_{gt}} \sum_{f=0}^{F_{gt}} E_{MPJPE}(Y_f, \varphi) \end{gathered} \tag{9}$$

where P_{fixed} denotes the gt sequence withouth spacial movement and is calculated by substracing $P_{gt,\varphi}$ sequence from pelvis $P^{(0)}_{gt,\varphi}$ at frame 0. F_{gt} denotes total number of frames in the sequence gt and φ denotes the corresponding skeleton. $P^{(f-1)}_{fixed}$ and $P^{(f)}_{fixed}$ are skeleton with all joints from pelvis-fixed sequence P_{fixed} at frames $f - 1$ and f. Y denotes a predicted sequence, while Y_f corresponds to the predicted skeleton at frame f.

Table 5 displays the results of evaluating each model on MW-MPJPE metrics on both CMU-Mocap and MuPoTS-3D datasets. The results highlight the effectiveness of the different methods in accurately predicting future poses and skeleton dynamics and provide insight into the difficulty of the dataset. The evaluation results on the CMU-Mocap and MuPoTS-3D datasets, expressed in the MW-MPJPE metric, clearly demonstrate that CMU-Mocap is a more challenging dataset. This is evident from the significantly larger errors obtained on CMU-Mocap, which correspond to the visually observed dynamic skeleton movements within the dataset. In contrast, MuPoTS-3D exhibits smaller skeleton movements, resulting in comparatively lower error rates.

Table 5: The table presents the evaluation results of various methods on the CMU-Mocap test set and MuPoTS-3D dataset, measured using the MW-MPJPE metric. The MW-MPJPE metric accurately assesses pose forecasting performance by considering both joint position errors and the magnitude of skeleton motion.

Method	CMU-Mocap Test Set				MuPoTS-3D Test Set				Size
	1 sec	2 sec	3 sec	Overall	1 sec	2 sec	3 sec	Overall	#Param (M)
LTD [11]	65.49	223.78	456.49	248.59	12.68	41.29	83.31	45.76	2.61
MRT [21]	72.32	249.70	492.35	271.46	13.48	47.34	102.40	54.41	6.62
SoMoFormer [18]	72.95	254.18	506.86	278.00	12.38	42.17	85.05	46.53	4.88
MPFSIR	**62.64**	**219.30**	**446.06**	**242.67**	**12.20**	**39.93**	**79.99**	**44.04**	**0.24**

7. Conclusion

In conclusion, this paper proposed the MPFSIR model, a novel approach for multi-person pose forecasting that leverages fully-connected layers with skip connections to capture temporal dependencies in the input pose sequences. The model demonstrates promising results in accurately predicting future poses and modeling social interactions between individuals in the scene. Through extensive experiments on the SoMoF Benchmark, CMU-Mocap, and MuPoTS-3D datasets, we have shown that our model outperforms existing methods in terms of accuracy and parameter efficiency. However, it is important to note the limitations of the MPFSIR model, such as its struggle to create valid long-term movements. Future research should focus on addressing these limitations and further enhancing the modeling of social interactions and long-term dependencies in pose forecasting. Overall, our work contributes to the advancement of pose forecasting techniques and opens up new avenues for applications in human motion analysis, virtual reality, and robotics.

This paper proposes the MPFSIR model, a new approach for multi-person pose forecasting that uses fully connected layers with skip-connections to model temporal dependencies in input pose sequences. The model has a small number of parameters and shows promising results in predicting future poses by relying on temporal information and modeling social interactions between individuals in a scene. Through extensive experiments on the SoMoF Benchmark, CMU-Mocap, and MuPoTS-3D datasets, we have shown that our model outperforms existing methods in terms of parameter accuracy and efficiency. However, it is important to note the limitations of the MPFSIR model, such as its problem generalizing to new or unusual poses, since the performance of the MPFSIR model is highly dependent on the quality and variety of the training data. Furthermore, the proposed model has a problem in predicting valid long-term movements as well as complex movement patterns. Other models that we evaluated and compared in this research have similar limitations. Namely, long-term movements often include different dependencies and intricate coordination among several individuals and different activities of different duration, which is difficult to predict if the semantic component, understanding the context, and recognizing the activities that are carried out are not included in the model. Future research should certainly focus on addressing these limitations and further improving the modeling of social interactions and long-term dependencies in position prediction. However, regardless of this, our work contributes to the advancement of position prediction techniques and opens new application possibilities in human motion analysis, virtual reality, and robotics.

Funding: This research was supported by Croatian Science Foundation under the project IP-2016-06-8345, "Automatic recognition of actions and activities in multimedia content from the sports domain" (RAASS), and by the University of Rijeka (project number 18-222).

References

[1] Vida Adeli et al. "Socially and contextually aware human motion and pose forecasting". In: *IEEE Robotics and Automation Letters* 5.4 (2020), pp. 6033–6040.

[2] Vida Adeli et al. "Tripod: Human trajectory and pose dynamics forecasting in the wild". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 13390–13400.

[3] *Carnegie Mellon University Motion Capture Database*. (accessed on 2 February 2025). URL: `https://paperswithcode.com/dataset/cmu-motion-capture`.

[4] Hsu-kuang Chiu et al. "Action-agnostic human pose forecasting". In: *2019 IEEE winter conference on applications of computer vision (WACV)*. IEEE. 2019, pp. 1423–1432.

[5] Wen Guo et al. "Back to mlp: A simple baseline for human motion prediction". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2023, pp. 4809–4819.

[6] Wen Guo et al. "Multi-person extreme motion prediction". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 13053–13064.

[7] Yingfan Huang et al. "Stgat: Modeling spatial-temporal interactions for human trajectory prediction". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 6272–6281.

[8] Naureen Mahmood et al. "AMASS: Archive of motion capture as surface shapes". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 5442–5451.

[9] Wei Mao, Miaomiao Liu, and Mathieu Salzmann. "History repeats itself: Human motion prediction via motion attention". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*. Springer. 2020, pp. 474–489.

[10] Wei Mao, Miaomiao Liu, and Mathieu Salzmann. "History repeats itself: Human motion prediction via motion attention". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*. Springer. 2020, pp. 474–489.

[11] Wei Mao et al. "Learning trajectory dependencies for human motion prediction". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 9489–9497.

[12] Angel Mart'inez-Gonz'alez, Michael Villamizar, and Jean-Marc Odobez. "Pose transformers (potr): Human motion prediction with non-autoregressive transformers". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 2276–2284.

[13] Omar Medjaouri and Kevin Desai. "Hr-stan: High-resolution spatio-temporal attention network for 3d human motion prediction". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 2540–2549.

[14] Dushyant Mehta et al. "Single-shot multi-person 3d pose estimation from monocular rgb". In: *2018 International Conference on 3D Vision (3DV)*. IEEE. 2018, pp. 120–130.

[15] Behnam Parsaeifard et al. "Learning decoupled representations for human pose forecasting". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 2294–2303.

[16] Xiaogang Peng et al. “SoMoFormer: Social-Aware Motion Transformer for Multi-Person Motion Prediction”. In: *arXiv preprint arXiv:2208.09224* (2022).

[17] Romeo Šajina and Marina Ivašić-Kos. “3D Pose Estimation and Tracking in Handball Actions Using a Monocular Camera”. In: *Journal of Imaging* 8.11 (2022), p. 308. DOI: `10.3390/jimaging8110308`.

[18] Edward Vendrow et al. “SoMoFormer: Multi-Person Pose Forecasting with Transformers”. In: *arXiv preprint arXiv:2208.14023* (2022).

[19] Timo Von Marcard et al. “Recovering accurate 3d human pose in the wild using imus and a moving camera”. In: *Proceedings of the European conference on computer vision (ECCV)*. 2018, pp. 601–617.

[20] Chenxi Wang et al. “Simple baseline for single human motion forecasting”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 2260–2265.

[21] Jiashun Wang et al. “Multi-person 3D motion prediction with multi-range transformers”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 6036–6049.

[22] Chongyang Zhong et al. “Spatio-temporal gating-adjacency GCN for human motion prediction”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 6447–6456.

B. GCN-Transformer: Graph Convolutional Network and Transformer for Multi-Person Pose Forecasting Using Sensor-Based Motion Data

This work was published as: Romeo Šajina, Goran Oreški, Marina Ivašić-Kos, 2025. GCN-Transformer: Graph Convolutional Network and Transformer for Multi-Person Pose Forecasting Using Sensor-Based Motion Data. Sensors, Volume 25, ISSN 1424-8220, DOI: 10.3390/s25103136

https://www.mdpi.com/1424-8220/25/10/3136

1. Introduction

Pose forecasting is a machine learning task that predicts future poses based on a historical sequence of poses. This task is inherently challenging, as it requires models to anticipate movements several seconds into the future, thereby necessitating the capture of intricate temporal dynamics. The goal of pose forecasting is to provide accurate predictions of future poses, which can have practical applications in a wide range of fields. For example, in robotics, pose forecasting models enable robots to infer human intentions and predict future movements, facilitating safer, more intuitive collaboration in environments such as manufacturing floors, healthcare, and assistive robotics [8, 15, 25, 27, 12, 18]. In sports analytics, forecasting player trajectories and body orientations several moments ahead supports tactical decision-making, performance evaluation, and even automated highlight generation. In autonomous driving, the accurate prediction of pedestrian motion improves vehicle navigation and enhances safety in complex urban settings. Intelligent surveillance systems use pose forecasting to proactively detect abnormal group behaviors, such as crowd surges or physical altercations, by identifying deviations from expected motion patterns. In virtual and augmented reality, forecasting full-body motion enables latency compensation and smoother avatar rendering during real-time collaborative experiences or immersive gameplay. These applications often rely on sensor-based motion capture systems, including vision-based sensors, inertial measurement units (IMUs), and depth cameras, to collect high-precision human movement data for training and inference [14, 30, 29].

One way to conceptualize pose forecasting is to divide it into two main categories: single-person [10, 4, 31, 25, 42, 16] and multi-person [43, 40, 36, 45, 32, 34] pose forecasting. In single-person pose forecasting, the task focuses on predicting the future poses of an individual based solely on their previous poses. This scenario is typically less complex, as it involves modeling the movement patterns of a single entity. On the other hand, multi-person pose forecasting extends the task by simultaneously predicting the future poses of multiple individuals. In this scenario, the forecasting model needs to consider each person's previous poses and extract social dependencies and interactions among them. These interactions could include factors such as proximity, response to a movement, and body language, which significantly influence the future movements of individuals within a scene.

Various deep learning methods have been employed to tackle the task of pose forecasting. Fully connected networks directly map input pose sequences to future predictions, which is suitable for straightforward temporal dependencies [4, 10, 36]. Recurrent neural networks (RNNs) capture long-range dependencies by maintaining hidden states across time steps [31]. Graph Convolutional Networks (GCNs) excel in modeling spatial dependencies and interactions in multi-person scenarios [25, 42, 34, 35]. Attention mechanisms and Transformer architectures focus on the relevant parts of input sequences, handling long-range dependencies effectively for precise predictions [43, 40, 45, 32].

The paper presents a novel model, GCN-Transformer, designed to address the challenges of multi-person pose forecasting. Our model integrates key features from various deep learning architectures to capture complex spatiotemporal dependencies and social interactions among multiple individuals in a scene. GCN-Transformer consists of two main modules: the Scene Module and the Spatiotemporal Attention Forecasting Module. The Scene Module leverages Graph Convolutional Networks (GCNs) to extract social features and dependencies from the scene context, while the Spatiotemporal Attention Forecasting Module utilizes a combination of Temporal Graph Convolutional Networks (T-GCNs) and Transformer decoder modules to predict future poses. By combining these components, GCN-Transformer achieves state-of-the-art performance in multi-person pose forecasting tasks, demonstrating its effectiveness in capturing intricate motion dynamics and social interactions. GCN-Transformer is trained and evaluated on sensor-based datasets CMU-Mocap, MuPoTS-3D, SoMoF Benchmark, and ExPI, which include motion capture data collected through real-world sensing systems. To enhance the learning process and improve the movement dynamics of predicted sequences while also capturing interaction dependencies, we introduce new loss terms during the training phase, specifically the multi-person joint distance loss and velocity loss. These loss terms are designed to encourage the model to learn both interaction dependencies and joint movement dynamics. The inter-individual joint distance loss focuses on maintaining realistic spatial relationships between joints, while velocity loss promotes the accurate modeling of movement dynamics.

Additionally, in this paper, we introduce a novel evaluation metric, Final Joint Position and Trajectory Error (FJPTE), designed to comprehensively assess pose forecasting performance. While several attempts have been made to develop evaluation metrics specifically for pose forecasting [36, 32, 1], these have predominantly been variations of well-known metrics such as MPJPE and VIM, both of which originate from the pose estimation domain. However, pose forecasting requires a more holistic approach that considers not only the final position of each joint but also the trajectory leading to that position. FJPTE addresses this need by evaluating both the final position and the movement dynamics throughout the trajectory, providing a more thorough assessment of how well a model captures the complexities of human motion over time.

Our contributions are as follows:

- We propose a new architecture and model that combines Graph Convolutional Networks (GCNs) and Transformer modules for multi-person pose forecasting; it is designed to handle complex interactions in dynamic scenes and consistently outperforms state-of-the-art models on standard evaluation metrics.
- Multi-person joint distance loss (MPJD) and Velocity Loss (VL) were designed to encourage the model to generate spatially interaction-dependent and temporally coherent pose sequences for dynamic and interaction-rich scenes.
- A new evaluation metric for pose forecasting, called FJPTE, that evaluates movement trajectories and the final position error, is proposed to better assess the realism and coherency of predicted pose sequences in dynamic and interaction-rich scenes.

In this work, we aim to address the challenge of forecasting future 3D poses in dynamic multi-person scenarios by designing a model that combines scene-level social context encoding with individual-specific forecasting using query token fusion. The architecture jointly models spatial dependencies within each individual and temporal motion patterns using both Transformer and GCN-based components. We evaluate the model across four datasets, CMU Mocap, MuPoTS 3D, SoMoF, and ExPI, which feature varying numbers of individuals and different levels of interaction complexity. This setup allows us to assess the robustness and generalization ability of the model across diverse motion conditions.

The organization of this paper is structured to comprehensively address the advancements and methodologies in multi-person pose forecasting. We begin with a review of the related work by discussing existing models and their limitations. Next, we define the problem formulation for multi-person forecasting, detailing the task's objectives and the necessary input and output representations. Following this, we introduce our proposed model, GCN-Transformer, which is elaborated through several subsections: the Spatiotemporal Fully Connected module for projecting sequences into a higher-dimensional embedding space; the Scene Module for capturing social interactions; and the Spatiotemporal Attention Forecasting Module for predicting future poses, data preprocessing, and augmentation techniques to enhance model performance, along with the training procedures employed. The Experimental Results Section follows, where we describe the metrics used for evaluation, the datasets involved, and the model's performance on the CMU-Mocap, MuPoTS-3D, SoMoF Benchmark, and ExPI datasets. We then present an ablation study to analyze the impact of different model components. Additionally, we introduce a novel evaluation metric, FJPTE, which assesses both local movement dynamics and global movement errors. Finally, we conclude the paper by summarizing the key findings and discussing future research directions.

2. Related Work

In the domain of pose forecasting, establishing a baseline is crucial, with the Zero-Velocity model serving as a simple yet effective benchmark. This model predicts future poses by duplicating the last observed pose. Remarkably, this baseline has emerged as a strong contender, outperforming numerous proposed models and thus providing a fundamental comparison point. Consequently, this paper exclusively discusses models that surpass this baseline performance.

2.1. Single-Person Pose Forecasting

Early explorations [42, 31, 25, 4, 10, 23, 48] focused predominantly on single-person pose forecasting. However, when applied to multi-person scenarios, these models independently conduct pose forecasting for each individual.

The LTD model introduced by Mao et al. in [25] uses a Graph Convolutional Network (GCN) with 12 blocks and residual connections, along with two additional graph convolutional layers placed at the beginning and end of the model to encode temporal information and decode features for pose prediction. The Future Motion model was proposed in [42] for single-person pose forecasting on a similar backbone architecture of 12 GCN blocks and also includes data augmentation, curriculum learning, and the use of Online Hard Keypoints Mining (OHKM) loss.

Parsaeifard et al. in [31] proposed a DViTA model that uses a Long Short-Term Memory (LSTM) encoder–decoder network for trajectory forecasting and a Variational LSTM AutoEncoder (VAE) for local pose dynamic forecasting in order to extract two distinct components of human movement: global trajectory and local pose dynamics.

MotionMixer, introduced by Bouazizi et al. in [4], proposes multi-layer perceptrons (MLPs) for pose forecasting and captures spatiotemporal dependencies through spatial mixing across body joints and temporal mixing across time steps by incorporating squeeze-and-excitation (SE) blocks to adjust the significance of different time steps. Guo et al. in [10] proposed siMLPe, a lightweight MLP-based model for pose forecasting that, in addition to having fully connected layers and carrying out layer normalization and transpose operations, contains a Discrete Cosine Transform (DCT) to encode temporal information and carry out residual displacement to predict motion.

Incorporating additional constraints into the problem's formulation, such as modeling human–scene interactions using per-joint contact maps to capture the distance between human joints and scene points, can enhance pose forecasting performance, as demonstrated by Mao, Hartley, Salzmann, et al. in [23]. This approach resolves issues such as "ghost motion", conditioning future human poses on predicted contact points.

Zhong et al. in [48] introduced a model called GAGCN that addresses the complex spatiotemporal dependencies in human motion data. The authors use a gating network to dynamically blend multiple adaptive adjacency matrices that capture joint dependencies (spatial) and temporal correlations.

2.2. Multi-Person Pose Forecasting

Recent advancements in multi-person pose forecasting have emphasized the integration of social interactions and dependencies among individuals within a scene, aiming to enhance model performance [43, 40, 36, 45, 32, 34, 17, 38, 46].

Wang et al. in [43] proposed a transformer-based architecture called the Multi-Range Transformer (MRT) that captures both local individual motion and global social interactions among multiple individuals. The MRT decoder predicts future poses for each person by attending to both local- and global-range encoder features. Additionally, a motion discriminator is incorporated into the training process to ensure the generated motions maintain natural characteristics.

The Transformer Encoder was used in the SoMoFormer model, introduced by Vendrow et al. in [40], which treats each input as a Discrete Cosine Transform (DCT)-encoded, padded trajectory of one joint. The SoMoFormer model simultaneously predicts pose trajectories for multiple individuals and uses attention mechanisms to model human body dynamics and the grid position of individuals for its spatial understanding.

In [36], Šajina and Ivasic-Kos proposed the MPFSIR model, which focuses on spatial and temporal pose information using fully connected layers with skip connections. Despite its relatively low model parameters, MPFSIR achieves state-of-the-art performances. Moreover, the model includes an auxiliary output to recognize social interactions between individuals, contributing to its overall performance improvement.

Xu et al. uses temporal differentiation of joints and explicit joint relations as inputs to a joint-relation transformer model called JRTransformer, introduced in [45], which models future relations between joints along with future joint positions.

TBIFormer, proposed by Peng, Mao, and Wu in [32], breaks down human poses into five body parts and models their interactions separately. It employs a Temporal Body Partition Module to transform sequences into a Multi-Person Body-Part sequence, retaining spatial and temporal information. The subsequent module, Social Body Interaction Self-Attention, aims to learn body part dynamics for both inter-individual and intra-individual interactions. Finally, a Transformer Decoder forecasts future movements based on the extracted features and Global Body Query Tokens.

In [34], Peng et al. proposed SocialTGCN, a convolution-based model comprising a Pose Refine Module (PSM) consisting of Graph Convolutional Network (GCN) layers, a Social Temporal GCN (SocialTGCN) encoder with GCN and Temporal Convolutional Network (TCN) layers, and a TCN decoder. Additionally, the SocialTGCN Module is fed a Spatial Adjacency Matrix constructed based on the Euclidean distance between the body root trajectories of individuals.

In recent years, several innovative approaches have emerged for creating multi-person forecasting models that diverge significantly from traditional approaches, offering new ways to handle the complexities of social interactions and motion dynamics. In the following, we discuss a few notable examples of these alternative approaches.

Jeong, Park, and Yoon in [17] have integrated pose forecasting with trajectory forecasting in their Trajectory2Pose model. This interaction-aware, trajectory-conditioned model first predicts multi-modal global trajectories and then refines local pose predictions based on these trajectories. It utilizes a graph-based person-wise interaction module to model inter-person dynamics and reciprocal forecasting of both global trajectories and local poses for improved prediction performance in multi-person scenarios.

In [38], Tanke et al. proposed a framework for predicting the poses of multiple individuals with mutual interactions that bases the prediction of future movements on past behaviors, and they also proposed a function that aggregates movement features across individuals, either by averaging or using multi-head attention to provide contextually plausible interactions for groups of different sizes. By leveraging causal temporal convolutional networks, the model processes the relationships between participants and generates realistic, socially consistent motions over extended time horizons.

Xu, Wang, and Gui in [46] proposed a framework (DuMMF) for stochastic multi-person pose forecasting that incorporates generative modeling and latent codes to model individual movements at the local level and social interactions at the global level. The model generates multiple different predictions for individual poses and social interactions, covering a range of possible outcomes. The approach is generalizable to various generative models, including GANs and diffusion models.

A prevalent technique in data preprocessing for pose forecasting involves the application of the Discrete Cosine Transform (DCT), which encodes human motion into the frequency domain represented by a set of coefficients. This transformation aids in noise reduction, thus improving the robustness of the data. Conversely, the Inverse DCT (IDCT) decodes predictions back to Cartesian coordinates, facilitating interpretation and application [10, 25, 23, 42, 43, 40, 32, 34, 17].

To further enhance the performance of pose forecasting models, a strategy often employed is dividing the task into short-term and long-term prediction models, also known as short-term and long-term optimization. In this approach, the final prediction is derived from a combination of outputs from both short-term and long-term models [42, 40, 45]. Additionally, another effective technique to improve transformer-based models is deep supervision. Here, the output of each block within the model is passed through the decoder model, thereby mitigating issues related to overfitting and enhancing model generalization [40, 45].

Despite the advancements in pose forecasting, including substantial advancements driven by GCN and Transformer architectures, several limitations persist that challenge the field. Current models often produce structurally invalid poses, where predicted poses do not reflect anatomically feasible configurations, rendering them unrealistic or impossible in real-world settings. Additionally, many models struggle to capture natural movement dynamics, leading to "ghosting" effects where poses appear frozen or drift unrealistically and lacking the fluidity and continuity expected in human motion. A further important issue is generalizability, where certain models achieve strong performance on specific datasets but frequently underperform when tested on different datasets, indicating an over-reliance on dataset-specific characteristics. To address these challenges, our proposed model is designed to improve the structural validity of predicted poses, enhance the realism of movement dynamics, and achieve more consistent performance across diverse datasets.

2.3. Pose Forecasting Evaluation Metrics

The evaluation of pose forecasting models involves adopting various metrics borrowed from related tasks, such as pose estimation [37, 21]. Initially, the Mean Per Joint Position Error (MPJPE) metric, borrowed from pose estimation, was widely used. However, it calculates the Euclidean distance (L2 norm) across all joints in the predicted sequence, providing an overall assessment of the model's performance without specifically focusing on human movement dynamics. To address this limitation, Adeli et al. in [1] introduced the Visibility-Ignored Metric (VIM). Unlike MPJPE, VIM evaluates the pose error solely at the last predicted frame, overlooking the trajectory of joints in preceding frames and focusing solely on the final pose error. MPJPE, along with VIM, has since become a standard evaluation metric for pose forecasting due to its simplicity, interpretability, and broad adoption in recent works.

Building upon the MPJPE metric, Šajina and Ivasic-Kos in [36] proposed the Movement-Weighted Mean Per Joint Position Error (MW-MPJPE). This metric enhances MPJPE by incorporating a weighting factor based on the overall movement exhibited by the individual throughout the target pose sequence. This weighting factor provides a more nuanced evaluation by considering the varying degrees of movement across different poses.

Peng, Mao, and Wu in [32] employed various evaluation metrics to assess multi-person pose forecasting models. These included the Joint Position Error (JPE), which resembles MPJPE but reports errors for all individuals in the scene; the Aligned Mean Per Joint Position Error (APE), which is akin to Root-MPJPE, focusing on pose position errors by removing global movement; and the Final Displacement Error (FDE), measuring the trajectory prediction error by considering only the final global position (e.g., pelvis) of each person.

Despite the introduction of several evaluation metrics, most existing metrics either focus solely on joint-wise positional errors or isolate specific aspects of motion, such as the final displacement. As a result, they often fail to provide a comprehensive view of both local movement dynamics and global motion trajectories over time. This highlights the need for a more complete pose forecasting metric that can jointly assess the error of predicted joint movements, as well as the overall realism and coherence of predicted human motion.

2.4. GCN and Transformer Hybrid Architectures in Related Fields

While significant progress has been made with Graph Convolutional Networks (GCNs) and Transformers individually, to the best of our knowledge, no prior work has successfully integrated these two architectures into a unified model specifically for the task of multi-person pose forecasting. This gap represents an opportunity for advancement, as combining the strengths of GCNs in capturing spatial dependencies and Transformers in modeling long-range temporal dynamics could lead to more robust and accurate predictions in complex, interaction-heavy scenarios. In this paper, we aim to bridge this gap by proposing GCN-Transformer, a novel model that leverages both GCN and Transformer architectures for multi-person pose forecasting, potentially setting a new standard in the field.

Although no previous work has applied a GCN-Transformer hybrid directly to multi-person pose forecasting, this combination has demonstrated considerable success across several related fields. These studies provide valuable insights into the benefits of integrating structured relational modeling with dynamic sequence modeling. In the following, we briefly review selected examples where GCN-Transformer hybrids have been effectively applied to tasks such as trajectory prediction [20, 3], time series forecasting [13, 44], and pose estimation [47, 7]. For example, Li, Pagnucco, and Song in [20] proposed a Graph-Based Spatial Transformer for predicting multiple plausible future pedestrian trajectories, which models both human-to-human and human-to-scene interactions by integrating attention mechanisms within a graph structure. Additionally, they present a Memory Replay algorithm to improve the temporal consistency of predicted trajectories by smoothing the temporal dynamics. Similarly, Aydemir, Akan, and Güney in [3] proposed a novel approach for predicting trajectories in complex traffic scenes. By utilizing a dynamic-weight learning mechanism, the model adapts to each person's state while maintaining a scene-centric representation to ensure efficient and accurate trajectory prediction for all individuals. The model leverages GCNs to capture spatial interactions between individuals and employs Transformer-based attention to model temporal dependencies.

GCN and Transformer architectures have also been successfully applied to time series forecasting, a task of predicting future time intervals based on historical data. For instance, Hu et al. in [13] introduced a GCN-Transformer model designed to handle complex spatiotemporal dependencies in EV-battery-swapping-station load forecasting. The model integrates Graph Convolutional Networks (GCNs) to capture spatial relationships between stations and a Transformer to model temporal dynamics, allowing it to manage both spatial and temporal information simultaneously. Similarly, Xiong et al. in [44] introduced a model for chaotic multivariate time series forecasting. The model utilizes a Dynamic Adaptive Graph Convolutional Network (DAGCN) to model spatial correlations across variables and applies multi-head attention from the Transformer to capture temporal relationships. This hybrid approach demonstrates the effective application of GCNs and Transformers in tasks that require managing complex nonlinear data, such as chaotic systems, showing strong interpretability and performance across benchmark datasets.

GCN and Transformer architectures have also been successfully applied to pose estimation, a task of detecting human joint positions from an image. For example, Zhai et al. in [47] proposed the Hop-wise GraphFormer (HGF) module, which groups joints by k-hop neighbors and applies a transformer-like attention mechanism to model joint synergies. Additionally, the Intragroup Joint Refinement (IJR) module refines joint features, particularly for peripheral joints, using prior limb information. Furthermore, Cheng et al. in [7] presents GTPose, a novel model combining Graph Convolutional Networks (GCNs) and Transformers to enhance 2D human pose estimation. The model uses multi-scale convolutional layers for initial feature extraction, followed by Transformers to model the spatial relationships between keypoints and image regions. To further refine predictions, a Graph Convolutional Network models the topological structure between keypoints, capturing the relationships between joints.

While prior works have combined GCNs and Transformers in tasks such as trajectory forecasting, time series prediction, and pose estimation, these models typically apply GCNs for spatial encoding followed by Transformers for temporal modeling in a sequential or stacked manner. In contrast, our architecture is structured as a modular pipeline that first models social contexts using a Spatial-GCN applied across all individuals in the scene. This shared context is then injected into per-person forecasting branches using query token fusion, allowing each branch to access global scene information alongside individual motion patterns. Additionally, our forecasting module jointly incorporates both Transformer-based attention mechanisms and Temporal GCNs, enabling the complementary modeling of long-range temporal dependencies and local graph-based dynamics. To our knowledge, no prior GCN-Transformer hybrid applies this architecture to multi-person pose forecasting with such explicit scene-person disentanglement and fusion.

3. Background of Graph Convolutional Networks and Transformers

In recent years, two of the most prominent architectures for tasks like pose forecasting have been Graph Convolutional Networks (GCNs) and Transformer architectures. To better understand their foundations and effectiveness, we will provide a formalized overview of these architectures. It is important to note that the following descriptions remain generalized relative to GCN and Transformer architectures and do not delve into their specific application to multi-person pose forecasting, as this has already been addressed in the Related Work Section.

3.1. Graph Convolutional Networks

Conventional Convolutional Neural Networks (CNNs) operate on grid-like data structures like images, while GCNs are designed to work with non-Euclidean data, such as graphs, which consist of nodes (vertices) and edges representing relationships between the nodes. A graph is formally defined as $G = (V, E)$, where V is the set of nodes and E is the set of edges. The key challenge in GCNs is to propagate information between nodes to capture the spatial structure of the graph.

GCNs can be broadly categorized into spatial and spectral graph convolutions [5]. Spatial-GCNs aggregate information from neighboring nodes based on their local structure. This aggregation can be extended to k-hop neighbors, where the neighborhood expands to include nodes within k steps of the target node, as in [2]. Spectral GCNs, on the other hand, transform graph data into the spectral domain, using the graph's Laplacian to perform convolutions, but these often encounter computational challenges due to the size of the graph kernel. A simplified version of spectral convolutions, proposed by Kipf and Welling in [19], utilizes a first-order approximation, which is widely adopted due to its computational efficiency.

The general form of a GCN layer can be represented as follows:

$$H^{(l+1)} = \sigma(\widetilde{A}\; H^l\; W^l) \tag{1}$$

where H^l represents the feature matrix at layer l, $\widetilde{A}$ is the normalized adjacency matrix, W^l is the learnable weight matrix at layer l, and σ is an activation function like ReLU.

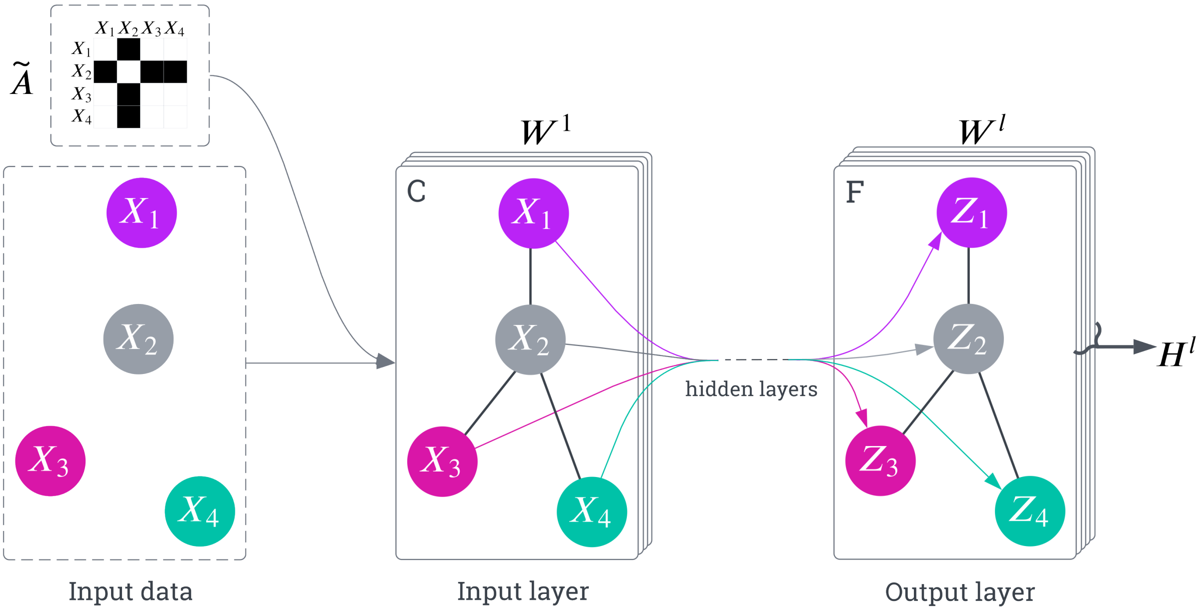


Figure 1: The figure depicts a multi-layer Graph Convolutional Network (GCN) architecture. The graph's structure, defined by the normalized adjacency matrix $\widetilde{A}$, is shared across all layers (edges shown as black lines). The input data (with C channels) are iteratively transformed at each layer l using $\widetilde{A}$ and a learnable weight matrix W^l. The final layer outputs feature maps, F, capturing node relationships and properties through stacked graph convolutions.

Figure 1 illustrates the multi-layer GCN architecture, highlighting how the input features are progressively transformed through successive layers using the shared graph structure defined by the normalized adjacency matrix $\widetilde{A}$. Traditionally, the adjacency matrix is predefined based on the structure of the graph (e.g., a human skeleton with fixed joint connections). However, in more advanced applications, especially in tasks like pose forecasting, the adjacency matrix $\widetilde{A}$ can be treated as a learnable parameter [9, 48], allowing the model to dynamically adapt the relationships between nodes (e.g., joints) based on the data. By making the adjacency matrix learnable, the network can adjust the strength or presence of connections between nodes, capturing more complex and data-driven relationships that may not be explicitly defined in the original graph. This is particularly useful for tasks involving non-static or flexible relationships, such as multi-person interactions or joint dynamics that change over time.

3.2. Transformer Architecture

The Transformer model, introduced by Vaswani in [39], has revolutionized the field of sequence modeling due to its effectiveness in capturing long-range dependencies and its parallel computation capabilities. Initially developed for natural language processing (NLP), where understanding contextual relationships between words across long sequences is essential, the Transformer architecture quickly surpassed traditional recurrent models such as LSTMs and GRUs. This success sparked widespread adoption across numerous domains, including computer vision, time-series forecasting, reinforcement learning, and human motion modeling.

Transformers rely on the attention mechanism that allows each element of the input sequence to interact with every other element. During processing, the attention mechanism assigns higher importance, or attention weights, to parts of the sequence that are most relevant for a given prediction or representation. This dynamic weighting enables the model to selectively focus on crucial inputs while diminishing the influence of less relevant ones, enhancing the ability to capture complex, long-range relationships without relying on sequential processing steps.

Because Transformers do not inherently model sequential order, they incorporate positional encodings into the input embeddings to preserve information about the position of each element within a sequence. These positional encodings can be predefined, typically using sine and cosine functions at varying frequencies [26, 43, 33, 32], or learned as trainable parameters during model optimization [40, 45]. By embedding positional information alongside content information, Transformers maintain the ability to reason about both the identity and the temporal order of elements, allowing them to capture complex sequential dependencies in various tasks.

Moreover, Transformers are inherently well suited for scenarios involving complex relational dynamics, a defining characteristic of sensor-based human motion data. Their global attention mechanism enables the model to dynamically prioritize the most relevant joints or individuals at each time step, allowing it to capture nuanced dependencies across space and time. This capability is particularly valuable in crowded or interaction-rich environments, where individual movements are not independent but influenced by the collective behavior of others in the scene.

At the core of the Transformer is the scaled dot-product attention, which computes the attention score as follows:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right) V \tag{2}$$

where Q, K, and V are the query, key, and value matrices derived from the input sequence, and d_k is the dimensionality of the key vectors. The softmax function ensures that the attention weights sum up to one, enabling the model to focus on relevant parts of the sequence. The scaling factor $\sqrt{d_k}$ prevents the dot-product values from growing too large, which could cause vanishing gradients during backpropagation [39].

To enhance the model's expressiveness, the Transformer uses multi-head attention, where multiple attention mechanisms run in parallel, and their outputs are concatenated:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)\ W^O \tag{3}$$

where $\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$, and W_i^Q, W_i^K, and W_i^V are learnable weight matrices for the queries, keys, and values, respectively. The outputs are then transformed by a final weight matrix W^O [39]. Figure 2 illustrates the calculations involved in the attention mechanisms of Transformers, including Scaled Dot-Product Attention and Multi-Head Attention, which aggregate multiple attention layers in parallel.

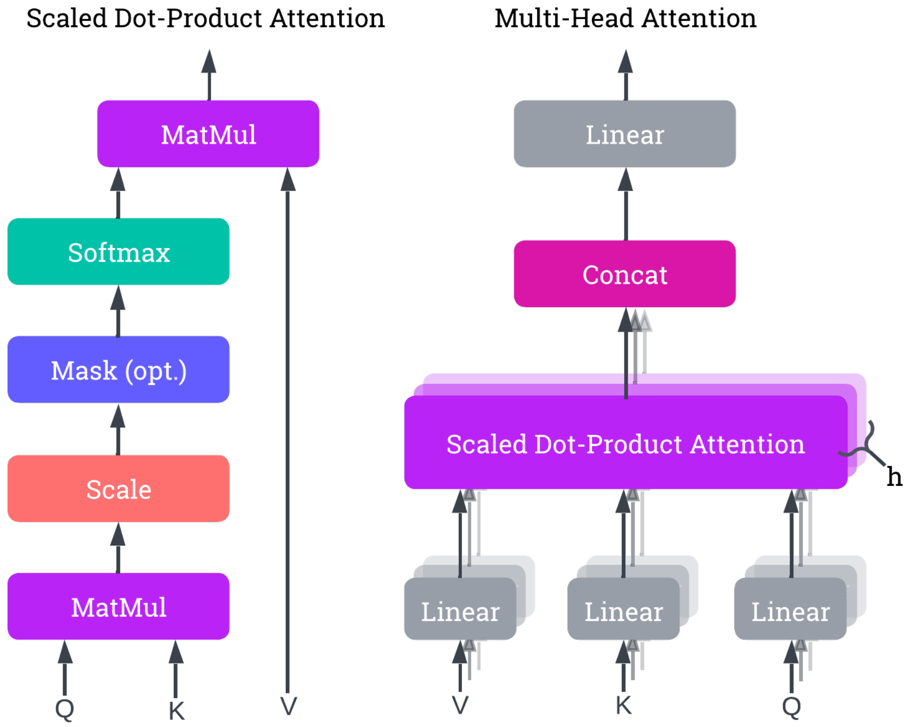


Figure 2: The figure illustrates the attention mechanism used in Transformer architecture. The left side depicts Scaled Dot-Product Attention, where the attention scores are computed using queries (Q), keys (K), and values (V), followed by scaling and a softmax operation. The right side shows Multi-Head Attention, consisting of multiple parallel Scaled Dot-Product Attention layers. The outputs of these parallel layers are concatenated and linearly transformed to produce the final attention output.

4. Problem Formulation for Multi-Person Forecasting

In the multi-person pose forecasting task, the aim is to forecast the forthcoming movements of multiple individuals within a given scene. Each individual in the scene is characterized by anatomical joints, typically including key areas such as elbows, knees, and shoulders. The task involves predicting the trajectories of these joints over a specified duration into the future, usually denoted by T time steps. To accomplish this predictive task, the model is provided with a sequence of historical poses for each individual. These historical poses encapsulate the positional information of each joint in three-dimensional Cartesian coordinates framed within a global coordinate system. This representation is standard in the field, as it reflects the native output of motion capture systems and 3D pose estimation models, and it allows for the straightforward computation of spatial relationships such as distances and velocities. For any given individual $n = 1 \ldots N$, each historical pose is represented by a vector of J dimensions, where J signifies the number of tracked joints. Consequently, the entire historical sequence for individual n is represented as $X^n_{1:t}$, capturing the temporal evolution of poses up to the present moment. The length of the input pose sequence, denoted as t, dictates the number of historical poses the model uses for prediction. The index n ranges from 1 to N, where N corresponds to the total number of individuals observed within the scene. At its core, the model's primary objective is to generate future pose sequences for each individual, denoted as $X^n_{t+1:T}$. Here, T reflects the future number of time steps that the model is tasked with forecasting. The problem's formulation is graphically shown in Figure 3.

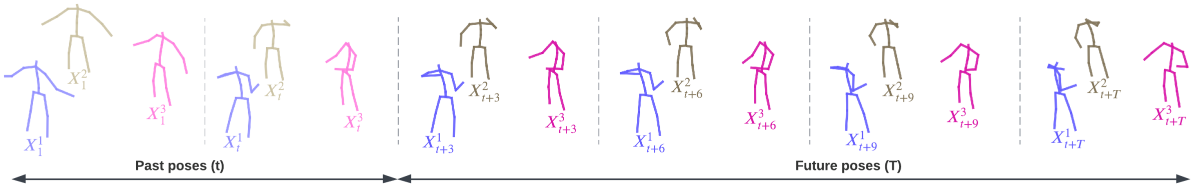


Figure 3: The figure illustrates the problem formulation for predicting the future movements of multiple individuals in a scene. Each individual is represented by joints (e.g., elbows, knees, shoulders), and the task is to forecast their trajectories over T time steps. The model receives historical pose sequences $X^n_{1:t}$ for each individual n, containing the positional data of joints in three-dimensional Cartesian coordinates. The objective is to predict future pose sequences $X^n_{t+1:T}$, extending T time steps into the future.

5. Proposed Architecture and Model

This paper proposes GCN-Transformer, a novel model for multi-person pose forecasting that emphasizes capturing complex interactions and dependencies between individuals within a scene. GCN-Transformer takes sequences of poses from all individuals in the scene as inputs, which are firstly preprocessed to enhance the data's richness. These sequences are then processed through the Scene Module, which is designed to capture the interactions and dependencies between individuals within the scene. Following this, the Spatiotemporal Attention Forecasting Module combines this contextual information with each individual's sequence to predict future poses. The following sections provide a detailed description of each component in the model's architecture.

The architecture of GCN-Transformer is guided by complementary theoretical principles from graph-based and attention-based modeling. Graph Convolutional Networks (GCNs) are well suited for capturing structured spatial relationships, such as the physical dependencies among joints and the social connections between individuals in a shared scene. These structures act as relational inductive biases that help the model reason over pose and proximity with minimal supervision. In contrast, Transformers are powerful tools for modeling long-range temporal dependencies and contextual interactions. Their self-attention mechanism allows for the dynamic weighting of information across time and between individuals, without requiring sequential computation. By combining GCNs and Transformers, GCN-Transformer is able to model both local and global dynamics, capturing individuals' joint relationships and interactions with temporal dependencies in multi-person scenes.

GCN-Transformer comprises two main modules: the Scene Module and Spatiotemporal Attention Forecasting Module. Initially, the input sequences, $X^{n \dots N}$, are padded with the last known pose's T times and augmented by incorporating their temporal differentiation, resulting in enriched sequences denoted as $Z^{n \dots N}$. Temporal differentiation refers to the process of computing the difference between joint positions across consecutive time steps to obtain motion velocity or first-order dynamics. Formally, for each person n, we compute $\Delta X_t^n = X_{t+1}^n - X_t^n$, and we concatenate this velocity signal with the original sequence along the joint feature's dimension. A zero-initialized frame is prepended to maintain temporal alignment. This results in a richer representation capturing both position and motion. These enriched sequences are concatenated and fed into the Scene Module. Within the Scene Module, a Spatiotemporal Fully Connected module encodes the poses into an embedding space. Subsequently, the output undergoes processing through the Spatial-GCN network designed to extract social features and dependencies. The resulting output S from the Scene Module is then forwarded into the Spatiotemporal Attention Forecasting Module for each n-th sequence Z^n, along with a query token Q^n generated through one-hot encoding based on the position of the n-th sequence within the scene.

In the Spatiotemporal Attention Forecasting Module, the sequence Z^n is encoded into the embedding space using a Spatiotemporal Fully Connected module (STFC). The resulting output is then concatenated with the extracted features S from the Scene Module and the query token Q^n to create W^n. This fusion combines individual motion, scene-level context, and identity-specific signal. $W^n = [\mathrm{STFC}(Z^n); S; Q^n]$, where $\mathrm{STFC}(Z^n) \in \mathbb{R}^{T\times d}$, $S \in \mathbb{R}^{T\times d}$, and $Q^n \in \mathbb{R}^{1\times d}$ (broadcasted across T). Subsequently, W^n is simultaneously passed into the Spatiotemporal Transformer Decoder and Temporal-GCN modules. The outputs from both modules are concatenated and processed through a Spatiotemporal Fully Connected module to generate the final prediction $\hat{y}^n$.

The architecture of GCN-Transformer is shown in Figure 4, and the full forward pass of GCN-Transformer is outlined in Algorithm 1.

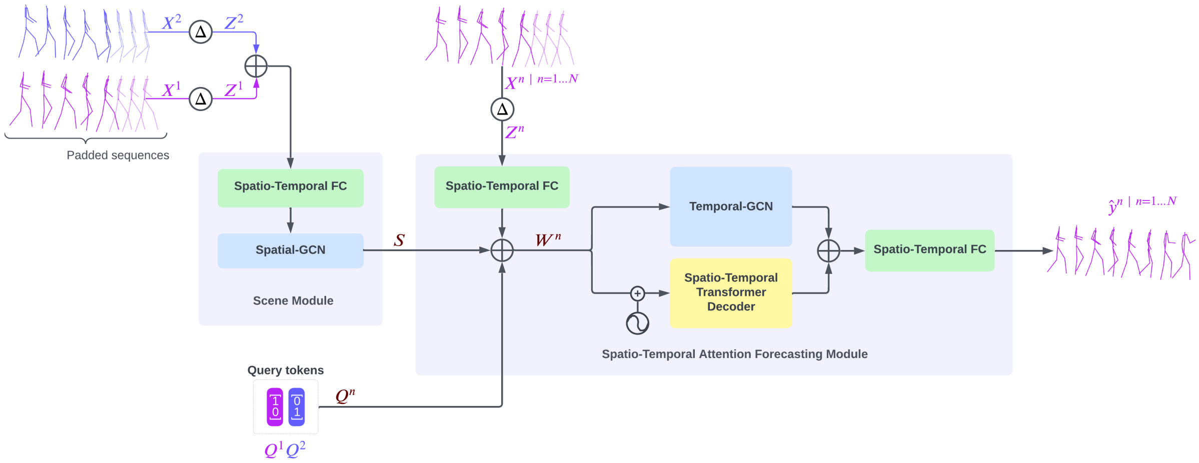


Figure 4: The figure depicts the architecture of the GCN-Transformer model. In the preprocessing step, input sequences X^1 and X^2 are padded with the last pose to match the full length of the sequence, and they are enriched with their temporal differentiation $\mathbf{\Delta}$, resulting in sequences Z^1 and Z^2. These sequences are then jointly processed by the Scene Module to extract social features and dependencies, producing the output S. Finally, to produce the final predictions, the output S is subsequently fed into the Spatiotemporal Attention Forecasting Module for each n-th sequence Z^n, along with a query token Q^n generated via one-hot encoding based on the position of the n-th sequence within the scene.

Algorithm 1: Pseudocode outlining the end-to-end forward pass of GCN-Transformer. The model first applies temporal differentiation to augment pose sequences for all individuals in the scene. These enriched sequences are embedded and passed through a Spatial GCN to extract scene-level context. Each individual's sequence is then fused with the scene context and an identity-specific query token before being processed in parallel by a Spatiotemporal Transformer Decoder and a Temporal GCN. The outputs are concatenated and passed through a final Spatiotemporal Fully Connected module to produce future pose predictions.

```
Input: Pose sequences $X_{1:t}^{1 \dots N}$ for $N$ individuals, each with $J$ joints in 3D space
Output: Predicted future pose sequences $\hat{Y}_{1:t+T}^{1 \dots N}$
Preprocessing:
foreach individual $n = 1$ to $N$ do
    Pad $X^n$ with last pose to length $t + T$
    Compute temporal difference: $\Delta X_t^n = X_{t+1}^n - X_t^n$
    Prepend zero velocity at $t = 1$ to align length
    Concatenate position and velocity: $Z^n = [X^n \parallel \Delta X^n]$
Stack enriched sequences: $Z = \{Z^1, \dots, Z^N\}$                // Shape: $T \times 3NJ$
Scene Encoding:                                                    // Run once per scene
Embed scene input: $E_{\text{scene}} = \textit{Spatiotemporal Fully Connected module}(Z)$
Compute context: $S = \textit{Spatial-GCN}(E_{\text{scene}})$                  // $S \in \mathbb{R}^{T \times d}$
Forecasting for each person:
foreach individual $n = 1$ to $N$ do
    Embed $Z^n$ using Spatiotemporal Fully Connected module:
    $E^n = \textit{Spatiotemporal Fully Connected module}(Z^n)$        // $E^n \in \mathbb{R}^{T \times d}$
    Generate identity token $Q^n$ (1-hot, broadcast to $T \times d$)
    Fuse inputs: $W^n = [E^n; S; Q^n]$
    Parallel decoding:
    $O^n_{\text{st-transformer}} = \textit{Spatio-Temporal Transformer Decoder}(W^n)$
    $O^n_{\text{temporal-gcn}} = \textit{Temporal-GCN}(W^n)$
    Concatenate: $O^n = [O^n_{\text{st-transformer}}; O^n_{\text{temporal-gcn}}]$
    Final prediction: $\hat{Y}^n = \textit{Spatiotemporal Fully Connected module}(O^n)$
return $\{\hat{Y}^1, \hat{Y}^2, \dots, \hat{Y}^N\}$
```

5.1. Spatiotemporal Fully Connected Module

The Spatiotemporal Fully Connected module is a lightweight component that projects pose sequences into a higher-dimensional embedding space, making them suitable for processing by downstream modules. It consists of two fully connected layers that independently process the spatial and temporal dimensions of the input. Given an input sequence $X \in \mathbb{R}^{T \times 3NJ}$, where T is the number of time steps, N is the number of individuals, J is the number of joints, and each joint is represented in 3D Cartesian space. The first fully connected layer operates along the spatial dimension, and it maps each frame-level pose vector of dimension $3NJ$ to a higher-dimensional representation, resulting in an intermediate output of shape $\mathbb{R}^{T \times d}$. Subsequently, a second fully connected layer is applied across the temporal dimension, allowing the model to capture short-term temporal patterns and refine the sequence-level encoding. The final output remains in $\mathbb{R}^{T \times d}$ and serves as the input to both the Scene Module and Spatiotemporal Attention Forecasting Module, where it is further processed by GCN and Transformer components.

5.2. Scene Module

The Scene Module is designed to enhance input data representation by leveraging temporal and spatial information. It comprises two key elements: a Spatiotemporal Fully Connected module and the Spatial-GCN. The Spatiotemporal Fully Connected module serves as an initial processing unit, transforming the enriched input sequence $Z^{n \ldots N}$ into a higher-dimensional embedding space, refining the input data and preparing them for subsequent modules through spatial and temporal transformations. In conjunction with the Spatiotemporal Fully Connected module, the Spatial-GCN module serves to uncover intricate patterns embedded within the data, specifically focusing on extracting interaction dependencies and dynamics among individuals within the scene. Comprising eight GCN blocks with learnable adjacency matrices, this module employs various techniques, including batch normalization, dropout, and Tanh activation functions, to enhance feature extraction and maintain the integrity of the structural information present in the input data. To further enhance the model's ability to capture social dependencies and maintain realistic spatial relationships between joints of the people in the scene, we compute the inter-individual joint distance loss on the output S.

5.3. Spatiotemporal Attention Forecasting Module

The Spatiotemporal Attention Forecasting Module predicts future poses by synthesizing information from various sources, including the input sequence Z^n, scene context S, and positional query token Q^n associated with sequence Z^n. Initially, the input sequence Z^n undergoes encoding via the Spatiotemporal Fully Connected module, transforming into an embedded space. Subsequently, this encoded sequence is concatenated with the scene context S and the positional query token Q^n to form W^n. This composite representation W^n undergoes parallel processing through two key components: the Spatiotemporal Transformer Decoder and the Temporal-GCN modules.

The Spatiotemporal Transformer Decoder comprises two attention blocks positioned after the learnable positional encoding of W^n. The first attention block is followed by fully connected layers that operate on the spatial dimension, facilitating the extraction of spatial features. Conversely, the second attention block is followed by Temporal Convolutional Network (TCN) layers, which specialize in capturing long-term temporal dependencies and temporal patterns within the data. Concurrently, the Temporal-GCN module, composed of eight GCN blocks with learnable adjacency matrices, operates on W^n to extract and refine temporal dependencies, thereby enhancing the temporal representation separate from the Spatiotemporal Transformer Decoder.

Finally, the Spatiotemporal Attention Forecasting Module integrates the extracted features using Spatiotemporal Fully Connected module, resulting in the generation of the final pose sequence prediction $\hat{y}^n$. This fusion process ensures that the module leverages the diverse information captured across spatial, temporal, and contextual dimensions to produce accurate and reliable predictions for future poses.

5.4. Data Preprocessing

We opted against employing any data preprocessing techniques for our model; instead, we utilized raw data from the datasets. This approach was chosen to compel the model to learn the intricate structure of the human skeleton and the dynamic nature of movement. Conventional preprocessing methods, such as employing Discrete Cosine Transform (DCT) to encode Cartesian coordinates into frequencies, often yield poses that appear ghost-like and lack the nuanced dynamics of human movement, like in [42, 43, 40, 36]. Moreover, techniques like predicting temporal differentiation that is subsequently added to the last known pose to generate the final result can produce invalid poses over the long term due to the model's lack of awareness regarding human structural information, like in [31, 43, 34, 32, 45].

5.5. Data Augmentation

Data augmentation is used for enhancing the robustness and generalization capability of pose forecasting models. Building upon methods utilized in [36], we extended the augmentation strategy with new methods to introduce further variations in the training data. Inspired by [36], we adopted several effective methods: sequence reversal, which reverses the temporal order of input sequences to expose the model to diverse temporal patterns; random person permutation, which shuffles the order of individuals within a scene to accommodate different person arrangements and interactions; random scaling, which introduces variations in pose scale to simulate varying heights of the people; random orientation, where poses are randomly rotated to simulate different camera viewpoints or human orientations; and random positioning, which shifts the positions of individuals within the scene to introduce spatial variability.

Expanding upon these methods, we introduced new techniques to enrich the dataset further. One method involved randomizing the joint order of individuals in a scene, encouraging the model to learn complex skeleton representations and adapt to different joint configurations. Additionally, we used a method to randomize the XYZ axes of individuals, enhancing pose variation by altering the orientation and positioning of poses in 3D space. Lastly, we varied the dataset's sampling frequency, using frequencies 1–4 to capture slower and faster sequences, though this type of sampling is performed during the preprocessing step.

All augmentations, except for sampling frequencies, are applied dynamically to each sampled batch of scene sequences during training. Each augmentation method is applied with a specific probability, introducing controlled variability into the training data. For instance, sequence reversal, random person permutation, random scaling, and random positioning each have a 50% probability of being applied, while random orientation, random joint order, and random XYZ-axis order are applied with a 25% probability. Furthermore, there is a 25% probability that no augmentation will be applied to a given sequence, ensuring that the model is exposed to both augmented and unaugmented data. These augmented datasets enable the model to learn robust features and adapt effectively to diverse scenarios, improving its performance and generalization capability in pose forecasting tasks.

We progressively introduced each method during development and empirically observed consistent reductions in training loss, indicating improved learning dynamics. All augmentation strategies were designed to preserve structural validity, and none produced implausible or invalid pose sequences. Importantly, all augmentations in our pipeline are applied consistently across the entire scene, meaning that the same transformation is applied to all individuals' pose sequences within a given scene to ensure that augmented motions remain coherent and socially consistent. Furthermore, since each augmentation process is applied with controlled probability and independently of others, we found no clear evidence of conflicting interactions or degradation in data quality. In practice, the combined use of all proposed augmentations led to the most effective training results across all datasets, as we also show in the ablation study (Section 7).

5.6. Training

Our model optimizes its parameters by minimizing the error between the predicted and ground truth poses, using a loss commonly referred to as reconstruction loss (REC).

This is a standard approach in pose forecasting and is widely adopted in prior work due to its simplicity and direct correlation with spatial prediction accuracy. REC is typically computed as the L2 distance between corresponding joints in the predicted and ground truth sequences, ensuring that the forecasted poses remain close to the true positions frame by frame.

However, while REC provides a useful baseline for learning pose positions, it has several limitations, particularly in the context of multi-person and dynamic motion forecasting. REC measures pose similarity on a per-joint, per-frame basis, and as such, it does not account for the temporal continuity of movements or the relational dynamics between individuals. This can lead to predicted sequences that are spatially accurate in isolated frames but lack smoothness over time or consistency in movement dynamics. For instance, a model trained with REC alone may generate plausible individual poses that result in jittery motion or unrealistic group behavior, such as individuals moving without regard for nearby participants.

To address these shortcomings, we introduce two additional loss terms that target complementary aspects of human motion. First, the multi-person joint distance (MPJD) loss enhances the model's ability to capture social and spatial interactions by penalizing discrepancies in joint distances between individuals across time. This encourages the Scene Module to improve model interaction dependencies and produce socially coherent pose sequences. Second, we incorporate a Velocity loss (VL), which prioritizes the learning of consistent temporal dynamics. By penalizing deviations in joint velocities between predicted and ground truth sequences, the VL term helps the model generate smoother and more realistic motion trajectories, reducing jitter and improving the fluidity of movement. The effectiveness of both additional losses is demonstrated in the ablation study (Section 7).

The final loss function is determined by combining the standard reconstruction loss with an additional multi-person joint distance loss (MPJD), scaled by a factor denoted as γ, used to adjust the effect of the MPJD loss on the overall loss. Both the output and scene predictions are subjected to Velocity Loss (VL), with Velocity Loss for the output from the Scene Module also scaled by the γ factor. To measure the error between the predicted and ground truth coordinates, we employ L_2-norm loss, aiming to minimize this error during training.

The final loss is calculated as follows:

$$\mathcal{L}_{\text{REC}} = \frac{1}{N}\sum_{n=1}^{N}\|\hat{y}_n - y_n\|_2 \tag{4}$$

$$\mathcal{L}_{\text{MPJD}} = \frac{1}{N(N{-}1)}\sum_{n=1}^{N}\sum_{p=1}^{N}\|(\hat{y}_n - \hat{y}_p) - (y_n - y_p)\|_2 \tag{5}$$

$$\mathcal{L}_{\text{REC\_VL}} = \frac{1}{N}\sum_{n=1}^{N}\|\Delta\hat{y}_n - \Delta y_n\|_2 \tag{6}$$

$$\mathcal{L}_{\text{MPJD\_VL}} = \frac{1}{N(N-1)}\sum_{n=1}^{N}\sum_{p=1}^{N}\left\|\Delta\hat{d}_{n,p} - \Delta d_{n,p}\right\|_2 \tag{7}$$

$$\mathcal{L} = \mathcal{L}_{\text{REC}} + \mathcal{L}_{\text{REC\_VL}} + \mathcal{L}_{\text{MPJD}} \times \gamma + \mathcal{L}_{\text{MPJD\_VL}} \times \gamma \tag{8}$$

where N represents the number of people in the scene; $\hat{y}_n$ and $\hat{y}_p$ represent the predicted pose sequence of the n-th and p-th person in the scene, while y_n and y_p represent the corresponding ground truth pose sequence of n-th and p-th person in the scene. $\|\cdot\|_2$ denotes the Euclidean distance (L2 norm), and $\frac{1}{N}\sum_{n=1}^{N}$ represents the mean distance across all people in the scene. The Δ represents temporal differentiation, where $\Delta y_n = y_n^t - y_n^{t+1}$ for $t = 0, 1, \ldots, T-1$ and $\Delta\hat{y}_n = \hat{y}_n^t - \hat{y}_n^{t+1}$ for $t = 0, 1, \ldots, T-1$. The predicted velocities of joint distances between individuals are represented with $\Delta\hat{d}_{n,p}$, while $\Delta d_{n,p}$ represents the ground truth velocities of joint distances between individuals.

Including MPJD and VL losses in the training process significantly enhances the practical applicability of multi-person pose forecasting models in real-world scenarios. The MPJD loss encourages the model to learn interaction dynamics between individuals in a scene, helping it capture how one individual's movements influence others. This is particularly useful in scenarios such as crowd monitoring, group behavioral analysis, and human–robot collaboration, where understanding interpersonal interactions is essential. On the other hand, the VL loss emphasizes temporal velocities between subsequent poses, promoting the generation of fluid and natural motion sequences. This is crucial in applications like animation, virtual reality, and autonomous systems, where smooth and realistic motion transitions are essential. Together, these losses address the challenges of producing rigid or disconnected poses, ensuring that the model generates dynamic, context-aware predictions.

We trained our model for 512 epochs with a batch size of 256, which was the largest manageable size given our hardware constraints. The extended training duration was chosen to accommodate the strong and dynamic augmentation strategy, which introduced extensive variability to the data, necessitating longer training for the model to effectively learn from these variations. Observing that the performance improvements plateaued at around 512 epochs, we determined that this duration was sufficient for optimal convergence. The Adam optimizer, a standard choice in pose forecasting, was chosen due to its adaptability and efficiency in handling complex, dynamic loss landscapes, especially with the strong augmentations applied. After testing multiple learning rates, we set an initial learning rate of 0.001, finding that it balanced effective learning with stability. A higher learning rate caused the loss to oscillate heavily, likely due to abrupt shifts in the solution space introduced by the strong augmentation, and in some cases, gradients would explode. To guide the model closer to the optimal solution, we reduced the learning rate to 0.0001 after 256 epochs, ensuring smoother convergence in the later stages of training. We also carefully tuned the γ parameter, which scales the MPJD loss, by analyzing values from 0 to 1. A value of 0.1 was selected, as it provided the best balance in guiding the model to capture both spatial dependencies and movement dynamics effectively.

6. Experimental Results

In our experimental evaluation of the GCN-Transformer, we employed four distinct datasets: CMU-Mocap, MuPoTS-3D, SoMoF, and ExPI. To assess the model's performance, we define evaluation metrics that quantify the error between predicted poses and ground truth. Through comprehensive analysis, we evaluated our model's performance on all datasets and conducted a comparative study against state-of-the-art models in the domain of multi-person pose forecasting. All models used for the experimental results were retrained from scratch using their official implementations, with the exception of Future Motion, which we re-implemented based on the details provided in the original paper. We followed the reported training protocols and hyperparameters wherever available and performed validation-based tuning only for Future Motion due to missing implementation details. All models were trained and evaluated under a consistent experimental setup to ensure a fair and meaningful comparison with our proposed method.

6.1. Metrics

The MPJPE (Mean Per Joint Position Error) is a commonly used metric for evaluating the performance of pose forecasting methods [24, 43, 40, 36, 45]. It measures the average Euclidean distance between the predicted joint positions and the corresponding ground truth positions across all joints. The lower the MPJPE value, the closer the predicted poses align with the ground truth. This metric provides a joint-level assessment of pose forecasting performance. The MPJPE metric is calculated as follows:

$$E_{\text{MPJPE}}(\hat{y}, y, \varphi) = \frac{1}{J_\varphi} \sum_{j=1}^{J_\varphi} \left\| P^{(f)}_{\hat{y},\varphi}(j) - P^{(f)}_{y,\varphi}(j) \right\|_2 \tag{9}$$

where f denotes a time step, and φ denotes the corresponding skeleton. $P^{(f)}_{\hat{y},\varphi}(j)$ is the estimated position of joint j, and $P^{(f)}_{y,\varphi}(j)$ is the corresponding ground truth position. J_φ represents the number of joints. $\|\cdot\|_2$ denotes the Euclidean distance (L2 norm), and $\frac{1}{J_\varphi}\sum_{j=1}^{J_\varphi}$ represents the mean distance across all joints.

Another commonly employed metric in pose forecasting evaluation is the Visibility-Ignored Metric (VIM), initially proposed by Adeli et al. in [1]. The VIM is computed by assessing the mean distance between the predicted and ground truth joint positions at the last pose T. This calculation involves flattening the joint positions and coordinates dimensions into a unified vector representation, resulting in a vector dimensionality of $3J$, where J denotes the number of joints. Subsequently, the Euclidean distance (L2 norm) is computed between the corresponding ground truth and predicted joint positions. The average distance across all joints yields the final VIM score. The SoMoF Benchmark adopts this metric for its evaluation framework. The VIM metric computation can be expressed as follows:

$$E_{\text{VIM}}(\hat{y}, y, \varphi) = \frac{1}{3J_\varphi} \sum_{j=1}^{3J_\varphi} \left\| P^{(j)}_{\hat{y},\varphi} - P^{(j)}_{y,\varphi} \right\|_2 \tag{10}$$

where J represents the number of joints, $P^{(i)}_{y,\varphi}$ is the ground truth position of the i-th joint (flattened), $P^{(i)}_{\hat{y},\varphi}$ is the predicted position of the i-th joint (flattened), $\|\cdot\|_2$ denotes the Euclidean distance (L2 norm), and $\frac{1}{3J_\varphi}\sum_{j=1}^{3J_\varphi}$ represents the mean distance across all joints.

6.2. Datasets

We employed distinct datasets for both training and evaluation, aligning with the methodology of previous models such as SoMoFormer [40], MRT [43], MPFSIR [36], and JRTransformer [45]. For training, we utilized the 3D Poses in the Wild (3DPW) [41] and Archive of Motion Capture As Surface Shapes (AMASS) [22] datasets. The 3DPW dataset contains over 60 video sequences containing scenes with two individuals, capturing human motion in real-world scenarios, including accurate reference 3D poses in natural scenes, such as people shopping in the city, having coffee, or playing sports, recorded with a moving hand-held camera. The dataset was collected using a combination of vision-based sensors and inertial measurement units (IMUs), which provided high-fidelity motion tracking in unconstrained environments. To adhere to the evaluation protocol of the SoMoF benchmark [1], we employed a specific split of the 3DPW dataset, where the train and test sets are inverted. Thus, we trained all models on the 3DPW test set and subsequently evaluated them on the 3DPW train set. This inversion was originally introduced by the authors of the SoMoF benchmark [1] due to the preprocessing of the 3DPW dataset, which created a larger number of sequences in the test set than in the training set, thus inverting the datasets allowed for a more robust training set. By following this protocol, we ensure that our results are directly comparable with other multi-person pose forecasting models evaluated under the same conditions. Specifically, for the SoMoF test set, data from the original 3DPW training set were sampled without overlap, producing distinct pose sequences. In contrast, the SoMoF training set was generated by sampling the original 3DPW testing set with overlap, employing a sliding window of 1 to capture a broader range of pose variations. The validation set remained consistent with the original 3DPW dataset, which was sampled without overlap.

On the other hand, the AMASS dataset provides an extensive collection of human motion capture sequences, totaling over 40 h of motion data and 11,000 motions represented as SMPL mesh models. AMASS unifies multiple optical marker-based motion capture datasets within a common framework, where motion data were originally collected using high-precision marker-based tracking systems. During the training process, we utilized the CMU, BMLMovi, and BMLRub subsets of the AMASS dataset, which provided a diverse and large-scale dataset. Given that many sequences within this dataset are single-person, we employed a technique to synthesize additional training data by combining sampled sequences to generate multi-person training data.

In contrast to recent works [43, 40, 36, 45, 32] that utilize the SoMoF Benchmark [1] alongside the Carnegie Mellon University Motion Capture Database (CMU-Mocap) [6] and the Multi-person Pose Estimation Test Set in 3D (MuPoTS-3D) [28] for model evaluation, our study additionally presents results on the Extreme Pose Interaction (ExPI) [11] dataset.

The CMU-Mocap and MuPoTS-3D datasets contain scenes with three individuals, with approximately 8000 annotated frames of poses across 20 real-world scenes. However, the movements captured are primarily simplistic, with limited interactions, often resulting in sequences where individuals maintain largely static poses or perform minimal motions. While we include evaluations on CMU-Mocap and MuPoTS-3D to ensure completeness and facilitate comparison with prior works, we emphasize that models trained or evaluated on these datasets may struggle to demonstrate their full capabilities in forecasting socially coherent, dynamic multi-person motion.

Therefore, after presenting initial results on CMU-Mocap and MuPoTS-3D, we focus our full analysis on the SoMoF Benchmark and the Extreme Pose Interaction (ExPI) dataset, both of which feature two-person scenes but offer significantly more challenging and realistic multi-person motion scenarios. In particular, ExPI contains dynamic sequences involving two couples engaged in physically demanding and interaction-heavy activities. The dataset was collected using a multi-sensor motion capture system comprising 68 synchronized and calibrated RGB cameras, along with a high-resolution infrared-based motion capture setup featuring 20 infrared mocap cameras. This comprehensive setup makes ExPI particularly well suited for evaluating complex, coordinated multi-person interactions in controlled yet naturalistic settings.

6.3. Results on CMU-Mocap and MuPoTS-3D

We first evaluate the GCN-Transformer against several state-of-the-art (SOTA) multi-person pose forecasting models, including MRT [43], Future Motion [42], SoMoFormer [40], JRTransformer [45], LTD [25], and MPFSIR [36]. Following established protocols, we trained all models using a synthesized dataset created by combining sampled motions from the CMU-Mocap database to simulate three-person interaction scenes. Evaluations were conducted on both test sets from the CMU-Mocap and MuPoTS-3D datasets.

For the Carnegie Mellon University Motion Capture Database (CMU-Mocap) [6], we adopt the training and testing splits provided by Wang et al. in [43]. Specifically, the dataset's construction involves combining two-person motion sequences with an additional randomly sampled third individual, introducing a degree of randomness into the generated scenes. To ensure fairness, the same generated datasets are used across all evaluated models.

Each input sequence consists of 15 historical frames (corresponding to 1000 ms), and the models are tasked with forecasting the subsequent 45 frames (3000 ms into the future). Each individual's pose is annotated with 15 joints, provided both as inputs and as ground truth for evaluation. We assessed performance using the Mean Per Joint Position Error (MPJPE) metric, which is reported at 1, 2, and 3 s into the future to align with evaluation from [43]. All models are retrained and evaluated under identical conditions using the official code and data released by [43].

As summarized in Table 1, the GCN-Transformer consistently outperforms all competing methods on both CMU-Mocap and MuPoTS-3D datasets, achieving new state-of-the-art performance in these settings.

The results demonstrate that the proposed GCN-Transformer consistently outperforms all competing models across both the CMU-Mocap and MuPoTS-3D test sets. These improvements are observed consistently across short-term and long-term forecasting horizons, indicating the model's strong ability to maintain prediction performance even as the forecast extends further into the future. Among the baselines, MPFSIR, JRTransformer, and LTD perform relatively competitively but still lag behind GCN-Transformer at all evaluation points. Interestingly, the model LTD, designed for single-person forecast-

Table 1: Performance comparison on the test sets of the CMU-Mocap and MuPoTS-3D datasets, featuring three-person scenes. Results are reported using the MPJPE metric (in meters), where lower values indicate better joint position prediction accuracy. Our proposed GCN-Transformer consistently achieves state-of-the-art results, outperforming all competing models on both datasets.

Method	**MPJPE Metric**								
	CMU-Mocap Test Set				**MuPoTS-3D Test Set**				**Average Overall**
	1 s	**2 s**	**3 s**	**Overall**	**1 s**	**2 s**	**3 s**	**Overall**	
Zero Velocity	5.55	9.23	12.30	9.03	2.05	3.43	4.57	3.35	6.29
MRT [43]	4.46	7.94	10.94	7.78	1.87	3.40	5.04	3.44	5.61
SoMoFormer [40]	4.50	8.15	11.27	7.79	1.69	3.02	4.15	2.95	5.37
Future Motion [42]	4.08	7.24	10.21	7.18	1.98	3.40	4.57	3.31	5.25
JRTransformer [45]	4.08	7.47	10.47	7.34	1.61	2.90	4.06	2.86	5.16
LTD [25]	4.03	7.06	9.91	7.00	1.75	2.98	4.10	2.94	4.97
MPFSIR [36]	3.94	7.04	9.87	6.95	1.67	2.87	3.93	2.82	4.89
GCN-Transformer (our)	**3.53**	**6.58**	**9.25**	**6.46**	**1.39**	**2.41**	**3.39**	**2.40**	**4.43**

Best results in each column are highlighted in bold.

ing, performs relatively well given its lack of explicit multi-person modeling capabilities. In contrast, models such as MRT, SoMoFormer, and Future Motion show substantially higher errors, particularly as the forecast horizon increases, suggesting weaker mechanisms for modeling long-term temporal dependencies in multi-person settings. It is also noteworthy that the ordering of model performance shifts between the CMU-Mocap and MuPoTS-3D datasets. This variability indicates that many models are sensitive to the specific characteristics of the dataset and highlights a lack of consistent generalization ability across different multi-person forecasting environments.

The strong results achieved by the GCN-Transformer highlight its ability to forecast complex multi-person motion accurately over both short and long time horizons. Its consistent improvements across different datasets demonstrate robustness and generalization. These findings validate the importance of combining spatial and temporal reasoning for multi-person forecasting tasks. In the following sections, we further evaluate GCN-Transformer on more socially complex datasets (SoMoF and ExPI) to assess its performance in even more dynamic and challenging scenarios.

6.4. Results on SoMoF Benchmark

The SoMoF Benchmark, introduced by Adeli et al. in [1], serves as a standardized assessment platform for evaluating the performance of multi-person pose forecasting models. The SoMoF Benchmark is derived from the 3DPW dataset, where every other frame is sampled to lower the original frames per second (FPS) from 30 to 15. This benchmark task involves predicting the subsequent 14 frames (equivalent to 930 milliseconds) based on 16 frames (1070 milliseconds) of preceding input data, encompassing joint positions for multiple individuals. The evaluation uses the Visibility-Ignored Metric (VIM), measuring performances across various future time steps. Similarly to [42, 40, 45, 36], all evaluated models in this paper were trained to utilize data from the 3DPW [41] and AMASS [22] datasets. During training, emphasis was placed solely on the 13 joints evaluated within the SoMoF framework. To ensure fairness in the comparisons, a practice observed in various studies such as [45, 32, 34] was adopted, whereby the final results are reported based on the epoch with the lowest average VIM score on the test dataset. Furthermore, problem formulation remained consistent for all evaluated models, focusing on predicting the next 14 frames using 16 input data frames. This differs from methodologies advocated by [42, 40, 45] to divide formulations into two separate problem formulations for short-term and long-term optimization, which inherently enhances the model's performance.

We conducted a comparative analysis of evaluated methods on the SoMoF Benchmark test set, as presented in Table 2, demonstrating that our model consistently achieves state-of-the-art results compared to competing models.

The results demonstrate the superior performance of the proposed GCN-Transformer across both VIM and MPJPE metrics, establishing it as a state-of-the-art solution in multi-person pose forecasting. While SoMoFormer emerges as a formidable competitor, particularly in long-term forecasting, GCN-Transformer consistently outperforms all models, especially when considering the overall metric, which aggregates performance across all evaluated time intervals. Interestingly, despite the reported similar performance to SoMoFormer, the JRTransformer fails to achieve competitive results in this evaluation. Conversely, the Future Motion model, introduced in 2021, demonstrates commendable performance, rivaling even the most recent state-of-the-art models. The MPFSIR model

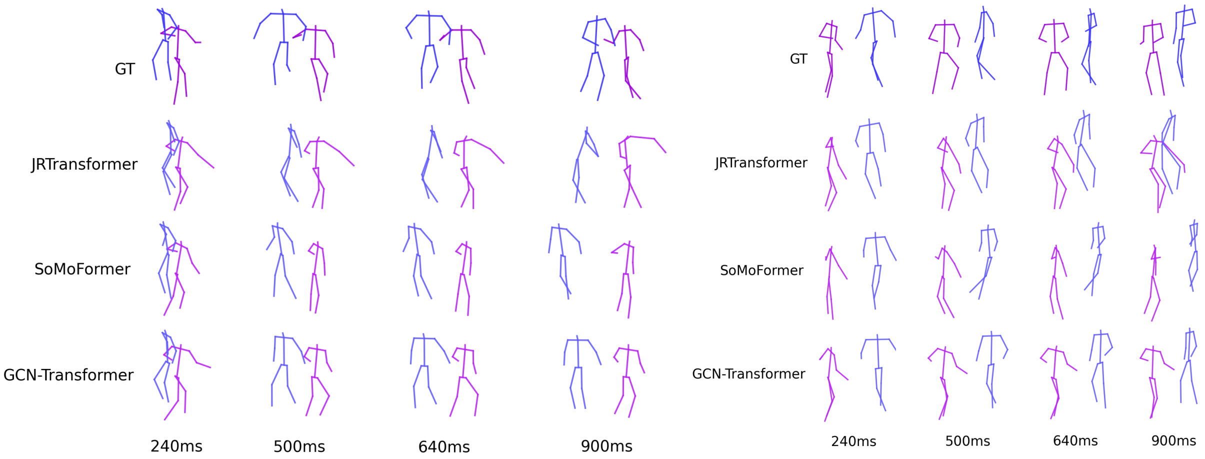
Table 2: Performance comparison on the SoMoF Benchmark test set featuring two-person scenes, using the VIM and MPJPE metrics, where lower values indicate better performances. Our proposed model, GCN-Transformer, achieves state-of-the-art results. The model marked with an asterisk (*) incorporated the validation dataset during training and currently leads the official SoMoF Benchmark leaderboard at `https://somof.stanford.edu`.

Method	**Metrics**											
	VIM						**MPJPE**					
	100 ms	**240 ms**	**500 ms**	**640 ms**	**900 ms**	**Overall**	**100 ms**	**240 ms**	**500 ms**	**640 ms**	**900 ms**	**Overall**
Zero Velocity	29.35	53.56	94.52	112.68	143.10	86.65	55.28	87.98	146.10	173.30	223.16	137.16
DViTA [31]	17.40	35.62	72.06	90.87	127.27	68.65	32.09	54.48	100.03	124.07	173.01	96.74
LTD [25]	18.07	34.88	68.16	85.07	116.83	64.60	33.57	55.21	97.57	119.58	163.69	93.92
TBIFormer [32]	17.62	34.67	67.50	84.01	116.38	64.03	32.26	53.65	95.61	117.22	160.99	91.94
MRT [43]	15.31	31.23	63.16	79.61	111.86	60.24	27.97	47.64	87.87	108.93	151.96	84.88
SocialTGCN [34]	12.84	27.41	58.12	74.59	107.19	56.03	23.10	40.24	76.91	96.89	139.01	75.23
JRTransformer [45]	11.17	25.73	56.50	73.19	106.87	54.69	18.44	35.38	72.26	92.42	135.12	70.73
MPFSIR [36]	11.57	25.37	54.04	69.65	101.13	52.35	20.31	35.69	69.58	88.36	128.37	68.46
Future Motion [42]	10.76	24.52	54.14	69.58	100.81	51.96	18.66	34.38	69.76	88.91	129.18	68.18
SoMoFormer [40]	10.45	23.10	49.76	64.30	**93.34**	48.19	17.63	32.42	63.86	81.20	**117.97**	62.62
GCN-Transformer (our)	**10.14**	**22.54**	**48.81**	**63.67**	94.94	**48.02**	**17.11**	**31.48**	**62.62**	**80.14**	118.14	**61.90**
GCN-Transformer * (our)	**9.82**	**21.80**	**46.61**	**60.88**	**91.95**	**46.21**	**16.41**	**30.36**	**60.31**	**76.94**	**113.36**	**59.48**

Best results in each column are highlighted in bold.

is not far off either, achieving this performance with only a fraction of parameters compared to others. Finally, the GCN-Transformer* showcases significantly superior results owing to its training with an integrated validation dataset. This variant currently leads the official SoMoF Benchmark leaderboard at `https://somof.stanford.edu`.

Figure 5 shows the predicted poses for two sequences from the SoMoF Benchmark test set, comparing the performance of the best-performing models, JRTransformer, SoMoFormer, and GCN-Transformer, with the ground truth (GT) also displayed for comparison. The figures reveal that both JRTransformer and SoMoFormer encounter difficulties in generating valid poses, often producing unrealistic joint configurations and movements. In contrast, the GCN-Transformer model demonstrates a clear advantage, consistently generating valid poses and realistic movements.



(a) (b)

Figure 5: The figure displays predicted poses on two example sequences from the SoMoF Benchmark test set for the best-performing models: JRTransformer, SoMoFormer, and GCN-Transformer, with GT representing the ground truth. Sequence (**a**) shows two people rotating around each other, while sequence (**b**) shows two people meeting and then walking together in the same direction. The visual comparison reveals that while JRTransformer and SoMoFormer struggle to create valid poses, the GCN-Transformer generates both valid poses and realistic movement.

6.5. Results on ExPI Dataset

The Extreme Pose Interaction (ExPI) dataset, described in [11], features two pairs of dancers engaging in 16 distinct extreme actions. These actions include aerial maneuvers, with the first seven being performed by both dancer couples. Subsequently, six additional aerials are executed by Couple 1, while the remaining three are carried out by Couple 2. Each action is repeated five times to capture variability, resulting in a collection of 115 sequences recorded at 25 frames per second (FPS) and 60,000 annotated 3D body poses.

Taking inspiration from the data partitioning outlined in [11], we designate all actions executed by Couple 2 as the training set and those performed by Couple 1 as the test set. This approach deviates slightly from the dataset's division presented by Guo et al. in [11], as we incorporate common actions performed by both couples and actions performed exclusively by one couple into the training set. This dataset split emulates both the Common action split and Unseen action split described in [11], consolidating them into a single split.

We employ a sliding-window technique with overlapping sequences to sample the training data, whereas the testing data are sampled sequentially without overlaps. Additionally, we downsample each sequence by selecting every other frame, reducing the original frames per second (FPS) from 25 to 12.5 FPS. Following the precedent set by the SoMoF Benchmark, we utilize 16 frames (equivalent to 1280 milliseconds) to predict the subsequent 14 frames (equivalent to 1080 milliseconds). Moreover, we apply a scaling factor of 0.39 to maintain consistency in person scale with the SoMoF Benchmark, the dataset on which the models are developed.

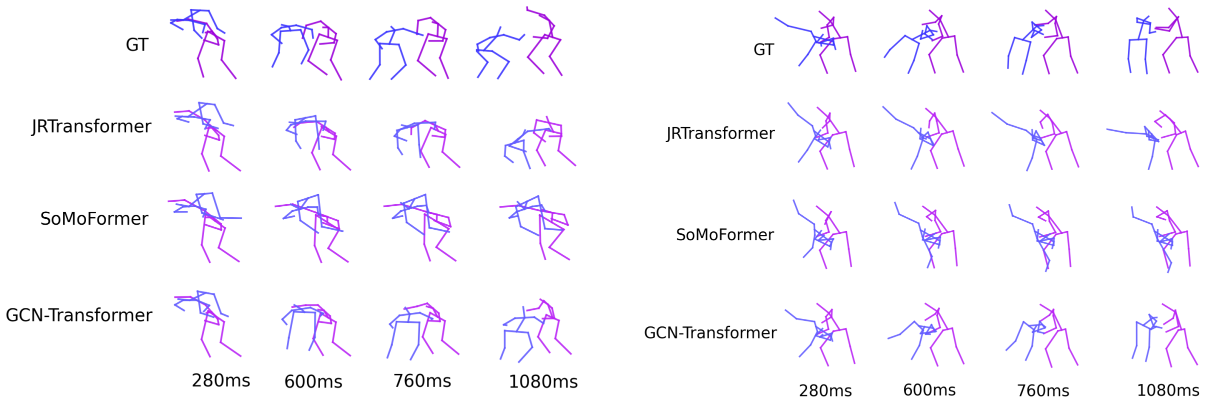
We conducted a comparative analysis of evaluated methods on the ExPI test set, as presented in Table 3, demonstrating that our model consistently achieves state-of-the-art results compared to competing models. The results on the ExPI dataset differ significantly from those on the SoMoF Benchmark dataset, revealing notable performance degradation in some of the previously strong models. SoMoFormer, a close competitor on the SoMoF Benchmark, performs substantially worse on the ExPI dataset, surpassed by JRTransformer and MPFSIR. This drop in performance highlights the model's sensitivity to different dataset characteristics. Similarly, the Future Motion model, which had proven to be a strong contender on the SoMoF Benchmark, is now outperformed by almost all other models. This indicates that the Future Motion model's performance is heavily influenced by the dataset's characteristics, showcasing its lack of robustness across diverse data scenarios. Interestingly, JRTransformer, which was not as competitive on the SoMoF Benchmark, emerges as a close competitor to GCN-Transformer on the ExPI dataset. Despite this, the proposed GCN-Transformer remains the clear winner across all time intervals, reaffirming its superior performance and generalizability.

Table 3: Performance comparison on the ExPI test set featuring two-person scenes using the VIM and MPJPE metrics, where lower values indicate better performance. Our proposed model, GCN-Transformer, achieves state-of-the-art results on both metrics.

Method	**Metrics**											
	VIM						**MPJPE**					
	120 ms	**280 ms**	**600 ms**	**760 ms**	**1080 ms**	**Overall**	**120 ms**	**280 ms**	**600 ms**	**760 ms**	**1080 ms**	**Overall**
Zero Velocity	25.61	48.66	84.39	97.41	118.10	74.84	46.16	74.66	124.32	145.22	181.33	114.34
DViTA [31]	15.44	35.27	74.43	91.44	119.51	67.22	28.31	51.63	100.85	124.49	167.98	94.65
LTD [25]	16.22	32.94	62.73	74.60	92.84	55.87	28.83	48.73	87.37	104.82	135.61	81.07
TBIFormer [32]	16.96	35.09	67.95	81.22	103.02	60.85	30.59	52.55	95.63	115.19	150.33	88.86
MRT [43]	15.32	32.07	61.84	74.04	94.59	55.57	27.79	47.91	87.01	104.80	137.22	80.95
SocialTGCN [34]	16.79	32.71	62.61	75.24	99.15	57.30	31.14	50.58	89.18	106.95	140.68	83.71
JRTransformer [45]	8.40	21.14	46.20	57.63	76.94	42.06	13.57	28.01	58.47	73.27	101.04	54.87
MPFSIR [36]	9.15	23.05	52.31	65.49	92.46	48.49	15.56	30.55	64.84	81.81	114.94	61.54
Future Motion [42]	16.94	34.83	68.45	83.33	108.03	62.32	30.51	52.37	96.06	116.88	156.04	90.37
SoMoFormer [40]	9.43	23.88	54.78	68.71	92.38	49.84	15.22	31.08	67.33	85.37	119.37	63.67
GCN-Transformer (our)	**8.32**	**20.84**	**44.56**	**54.81**	**74.66**	**40.64**	**13.37**	**27.63**	**57.27**	**71.25**	**97.71**	**53.45**

Best results in each column are highlighted in bold.

Figure 6 shows the predicted poses for two sequences from the ExPI test set, showcasing the performance of the best-performing models, JRTransformer, SoMoFormer, and GCN-Transformer, with the ground truth (GT) also displayed for comparison. The results highlight a significant distinction in model performance. JRTransformer and SoMoFormer struggle to generate valid movements, often defaulting to repeating the last known pose rather than predicting dynamic and realistic trajectories. In contrast, the GCN-Transformer model maintains the integrity of the poses and successfully predicts realistic and coherent movement patterns.



(a)

(b)

Figure 6: The figure displays predicted poses on two example sequences from the ExPI test set for the top-performing models, JRTransformer, SoMoFormer, and GCN-Transformer, with GT indicating the ground truth. Sequence (**a**) shows one person jumping off the shoulders of another, while sequence (**b**) shows one person performing a cartwheel assisted by another. The comparison illustrates that JRTransformer and SoMoFormer struggle with generating valid movements, often repeating the last known pose. In contrast, the GCN-Transformer demonstrates its capability to create realistic and dynamic movements.

6.6. Discussion of Comparative Advantages

While quantitative results establish the superior performance of our proposed GCN-Transformer model across all datasets, a deeper examination helps explain why it consistently outperforms prior approaches, particularly in interaction-heavy or socially complex scenarios. Methods such as MPFSIR and SoMoFormer primarily rely on dense fully connected layers or sequence-level attention, often treating individuals independently or relying on predefined assumptions about social structure. As a result, these models may struggle to encode fine-grained interaction dependencies or adapt to dynamically changing social configurations. In contrast, GCN-Transformer introduces a modular pipeline that combines learnable spatial reasoning (via the Spatial-GCN) with long-range temporal and spatial attention (via the Spatiotemporal Transformer Decoder), allowing it to reason jointly over the entire scene.

This design proves to be especially effective in datasets like ExPI, where highly coordinated motions (e.g., one person lifting or reacting to another) require the model to interpret subtle cues in one person's movement that inform another's. In these cases, baseline models often fail to capture the anticipatory or dependent nature of motion between individuals, producing disjointed or static predictions. We observe that GCN-Transformer maintains synchronization across subjects in such sequences and adapts more effectively to rapid transitions or uncommon poses, suggesting that its architectural integration of scene context and temporal dynamics enables stronger generalization.

Furthermore, the attention mechanisms in GCN-Transformer contribute to robustness in the presence of joint noise, as is sometimes the case in CMU-Mocap or MuPoTS-3D. Instead of relying uniformly on all joints or time steps, the model learns to attend selectively to informative joints and keyframes. This results in more stable predictions, even when input signals are imperfect, a scenario frequently encountered in real-world settings. Taken together, these architectural choices explain GCN-Transformer's consistently strong performance across diverse motion types, social contexts, and temporal horizons.

To assess the generalization ability and performance consistency of the evaluated models, we compute the percentage improvement over the Zero-Velocity baseline across all four datasets, as summarized in Table 4. This analysis uses the "Overall" MPJPE values reported in the earlier result tables, which reflect the average prediction error across the entire forecasting horizon. The percentage improvement is calculated using the following formula: $\text{Improvement} = (\text{Zero Velocity} - \text{Method})/\text{Zero Velocity} \times 100\%$. We use the Zero-Velocity model as a consistent reference point because it represents the most basic forecasting strategy, where the model simply repeats the last observed pose. Comparing raw MPJPE values across datasets is often not meaningful, as these values are strongly influenced by dataset-specific characteristics such as the amount of movement in the scenes, the difficulty of the motion patterns, and the prediction horizon. By instead reporting the improvement relative to the Zero-Velocity baseline, we obtain a normalized measure of model performance that enables more interpretable comparisons across different datasets.

For this analysis, we group the datasets into two categories based on the number of individuals in the scene and other shared characteristics. The CMU-Mocap and MuPoTS-3D datasets form a group of three-person scenes. These datasets both feature a three-second prediction horizon and relatively simple, low-motion sequences. The SoMoF Benchmark and ExPI datasets form a group of two-person scenes. These datasets have a shorter prediction horizon of approximately one second and include more active and socially complex motions, which generally result in higher forecasting errors.

Table 4: Percentage improvement over the Zero-Velocity baseline across all evaluated datasets, grouped by 3-person and 2-person scenes. Each value indicates the relative reduction in MPJPE, where higher values represent better performance. The table includes average improvements (Avg) and the standard deviation (Std) to reflect generalization consistency across datasets within each group. The best values in each group are shown in bold. The percentage improvement is computed as follows: Improvement = (Zero Velocity − Method) /Zero Velocity × 100%.

Method	**Percentage Improvements over Zero-Velocity Baseline (Based on Overall MPJPE Across All Datasets)**							
	2-Person Scenes				**3-Person Scenes**			
	SoMoF ↑	**ExPI ↑**	**Avg (%) ↑**	**Std (%) ↓**	**CMU-Mocap ↑**	**MuPoTS-3D ↑**	**Avg (%) ↑**	**Std (%) ↓**
Zero Velocity	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
DViTA [31]	29.47	17.22	23.34	6.12				
TBIFormer [32]	32.97	22.29	27.63	5.34				
LTD [25]	31.52	29.10	30.31	**1.21**	22.48	12.24	17.36	5.12
MRT [43]	38.12	29.21	33.66	4.45	13.84	-2.69	5.58	8.26
Future Motion [42]	50.30	20.96	35.63	14.67	20.49	1.19	10.84	9.65
SocialTGCN [34]	45.15	26.79	35.97	9.18				
MPFSIR [36]	50.09	46.18	48.14	1.96	23.03	15.82	19.42	3.61
SoMoFormer [40]	54.35	44.31	49.33	5.02	13.73	11.94	12.84	0.90
JRTransformer [45]	48.44	52.01	50.22	1.78	18.72	14.63	16.68	2.04
GCN-Transformer (our)	**56.64**	**53.26**	**54.95**	1.69	**28.46**	**28.66**	**28.56**	**0.1**

Best results in each column are highlighted in bold. Arrows next to the column names indicate the direction of better performance: ↑ means higher is better, ↓ means lower is better.

Table 4 reports the percentage improvement for each model on each dataset, along with the average improvement and standard deviation within each group. A higher average value indicates better overall performance, while a lower standard deviation reflects more consistent behavior across datasets within the same group. Our proposed model achieves the highest average improvement in both categories: 54.95% for the two-person scenes and 28.56% for the three-person scenes. Furthermore, the standard deviation of its improvements is low in both groups at 1.69% and 0.1%, respectively, suggesting that the model maintains consistent performance across diverse motion scenarios.

Other models show less consistent behavior. For example, Future Motion achieves relatively strong results on the SoMoF Benchmark but performs much worse on the ExPI dataset, resulting in a high standard deviation of 14.67 percent in the two-person group. This indicates that its performance is heavily dependent on the dataset's characteristics, limiting its generalizability. A similar pattern is observed with models such as SoMoFormer, SocialTGCN, DViTA, and TBIFormer, which exhibit noticeable variance in their performance across datasets. Even when these models do not rank the best in terms of absolute performance, their higher standard deviation values suggest limited robustness when applied to scenes with different motion dynamics or interaction complexities.

In contrast, two models that demonstrate better consistency in their generalization behavior are JRTransformer and MPFSIR. Both achieve relatively low standard deviation values across datasets in each group, indicating that their performance is more stable and less influenced by the specific characteristics of the test data. However, while they generalize more consistently, they still lag behind our proposed GCN-Transformer in terms of overall performance. Our proposed GCN-Transformer model achieves a percentage improvement over the Zero-Velocity model that is 4.7% higher than JRTransformer in the two-person group and 11.9% higher in the three-person group.

Overall, the normalized evaluation using improvements over the Zero-Velocity baseline offers a clearer and more meaningful interpretation of model performance across datasets with different characteristics. By comparing both average improvements and standard deviations, we can better understand each model's ability to generalize beyond a single dataset, revealing that GCN-Transformer achieves the best balance of performance and consistency among all evaluated models.

7. Ablation Study

We conducted an ablation study on GCN-Transformer to systematically assess the impact of different components and methods on the model's performance. This comprehensive analysis involved iteratively integrating various components and methods into the baseline model and evaluating performance at each stage. Initially, we established a baseline model comprising a Scene Module and Spatiotemporal Transformer Decoder. Subsequently, we extend the Spatiotemporal Attention Forecasting Module with Temporal-GCN, slightly enhancing model performance. Next, we introduced multi-person joint distance (MPJD) loss, further improving both short-term and long-term forecasting accuracy. Incorporating the Velocity Loss yielded a marginal improvement in overall performance, enhancing intra-sequence accuracy while slightly compromising short-term accuracy. Lastly, adding data augmentation significantly improved the model's performance across all evaluated time intervals, representing the most substantial improvement among all modifications. Table 5 presents the evaluation results of each model on VIM and MPJPE metrics, trained exclusively on the 3DPW training set and tested on the SoMoF Benchmark validation set.

Table 5: The ablation study results are derived from the SoMoF Benchmark validation set and presented in VIM (top) and MPJPE (bottom) metrics. The baseline model comprises Scene Module and the Spatiotemporal Transformer Decoder, with subsequent additions incrementally incorporated into the model. All models are trained solely on the SoMoF Benchmark training dataset, excluding AMASS.

Metric	Method	100 ms	240 ms	500 ms	640 ms	900 ms	Overall
VIM	Baseline	15.39	28.53	55.90	68.72	93.92	52.49
	+ Temporal-GCN	12.69	28.96	58.96	69.74	89.56	51.98
	+ MPJD loss	11.08	28.80	57.52	67.55	87.95	50.58
	+ Velocity loss	12.21	28.30	56.12	66.42	87.67	50.14
	+ Augmentation	**7.56**	**19.66**	**44.72**	**56.08**	**75.12**	**40.63**
MPJPE	Baseline	31.81	45.19	77.03	93.68	127.60	75.06
	+ Temporal-GCN	23.99	41.47	79.33	96.38	127.61	73.76
	+ MPJD loss	18.09	37.54	76.08	92.69	123.51	69.58
	+ Velocity loss	22.79	39.90	75.28	91.15	121.77	70.18
	+ Augmentation	**11.68**	**24.35**	**53.50**	**68.34**	**96.97**	**50.97**

Best results in each column are highlighted in bold.

8. FJPTE: Final Joint Position and Trajectory Error

The multitude of metrics available for pose forecasting complicates the evaluation process, as different metrics assess distinct aspects of the model's performance. Consequently, model rankings can vary significantly depending on the chosen evaluation metric, making it challenging to identify the optimal model for the task. To address this issue, we introduce a novel metric, Final Joint Position and Trajectory Error (FJPTE), designed to consolidate the diverse objectives of pose forecasting into a single comprehensive measure. Our metric aims to capture key goals of pose forecasting, including predicting the final (N-th frame) global position (e.g., pelvis) and the trajectory of global movement leading up to that position, as well as forecasting the final pose position without global movement and its accompanying trajectory. FJPTE tackles this challenge by independently evaluating four distinct components and aggregating their results: the error in the final global position (measured by Euclidean distance), the error of the global movement trajectory (measured using the Euclidean distance of the temporal differentiation of the root joint), the error in the final pose position excluding global movement (assessed using Euclidean distance), and the trajectory error of the pose position without global movement (measured using the Euclidean distance of the temporal differentiation for all pose joints). Through this comprehensive approach, FJPTE provides a holistic assessment of a model's performance, capturing its proficiency in capturing natural human motion dynamics and the validity of its predicted poses. An illustrative comparison of joint movement evaluation using our metric is presented in Figure 7.

Additionally, Figure 8 illustrates an example where FJPTE provides a more comprehensive evaluation than MPJPE or VIM. The example shows a predicted sequence where the global position is accurate, but the pose remains frozen or ghost-like, floating unnaturally through global space, an issue that is commonly seen in pose forecasting. Unlike MPJPE, which evaluates joint distances independently across time intervals, or VIM, which focuses solely on the final interval ($T = 30$), FJPTE comprises two key components: movement dynamics ($\text{FJPTE}_{\text{local}}$) and global position and trajectory ($\text{FJPTE}_{\text{global}}$). By

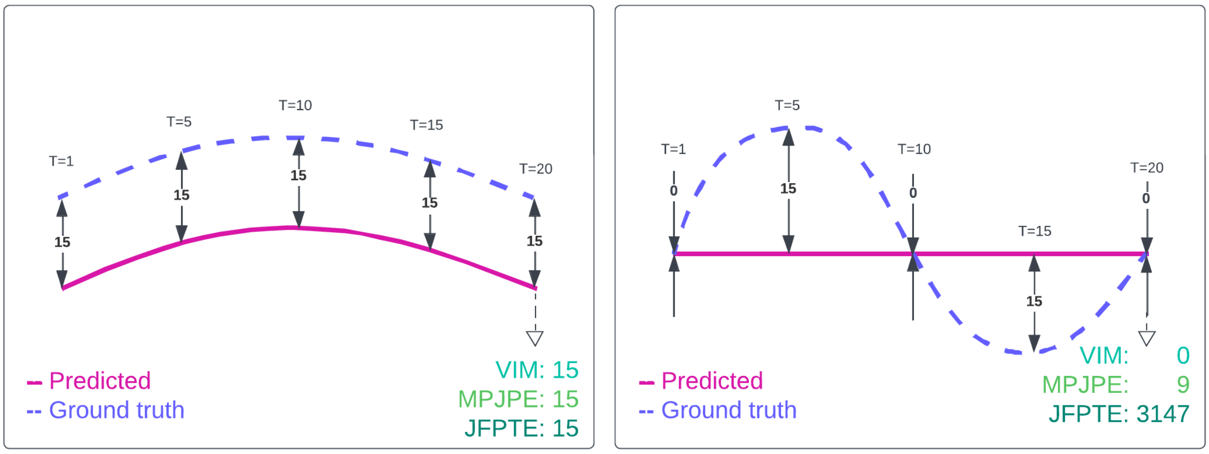


Figure 7: The figure illustrates an example of predicted (purple) and ground truth (blue) joint trajectories, where T represents the time interval, and the values between the trajectories indicate their distances at time T. When the trajectories are identical but have a slight offset, FJPTE yields the same results as MPJPE and VIM. However, when the trajectories diverge, the metrics produce significantly different results. MPJPE and FJPTE evaluate full joint trajectories, while VIM only evaluates the last time interval $T = 20$.

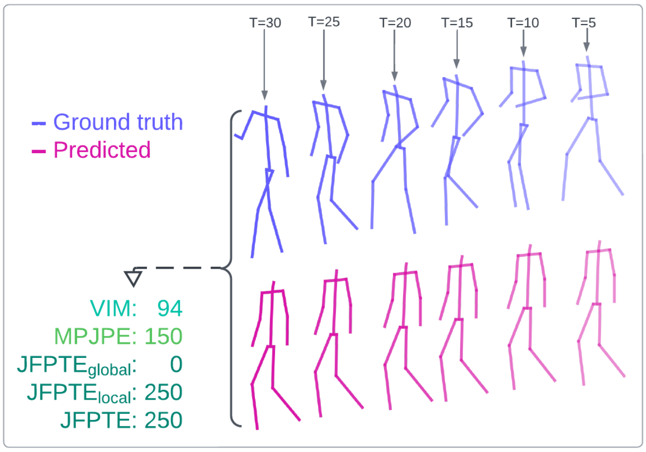


Figure 8: The figure illustrates an example of predicted (purple) and ground truth (blue) sequences of poses, with T representing the time interval. The predicted sequence demonstrates a scenario where the global position aligns well with the ground truth, but the pose remains frozen or ghost-like, floating through space, a common issue in pose forecasting. Metrics like MPJPE and VIM evaluate joint distances independently across time intervals, while the proposed FJPTE goes further by assessing joint trajectories and distinguishing between local (FJPTE$_{\text{local}}$) and global (FJPTE$_{\text{global}}$) movement. MPJPE and FJPTE evaluate the entire sequence, whereas VIM focuses only on the final time interval at $T = 30$.

breaking down errors into these components, FJPTE identifies whether a model struggles more with local movement dynamics or global trajectory alignment. Furthermore, by combining these errors, FJPTE enables a holistic evaluation and effective ranking of models based on their overall performance.

FJPTE is calculated as follows:

$$\begin{aligned}
E_{position}(\hat{y}, y) &= \frac{1}{J}\sum_{j=1}^{J}\|\hat{y}(j) - y(j)\|_2 \\
E_{trajectory}(\hat{Y}, Y) &= \frac{1}{T-1}\sum_{t=1}^{T-1} E_{position}(\hat{Y}^t - \hat{Y}^{t+1}, Y^t - Y^{t+1}) \\
E_{global}(\hat{Y}, Y) &= (E_{trajectory}(\hat{Y}_{\varphi_{pelvis}}, Y_{\varphi_{pelvis}}) + E_{position}(\hat{Y}^T_{\varphi_{pelvis}}, Y^T_{\varphi_{pelvis}})) \times 1000 \\
E_{local}(\hat{Y}, Y) &= (E_{trajectory}(\hat{Y} - \hat{Y}_{\varphi_{pelvis}}, Y - Y_{\varphi_{pelvis}}) + E_{position}(\hat{Y}^T - \hat{Y}^T_{\varphi_{pelvis}}, Y^T - Y^T_{\varphi_{pelvis}})) \times 1000 \\
E_{\text{FJPTE}}(\hat{Y}, Y) &= E_{global}(\hat{Y}, Y) + E_{local}(\hat{Y}, Y)
\end{aligned} \tag{11}$$

where $\hat{y}$ denotes the predicted sequence, while y denotes the ground truth sequence. The number of joints is denoted with J, while the number of time intervals is denoted with T. $\|\cdot\|_2$ denotes the Euclidean distance (L2 norm), and $\frac{1}{T-1}\sum_{t=1}^{T-1}$ represents the mean errors across all time intervals. $E_{global}(\hat{Y}, Y)$ represents the global position and trajectory error between predicted and ground truth sequences measured at the pelvis joint. $E_{local}(\hat{Y}, Y)$ represents the local movement dynamic errors between the predicted and ground truth sequences, excluding the pelvis joint and global movement. $E_{\text{FJPTE}}(\hat{Y}, Y)$ unifies local and global errors into a single metric.

We compared the models using the proposed $\text{FJPTE}_{\text{local}}$ and $\text{FJPTE}_{\text{global}}$ metrics on the SoMoF Benchmark test set and the reported results are shown in Table 6. The results demonstrate that GCN-Transformer significantly outperforms all other models on the $\text{FJPTE}_{\text{local}}$ metric. This underscores GCN-Transformer's superior ability to model human movement dynamics and interaction dynamics compared to the other models. While the overall performance hierarchy of the models remains consistent with evaluations using VIM and MPJPE metrics, LTD and JRTransformer exhibit slightly better performance in modeling movement dynamics than their immediate competitors TBI-Former and MPFSIR. When assessing the $\text{FJPTE}_{\text{global}}$ metric, GCN-Transformer shows a slight performance gap behind SoMoFormer in long-term forecasting, indicating that SoMoFormer has a marginal edge in predicting long-term global movements. Additionally, MPFSIR emerges as a notable performer, significantly outperforming its closest competitor, Future Motion, in forecasting global positions and trajectories.

Similarly, Table 7 presents the performance of evaluated models on the ExPI test set using the proposed $\text{FJPTE}_{\text{local}}$ and $\text{FJPTE}_{\text{global}}$ metrics. The results indicate that GCN-

Table 6: Comparison of performance on the SoMoF Benchmark test set using the proposed FJPTE metric, with lower values indicating superior performance. The table distinguishes between FJPTE$_{\text{local}}$ and FJPTE$_{\text{global}}$ errors, with FJPTE$_{\text{local}}$ representing movement dynamics errors and FJPTE$_{\text{global}}$ measuring global position and trajectory errors. The asterisk (*) denotes the model that integrated the validation dataset during training.

Method	**Components of Proposed FJPTE Metric**											
	Proposed FJPTE$_{\text{local}}$						**Proposed FJPTE$_{\text{global}}$**					
	100 ms	**240 ms**	**500 ms**	**640 ms**	**900 ms**	**Overall**	**100 ms**	**240 ms**	**500 ms**	**640 ms**	**900 ms**	**Overall**
Zero Velocity	65.36	97.18	142.35	158.79	178.72	128.48	91.12	146.51	241.69	284.08	363.52	225.38
DViTA [31]	55.15	91.84	147.91	168.07	194.29	131.45	47.60	81.35	162.46	212.71	319.11	164.65
LTD [25]	48.96	78.96	127.59	145.98	170.41	114.38	52.86	88.66	159.64	201.40	290.96	158.70
TBIFormer [32]	55.24	88.28	138.76	156.81	178.97	123.61	51.19	84.53	150.47	190.78	283.36	152.07
MRT [43]	56.38	90.59	143.17	162.19	186.11	127.69	46.74	77.70	147.95	189.65	279.84	148.37
SocialTGCN [34]	51.50	83.54	137.45	157.54	183.19	122.64	39.76	65.92	132.28	175.90	271.09	136.99
JRTransformer [45]	41.20	72.47	124.75	145.87	174.81	111.82	26.87	54.81	122.92	166.64	264.94	127.24
MPFSIR [36]	43.53	75.36	127.59	148.60	180.67	115.15	27.37	51.27	109.84	151.17	248.05	117.54
Future Motion [42]	42.74	72.22	122.18	140.77	165.83	108.75	31.04	54.72	117.86	158.93	249.45	122.40
SoMoFormer [40]	37.69	65.48	111.48	128.79	154.44	99.58	26.13	48.37	**104.01**	**139.66**	**217.92**	**107.22**
GCN-Transformer (our)	**37.22**	**63.78**	**109.06**	**126.12**	**152.72**	**97.78**	**24.35**	**47.42**	107.12	146.38	234.51	111.96
GCN-Transformer * (our)	**36.76**	**62.29**	**104.96**	**121.68**	**147.97**	**94.73**	**23.63**	**45.89**	**102.05**	**138.45**	228.94	107.79

Best results in each column are highlighted in bold.

Table 7: Comparison of performances on the ExPI test set using the proposed FJPTE metric, with lower values indicating superior performance. The table distinguishes between FJPTE$_{\text{local}}$ and FJPTE$_{\text{global}}$ errors, with FJPTE$_{\text{local}}$ representing movement dynamics errors and FJPTE$_{\text{global}}$ measuring global position and trajectory errors.

Method	**Components of Proposed FJPTE Metric**											
	Proposed FJPTE$_{\text{local}}$						**Proposed FJPTE$_{\text{global}}$**					
	120 ms	**280 ms**	**600 ms**	**760 ms**	**1080 ms**	**Overall**	**120 ms**	**280 ms**	**600 ms**	**760 ms**	**1080 ms**	**Overall**
Zero Velocity	76.63	119.52	182.09	205.19	240.31	164.75	79.80	127.56	201.88	230.77	280.05	184.01
DViTA [31]	56.91	101.25	176.21	206.20	252.27	158.57	45.58	83.58	164.19	202.36	271.01	153.34
LTD [25]	60.27	97.73	159.16	182.82	217.66	143.53	47.42	80.89	141.84	169.41	215.70	131.05
TBIFormer [32]	67.38	109.04	174.85	200.29	239.29	158.17	50.23	86.97	155.57	184.96	238.15	143.18
MRT [43]	65.77	107.77	173.87	199.12	236.71	156.65	43.80	75.45	133.75	162.58	214.24	125.96
SocialTGCN [34]	72.62	110.05	174.62	201.84	247.24	161.27	52.04	83.27	149.11	178.12	237.98	140.10
JRTransformer [45]	**37.98**	71.62	130.94	155.35	197.44	118.67	**26.21**	**52.63**	102.44	126.11	168.75	95.23
MPFSIR [36]	41.12	77.88	145.78	174.01	225.03	132.76	27.21	54.68	112.28	140.63	207.33	108.43
Future Motion [42]	64.87	105.26	175.12	206.69	247.48	159.88	48.70	86.51	160.21	197.70	270.41	152.71
SoMoFormer [40]	41.91	80.52	150.92	179.58	224.17	135.42	28.82	57.92	118.39	148.45	204.18	111.55
GCN-Transformer (our)	38.39	**71.60**	**125.41**	**146.24**	**181.17**	**112.56**	26.67	52.74	**100.23**	**122.83**	**172.73**	**95.04**

Best results in each column are highlighted in bold.

Transformer consistently outperforms all other models on the FJPTE$_{\text{local}}$ metric, except at the 120ms time interval, where JRTransformer marginally surpasses GCN-Transformer. Notably, SoMoFormer confirms that it is struggling with this dataset, while JRTransformer confirms it to be a strong contender. Another key observation is that LTD outperformed MRT on this metric compared to evaluations using the VIM and MPJPE metrics. When examining the FJPTE$_{\text{global}}$ metric, GCN-Transformer narrowly outperforms JR-Transformer, demonstrating a slight edge in overall performance despite JRTransformer's better short-term forecasting capabilities. SoMoFormer again shows a notable decline in performance, finishing behind both JRTransformer and MPFSIR. The overall performance hierarchy of the models on the ExPI dataset remains consistent with their evaluations using the VIM and MPJPE metrics.

These results indicate that models can perform well on VIM and MPJPE metrics by focusing on global movement or movement dynamics, as models typically excel in one of these areas but not both. In contrast, $\text{FJPTE}_{\text{local}}$ and $\text{FJPTE}_{\text{global}}$ provide a clear distinction, making it easier to identify the best-performing models for each specific area.

Table 8 presents a comprehensive evaluation of forecasting errors using the proposed FJPTE metric, which combines $\text{FJPTE}_{\text{local}}$ and $\text{FJPTE}_{\text{global}}$. On the SoMoF Benchmark test set, SoMoFormer emerges as the leading model, with only GCN-Transformer*, which included the validation set during training, surpassing its performance. Most models maintain a similar performance hierarchy, as seen with VIM and MPJPE evaluations, although LTD notably outperforms both TBIFormer and MRT.

Table 8: Comparison of performance on the SoMoF Benchmark test set (left) and the ExPI test set (right) using the proposed FJPTE metric, where lower values indicate better performance. The table presents FJPTE metric, combining $\text{FJPTE}_{\text{local}}$ and $\text{FJPTE}_{\text{global}}$ errors for a comprehensive performance evaluation. Our model achieves state-of-the-art results on the FJPTE metric. The asterisk (*) indicates models that integrated the validation dataset during training.

Method	**Proposed FJPTE Metric**											
	SoMoF Benchmark						**ExPI**					
	100 ms	**240 ms**	**500 ms**	**640 ms**	**900 ms**	**Overall**	**120 ms**	**280 ms**	**600 ms**	**760 ms**	**1080 ms**	**Overall**
Zero Velocity	156.48	243.69	384.04	442.87	542.24	353.86	156.43	247.07	383.97	435.95	520.36	348.76
DViTA [31]	102.75	173.20	310.36	380.78	513.40	296.10	102.48	184.82	340.40	408.56	523.29	311.91
LTD [25]	101.82	167.62	287.23	347.38	461.37	273.08	107.69	178.62	301.01	352.23	433.36	274.58
TBIFormer [32]	106.43	172.81	289.23	347.59	462.33	275.68	117.61	196.01	330.42	385.25	477.45	301.35
MRT [43]	103.11	168.29	291.12	351.84	465.95	276.06	109.58	183.22	307.63	361.70	450.95	282.62
SocialTGCN [34]	91.26	149.46	269.73	333.44	454.28	259.63	124.66	193.32	323.73	379.95	485.22	301.38
JRTransformer [45]	68.07	127.29	247.68	312.51	439.75	239.06	**64.19**	**124.25**	233.39	281.46	366.19	213.90
MPFSIR [36]	70.91	126.63	237.44	299.78	428.72	232.69	68.33	132.56	258.06	314.65	432.35	241.19
Future Motion [42]	73.78	126.94	240.04	299.70	415.28	231.15	113.57	191.77	335.33	404.39	517.89	312.59
SoMoFormer [40]	63.82	113.85	**215.50**	**268.45**	**372.35**	**206.79**	70.73	138.44	269.31	328.03	428.35	246.97
GCN-Transformer (our)	**61.57**	**111.21**	216.17	272.50	387.22	209.73	65.07	124.34	**225.64**	**269.07**	**353.90**	**207.60**
GCN-Transformer * (our)	**60.39**	**108.19**	**207.01**	**260.13**	**376.91**	**202.53**	-	-	-	-	-	-

Best results in each column are highlighted in bold.

In contrast, the ExPI test set results highlight GCN-Transformer as the top performer overall. While JRTransformer slightly outperforms GCN-Transformer in short-term forecasting, GCN-Transformer consistently delivers superior results across broader time intervals. The performance ranking of other models remains largely consistent with the VIM and MPJPE evaluations. However, LTD surpasses MRT, and DViTA outperforms Future Motion, making Future Motion the lowest-performing model on the ExPI dataset using FJPTE.

To summarize, the proposed FJPTE metric significantly enhances the evaluation of pose forecasting models by providing a more detailed analysis of movement dynamics alongside global position and trajectory errors. FJPTE delivers valuable insights into how accurately predictions capture realistic motion, as demonstrated in Figures 7 and 8. These examples highlight the metric's ability to pinpoint errors in movement dynamics versus global position and trajectory deviations, offering greater clarity during evaluation. This precision is particularly impactful in applications such as surveillance, animation, and autonomous systems, where natural movement dynamics are essential for effective human–robot interaction, motion tracking, and scene understanding. By quantifying both global alignment and detailed movement nuances, FJPTE that ensures models are rewarded for producing smooth, realistic motion. Furthermore, its focus on dynamics helps mitigate common issues such as ghost-like poses or unrealistic trajectories, boosting the robustness of models in real-world, dynamic scenarios.

9. Limitations

While the proposed GCN-Transformer demonstrates state-of-the-art performances in multi-person pose forecasting, it is not without limitations. A key drawback of the model lies in its size; GCN-Transformer has a large number of parameters (˜5.9 M), which makes it computationally expensive and memory-intensive compared to lighter models like MPFSIR (˜0.15 M). While MPFSIR performs nearly as well as state-of-the-art models with significantly fewer parameters, GCN-Transformer's parameter count is more comparable to its closest competitors, SoMoFormer (˜4.9 M) and JRTransformer (˜3.6 M), which mitigates this limitation to some extent.

Beyond the parameter count, the model's computational complexity is primarily driven by the Spatiotemporal Transformer Decoder. This component scales with $\mathcal{O}(N \cdot T^2 \cdot d)$, where N is the number of individuals, T is the temporal sequence length, and d the embedding dimension. The quadratic time complexity with respect to sequence lengths is typical relative to the self-attention mechanism. The Spatial-GCN and Temporal-GCN modules are less intensive, with complexities of $\mathcal{O}(N \cdot J^2)$ and $\mathcal{O}(T \cdot J^2)$, respectively, where J is the number of joints.

A more significant limitation, which is shared by GCN-Transformer and other models in the field, is the inability to forecast movements that are not represented in the training dataset. When encountering novel movements, models tend to repeat the last observed poses, resulting in frozen or static sequences. Figure 9 illustrates examples from the SoMoF and ExPI datasets, where unseen movements lead to poor forecasts. In such cases, the model fails to generalize effectively, underscoring the importance of diverse and representative training datasets to address this issue.

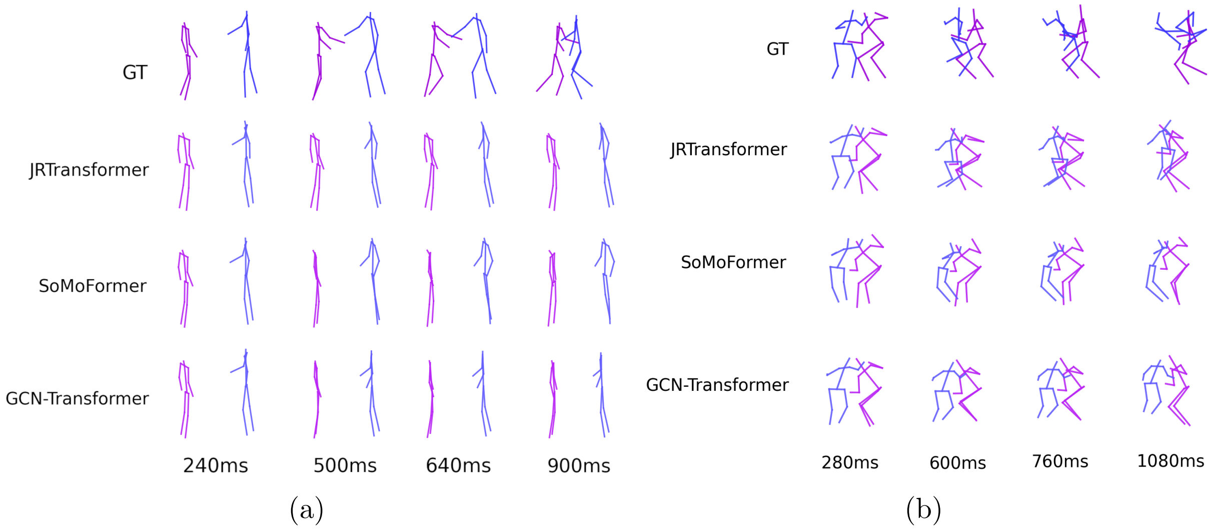


Figure 9: Examples from the SoMoF (**a**) and ExPI (**b**) dataset illustrating the limitations of GCN-Transformer and other models in forecasting movements not observed during training. In the SoMoF sequence (**a**), one individual approaches another, initiating a complex movement where the two prepare to spin around each other in a dance-like motion. In the ExPI sequence (**b**), two individuals perform a complex action where one lifts the other overhead to execute a backflip. Due to the absence of such intricate interactions in the training data, the models struggle to predict the dynamic sequences and instead produce a static forecast, merely repeating the last observed poses of the individuals and failing to capture the expected motion.

Another limitation of GCN-Transformer is the complexity of training due to its reliance on strong augmentations. While these augmentations improve generalization, they also necessitate longer training cycles and careful hyperparameter tuning to stabilize learning. Furthermore, despite its ability to capture interactions and dependencies between individuals, the model may struggle in scenes with highly intricate or unusual social dynamics, where interactions are more ambiguous or rare.

Lastly, the evaluation of model performance still heavily relies on benchmark datasets, which may not fully capture the diversity and variability of real-world scenarios. Consequently, there remains room for improvement in assessing and optimizing model robustness for broader applications.

These limitations provide multiple promising directions for future research. One direction is the development of more efficient, lightweight architectures that retain the ability to model complex interaction dynamics, making them suitable for deployment in real-time or resource-constrained environments. Another avenue is improving generalization relative to unseen or rare motions, which could be addressed through techniques such as data-driven motion priors, transfer learning, or motion synthesis via generative models. To support this, the field would greatly benefit from the creation of new multi-person pose forecasting datasets that include more diverse, socially rich, and dynamic interactions. Current datasets are limited in scope and variety, and expanding this benchmark space would allow models to better reflect real-world challenges and enhance their robustness in varied applications. Furthermore, improving training efficiency through adaptive enhancement strategies or self-supervised pre-training could reduce computational costs while maintaining performance.

A further limitation is that, like most multi-person forecasting models, the GCN-Transformer is trained for a fixed number of individuals per scene (e.g., two-person scenarios). When applied to datasets with a different number of individuals, minor modifications to the preprocessing pipeline are required: for example, artificially creating new sub-scenes by selecting two individuals out of a three-person scene. This design constraint is shared by all other models except SoMoFormer, which supports direct prediction for an arbitrary number of individuals without additional adjustments. Addressing this flexibility limitation without sacrificing performance in future model designs could broaden its applicability to real-world settings, where the number of individuals in a scene may vary.

10. Conclusions

In conclusion, this paper introduces GCN-Transformer, a novel model for multi-person pose forecasting that leverages the synergies of Graph Convolutional Network and Transformer architectures. We conducted a thorough evaluation of GCN-Transformer alongside other state-of-the-art models, presenting results on the CMU-Mocap, MuPoTS-3D, SoMoF Benchmark, and ExPI datasets using the VIM and MPJPE metrics. The results on the CMU-Mocap and MuPoTS-3D datasets, which feature three-person interaction scenes with generally simpler and lower interaction motions compared to ExPI, show that our model consistently achieves state-of-the-art performance across both datasets, demonstrating its robustness across varying levels of interaction complexities and different numbers of people in the scene. The results on the SoMoF Benchmark should be cautiously interpreted due to the dataset's inherent randomness, attributed to the sequences recorded with a moving camera. This introduces complexities as models must predict human and camera movements, often perceived as erratic. To mitigate this, we additionally evaluated all models on the ExPI dataset, featuring challenging actions performed by two couples without camera movement. Conclusively, GCN-Transformer consistently outperforms existing state-of-the-art models on all datasets.

Furthermore, we propose a novel evaluation metric, FJPTE, which comprehensively assesses pose forecasting errors by accounting for both local movement dynamics ($\text{FJPTE}_{\text{local}}$) and global movement ($\text{FJPTE}_{\text{global}}$). These components are computed based on errors at the final position and along the trajectory leading up to that point. Our evaluation of all models using FJPTE reveals that GCN-Transformer excels in capturing both intricate movement dynamics and accurate global position trajectory, where it consistently achieves state-of-the-art results.

The superior performance of GCN-Transformer can be attributed to its hybrid architecture that allows the model to capture fine-grained spatial dependencies within individuals while also modeling long-range temporal and social interactions across people in the scene. The attention mechanism further enhances robustness by enabling the model to focus dynamically on relevant joints and individuals, which is particularly effective in handling socially complex behaviors, such as those found in the ExPI dataset. As a result, GCN-Transformer demonstrates strong generalization across varying motion types and interaction intensities, outperforming prior approaches that lack either spatial specificity or long-term temporal modeling capacity.

Overall, the success of the proposed GCN-Transformer underscores its potential to drive the field of multi-person pose forecasting, with promising applications in human–computer interaction, sports analysis, and augmented reality. Beyond its empirical performance, this work introduces a modular modeling and evaluation perspective for interaction-rich forecasting, where generating socially coherent pose sequences and evaluating them using trajectory and position-aware metrics are addressed together. These design choices contribute toward advancing more expressive, generalizable, and testable architectures for multi-person pose forecasting. As future work, we aim to explore further enhancements for GCN-Transformer's architecture, including the integration of activity recognition to aid in pose forecasting, and we will investigate its applicability to real-world scenarios.

References

[1] Vida Adeli et al. "Tripod: Human trajectory and pose dynamics forecasting in the wild". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 13390–13400.

[2] James Atwood and Don Towsley. "Diffusion-convolutional neural networks". In: *Advances in neural information processing systems* 29 (2016).

[3] Görkay Aydemir, Adil Kaan Akan, and Fatma Güney. "Adapt: Efficient multi-agent trajectory prediction with adaptation". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 8295–8305.

[4] Arij Bouazizi et al. "MotionMixer: MLP-based 3D Human Body Pose Forecasting". In: *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22, Vienna, Austria, 23–29 July 2022*. July 2022, pp. 791–798.

[5] Joan Bruna et al. "Spectral networks and locally connected networks on graphs". In: *arXiv preprint arXiv:1312.6203* (2013).

[6] *Carnegie Mellon University Motion Capture Database*. (accessed on 2 February 2025). URL: `https://paperswithcode.com/dataset/cmu-motion-capture`.

[7] Hongren Cheng et al. "Joint graph convolution networks and transformer for human pose estimation in sports technique analysis". In: *Journal of King Saud University - Computer and Information Sciences* 35.10 (2023), p. 101819.

[8] Hsu-kuang Chiu et al. "Action-agnostic human pose forecasting". In: *2019 IEEE winter conference on applications of computer vision (WACV)*. IEEE. 2019, pp. 1423–1432.

[9] Qiongjie Cui, Huaijiang Sun, and Fei Yang. "Learning dynamic relationships for 3d human motion prediction". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 6519–6527.

[10] Wen Guo et al. "Back to mlp: A simple baseline for human motion prediction". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2023, pp. 4809–4819.

[11] Wen Guo et al. "Multi-person extreme motion prediction". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 13053–13064.

[12] Xiaxia He et al. "TEA-GCN: Transformer-Enhanced Adaptive Graph Convolutional Network for Traffic Flow Forecasting". In: *Sensors* 24.21 (2024). ISSN: 1424-8220. DOI: `10.3390/s24217086`.

[13] Xiao Hu et al. "GCN-Transformer-Based Spatio-Temporal Load Forecasting for EV Battery Swapping Stations under Differential Couplings". In: *Electronics* 13.17 (2024), p. 3401.

[14] Xinxin Huang et al. "Sensor-Based Wearable Systems for Monitoring Human Motion and Posture: A Review". In: *Sensors* 23.22 (2023). ISSN: 1424-8220. DOI: `10.3390/s23229047`.

[15] Yingfan Huang et al. "Stgat: Modeling spatial-temporal interactions for human trajectory prediction". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 6272–6281.

[16] Ismael Espinoza Jaramillo et al. "Human Activity Prediction Based on Forecasted IMU Activity Signals by Sequence-to-Sequence Deep Neural Networks". In: *Sensors* 23.14 (2023). ISSN: 1424-8220. DOI: `10.3390/s23146491`.

[17] Jaewoo Jeong, Daehee Park, and Kuk-Jin Yoon. "Multi-agent Long-term 3D Human Pose Forecasting via Interaction-aware Trajectory Conditioning". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024, pp. 1617–1628.

[18] Jiaming Jiang et al. "A Survey of Deep Learning-Based Pedestrian Trajectory Prediction: Challenges and Solutions". In: *Sensors* 25.3 (2025). ISSN: 1424-8220. DOI: `10.3390/s25030957`.

[19] Thomas N Kipf and Max Welling. "Semi-supervised classification with graph convolutional networks". In: *arXiv preprint arXiv:1609.02907* (2016).

[20] Lihuan Li, Maurice Pagnucco, and Yang Song. "Graph-based spatial transformer with memory replay for multi-future pedestrian trajectory prediction". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 2231–2241.

[21] Wen-Nung Lie and Veasna Vann. "Estimating a 3D Human Skeleton from a Single RGB Image by Fusing Predicted Depths from Multiple Virtual Viewpoints". In: *Sensors* 24.24 (2024). ISSN: 1424-8220. DOI: `10.3390/s24248017`.

[22] Naureen Mahmood et al. "AMASS: Archive of motion capture as surface shapes". In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 5442–5451.

[23] Wei Mao, Richard I Hartley, Mathieu Salzmann, et al. "Contact-aware human motion forecasting". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 7356–7367.

[24] Wei Mao, Miaomiao Liu, and Mathieu Salzmann. "History repeats itself: Human motion prediction via motion attention". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*. Springer. 2020, pp. 474–489.

[25] Wei Mao et al. "Learning trajectory dependencies for human motion prediction". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 9489–9497.

[26] Angel Mart'inez-Gonz'alez, Michael Villamizar, and Jean-Marc Odobez. "Pose transformers (potr): Human motion prediction with non-autoregressive transformers". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 2276–2284.

[27] Omar Medjaouri and Kevin Desai. "Hr-stan: High-resolution spatio-temporal attention network for 3d human motion prediction". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 2540–2549.

[28] Dushyant Mehta et al. "Single-shot multi-person 3d pose estimation from monocular rgb". In: *2018 International Conference on 3D Vision (3DV)*. IEEE. 2018, pp. 120–130.

[29] Matteo Menolotto et al. "Motion Capture Technology in Industrial Applications: A Systematic Review". In: *Sensors* 20.19 (2020). ISSN: 1424-8220. DOI: `10.3390/s20195687`.

[30] L. Minh Dang et al. "Sensor-based and vision-based human activity recognition: A comprehensive survey". In: *Pattern Recognition* 108 (2020), p. 107561. ISSN: 0031-3203. DOI: `https://doi.org/10.1016/j.patcog.2020.107561`.

[31] Behnam Parsaeifard et al. "Learning decoupled representations for human pose forecasting". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 2294–2303.

[32] Xiaogang Peng, Siyuan Mao, and Zizhao Wu. "Trajectory-aware body interaction transformer for multi-person pose forecasting". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 17121–17130.

[33] Xiaogang Peng et al. "SoMoFormer: Social-Aware Motion Transformer for Multi-Person Motion Prediction". In: *arXiv preprint arXiv:2208.09224* (2022).

[34] Xiaogang Peng et al. "The MI-Motion Dataset and Benchmark for 3D Multi-Person Motion Prediction". In: *arXiv preprint arXiv:2306.13566* (2023).

[35] Muhammad Rameez Ur Rahman et al. "Best Practices for 2-Body Pose Forecasting". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. June 2023, pp. 3614–3624.

[36] Romeo Šajina and Marina Ivasic-Kos. “MPFSIR: An Effective Multi-Person Pose Forecasting Model With Social Interaction Recognition”. In: *IEEE Access* 11 (2023), pp. 84822–84833. DOI: 10.1109/ACCESS.2023.3303018.

[37] Romeo Šajina and Marina Ivašić-Kos. “3D Pose Estimation and Tracking in Handball Actions Using a Monocular Camera”. In: *Journal of Imaging* 8.11 (2022), p. 308. DOI: 10.3390/jimaging8110308.

[38] Julian Tanke et al. “Social diffusion: Long-term multiple human motion anticipation”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 9601–9611.

[39] A Vaswani. “Attention is all you need”. In: *Advances in Neural Information Processing Systems* (2017).

[40] Edward Vendrow et al. “SoMoFormer: Multi-Person Pose Forecasting with Transformers”. In: *arXiv preprint arXiv:2208.14023* (2022).

[41] Timo Von Marcard et al. “Recovering accurate 3d human pose in the wild using imus and a moving camera”. In: *Proceedings of the European conference on computer vision (ECCV)*. 2018, pp. 601–617.

[42] Chenxi Wang et al. “Simple baseline for single human motion forecasting”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 2260–2265.

[43] Jiashun Wang et al. “Multi-person 3D motion prediction with multi-range transformers”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 6036–6049.

[44] Lang Xiong et al. “Dynamic adaptive graph convolutional transformer with broad learning system for multi-dimensional chaotic time series prediction”. In: *Applied Soft Computing* 157 (2024), p. 111516.

[45] Qingyao Xu et al. “Joint-Relation Transformer for Multi-Person Motion Prediction”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 9816–9826.

[46] Sirui Xu, Yu-Xiong Wang, and Liangyan Gui. “Stochastic multi-person 3d motion forecasting”. In: *The Eleventh International Conference on Learning Representations*. 2023.

[47] Kai Zhai et al. “Hopfir: Hop-wise graphformer with intragroup joint refinement for 3d human pose estimation”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 14985–14995.

[48] Chongyang Zhong et al. “Spatio-temporal gating-adjacency gcn for human motion prediction”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 6447–6456.

C. 3D Pose Estimation and Tracking in Handball Actions Using a Monocular Camera

This work was published as: Romeo Šajina, Marina Ivašić-Kos, 2022. 3D Pose Estimation and Tracking in Handball Actions Using a Monocular Camera. Journal of Imaging, Volume 8, ISSN 2313-433X, DOI: 10.3390/jimaging8110308

https://www.mdpi.com/2313-433X/8/11/308

1. Introduction

Human Pose Estimation (HPE) is a subfield of computer vision that aims to recognise the joints and skeleton of the human body in an image or video so that, based on these keypoints, a person's position and orientation can be analysed, movements can be monitored and compared, motion and positions can be tracked, and various insights into the person's activities can be drawn. It is a rapidly growing research area that has applications in various industries, including sports, dance, computer gaming, and healthcare. Some common use cases include action recognition and tracking, augmented reality experiences, animation, gaming, etc.

Today, almost all sports, both professional and recreational, rely heavily on data analytics and monitoring of athletes' performance using various sensors and cameras in cell phone applications, smartwatches, and other human performance monitoring devices. As a result of an athletic activity, large amounts of recorded material are generated, which must be analysed to be useful. Analysing the material, especially the performance and movement of each athlete, is a tedious task that requires the expertise of kinesiologists, physical therapists, and sports experts, as well as many resources, and is therefore available only to clubs and high-level athletes.

The use of pose estimation methods can facilitate and speed up the process of analysing the athletes' performance, especially in monitoring their movements, comparing techniques, and evaluating the proper execution of activities, and it can be made available to young athletes, small clubs, and recreational players. However, for automatic pose detection to be useful and usable in maintaining and improving physical activity and achieving the desired fitness, it must achieve high accuracy under real-world conditions and operate in real time. Useful and promising results in body posture detection and estimation were achieved in the era of Deep Learning, when deep convolutional neural networks were used to estimate the positions of keypoints on the body. One of the first deep neural networks that achieved promising results in human pose detection was DeepPose [88], which showed that deep neural networks could model invisible joints and perform much better under non-ideal conditions with occlusions. These results reversed the trend and paved the way for further research relying primarily on deep neural networks for pose estimation.

Currently, the best results with deep learning models are obtained for individual sports and stationary exercises such as yoga or Pilates. However, the goal is to learn models for more complex scenarios, including more complex actions, non-standard poses, players, and team sports.

However, to analyse the execution of an action, in most cases one image or one frame is not sufficient, but it is necessary to track the person and his activity over a certain time sequence and a series of frames. The case where multiple objects appear on a scene that is observed over a period of time, taking into account the change in position of each object in the video sequence, is called multiple object tracking (MOT). Tracking provides the best results when the objects move uniformly, in the same direction, and without occlusion. However, this is usually not a realistic scenario, especially in complex scenes such as sporting events where a large number of players are being tracked, moving rapidly, changing their direction and speed, as well as their position and distance from the camera and the activity they are performing. In such dynamic scenes, tracking multiple objects remains a major challenge. However, thanks to improved object and skeleton detectors and computer power, pose tracking with object detection has become the leading paradigm for MOT.

In this paper, we present the current state of research on HPE based on Deep Learning, which can be useful for position estimation, tracking, action recognition, and action comparison of players in a dynamic team sport such as handball. First, we analyzed and compared related research that deals with tracking methods for observing and analyzing the motion of individuals based on the skeleton as a representation of a person. We also provide a list of publicly available datasets that can be used to learn person pose models. In addition, we test and evaluate 12 popular 2-stage models for 3D HPE with a monocular camera trained on public and custom datasets in unseen environments and scenes such as handball jump shots to assess the robustness and applicability of the methods in a new and unfamiliar sports domain and environment.

Finally, to improve the performance of pose estimation methods for action recognition and comparison tasks where a sequence of aligned pose detection is a prerequisite, we have defined a method-independent pipeline that includes smoothing (to remove noise from the prediction) and retargeting (to standardize the distance between keypoints before pose estimation), and experimentally tested the effects on performance improvement on different models. The procedure for obtaining a sequence of poses is shown in Figure 1. Human pose estimation is used to create keypoints of the human skeleton, and object tracking is used to group poses collected through the sequence of frames into a series of poses corresponding to an activity.

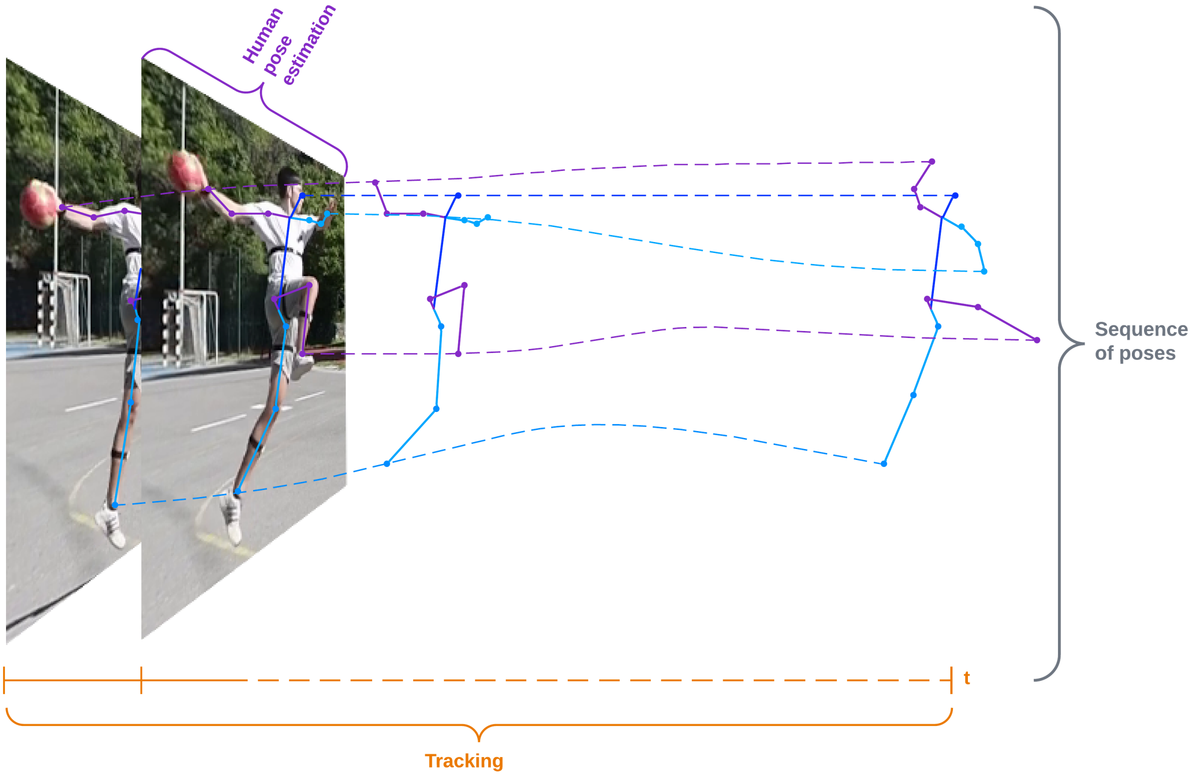


Figure 1: Creating a sequence of poses using human pose estimation to produce human skeleton keypoints and object tracking for grouping collected poses across frames (t) into a single sequence of poses.

The contributions of this work can be summarized as follows:

- Overview of the methods, models, and algorithms used in pose estimation and tracking with a monocular camera;
- Evaluation of 12 selected 2-stage pose estimation models based on deep learning in a 3D pose estimation task with a monocular camera trained on public and custom datasets to test the robustness of the model in the new sports domain and environment;
- Proposed method-independent pipeline for smoothing and retargeting 3D pose estimation sequences for action recognition and comparison tasks where an aligned pose sequence is a prerequisite;

- Evaluation of the prediction performance of 12 selected 2-stage deep learning models on a 3D pose estimation task when the proposed method-independent pipeline is used to smooth the estimated 3D sequences;
- Evaluation of selected 5 state-of-the-art tracking methods to assess the robustness of the models in an unseen sports environment.

The rest of the paper is organized as follows: Section 2 describes the methods for pose estimation using a monocular camera, as well as various approaches for improving the accuracy of pose estimation, methods for standardizing poses, and data sets for pose estimation. Section 3 describes tracking algorithms that allow detected poses to be linked in sequences in a multi-person environment along with tracking datasets, while Sections 4 and 5 describe the evaluation of 3D pose estimation pipelines and tracking methods on public and custom datasets. The paper ends with a conclusion and discussion.

2. Pose Estimation

Estimation of human posture is essentially a matter of identifying and classifying the joints of the human body so that a skeleton can represent the human body in such a way that each joint (arm, head, torso, etc.) important in representing a person's posture is given as a set of coordinates and is connected to the adjacent keypoints. Typically, posture estimation is based on the determination of 18 standard keypoints representing important body parts and joints, as shown in Figure 2.

The goal of HPE is to design the representation of the human body in such a way that geometric information and information about the movement of the human body can be understood, further processed, and applied to specific tasks. At various stages of development, three different representations of the human body were considered: the skeleton-based model, the contour-based model, and the volume-based model. Today, however, the skeleton-based representation is predominant.

The evaluation of the human position can be done in the plane or in space, and 2D or 3D methods are therefore used to predict and represent the position of the human body.

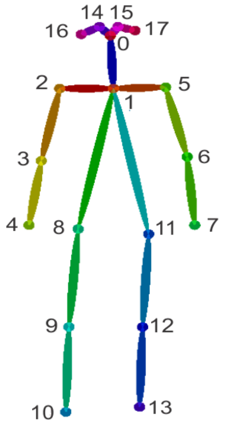


Figure 2: Standard 18-person keypoints in pose estimation.

Traditional approaches use 2D models to estimate the 2D position or spatial location of human body points from images and video frames and rely on hand-crafted low-level features such as Gaussian-oriented histograms (HOG), contours, colour histograms, and machine learning methods such as Random Forest to determine joints in the human body. However, all traditional methods have one problem: they only work when all body parts are visible and clearly represented. These problems are largely overcome by the use of deep neural networks, which can learn complex features and achieve higher accuracy when enough data is available. As a result, they are now predominantly used for all computer vision tasks, including human pose estimation.

Toshev and Szegedy [88] were the first to use a Deep Convolutional Neural Network (CNN) for the human pose estimation problem. They developed the DeepPose model, which yielded promising results and showed that the network can model poses with hidden and occluded joints. They also focused on further research on approaches based on Deep Learning.

Several Deep Learning based approaches have been introduced to achieve better pose estimation results, which according to Ref. [19] can be generally divided into two categories: Single Person Approaches and Multiple Person Approaches, as shown in Figure 3.

In the single-person approach, the pose of a person in an image is recognised based on the position of the person and an implicit number of keypoints, so it is essentially a regression problem. The multi-person approach, on the other hand, aims to solve an unconstrained problem, since the number and positions of the persons in the image are

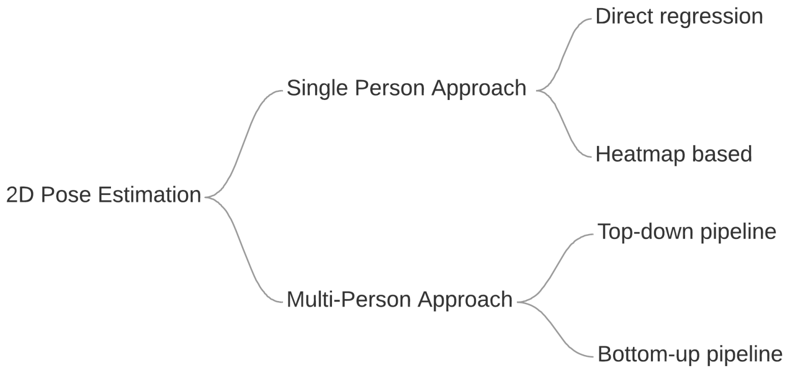


Figure 3: Taxonomy of pose estimation approaches based on Ref. [19].

unknown.

2.1. The Single-Person Approach

The single-person approach is divided into two frameworks based on the keypoint prediction method: direct regression of keypoints from features (i.e., direct regression-based framework) or generating heatmaps and inferring keypoints via heatmap (i.e., heatmap-based framework).

2.1.1. Direct Regression-Based Framework

Toshev and Szegedy presented DeepPose in Ref. [88], where they proposed a cascaded Deep Neural Network (DNN) regressor for predicting keypoints directly from feature maps. The model follows a simple architecture with convolutional layers followed by dense layers that generate (x, y) values for keypoints. Carreira et al. [12] proposed a method to iteratively refine the model output by feeding back error predictions, resulting in a significant increase in accuracy. Luvizon et al. [52] proposed a soft Argmax function to directly convert feature maps into common coordinates using a keypoint error distance-based loss function and a context-based structure to achieve competitive results compared to a heatmap-based framework. Sun et al. [85] proposed a structure-aware regression approach using a reparametrized pose representation with bones instead of joints. Bones are easier to recognize because they are more primitive and stable, cover a larger area, and are more robust to occlusion, making them easier to learn than joints. The presented results show an improvement in performance over previous direct regression-based systems, but are also very competitive with heatmap based systems.

2.1.2. Heatmap-Based Framework

Instead of predicting keypoints directly, an alternative approach can be used to create heat maps of all keypoints within the image. Then, additional methods are used to construct the final stick figure, as shown in Figure 4.

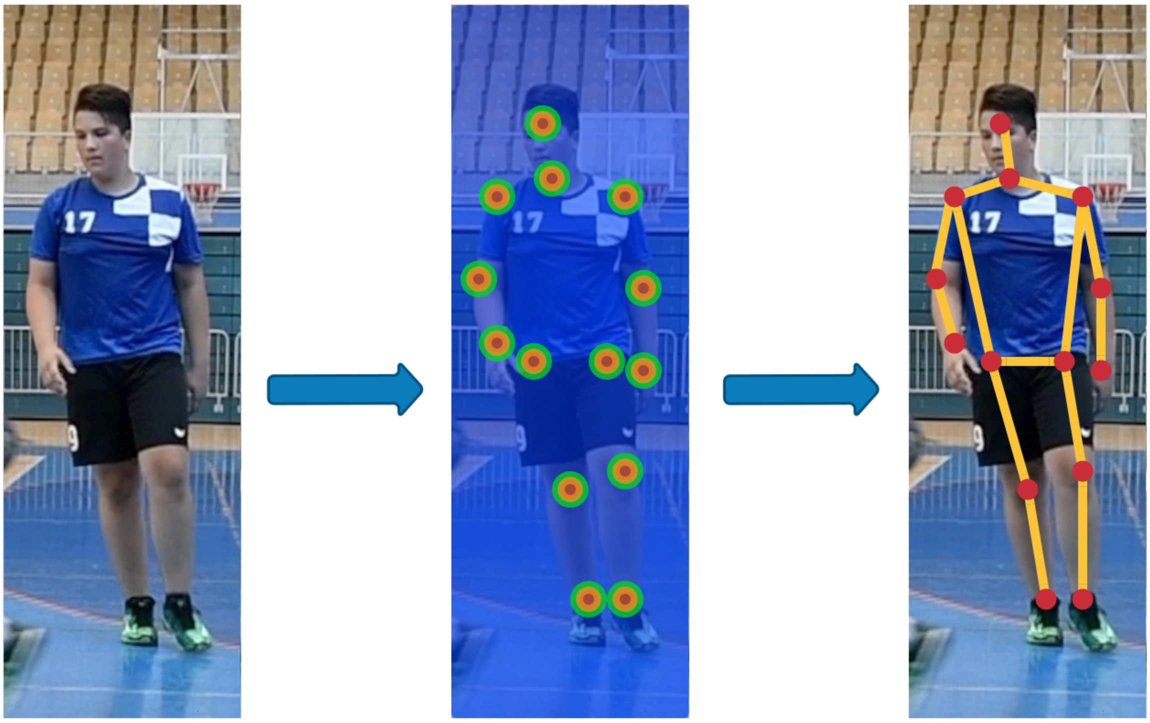


Figure 4: Heatmap poses estimation. It starts by creating heatmaps of all keypoints within the image, and then additional methods are used to construct the final stick figure.

Chen and Yuille [14] proposed a graphical model with pairwise relations for adaptive use of local image measurements. Local image measurements can be used both to detect joints and to predict the relationships between joints. Newell et al. [58] designed a ”stacked hourglass” network closely related to the encoder-decoder architecture, based on the sequential steps of pooling and upsampling before generating the final prediction set. They showed that repeated bottom-up and top-down processing with intermediary supervision is critical for improving performance in human pose detection. Later research commonly used a stacked hourglass network. Adversarial PoseNet [16] uses a discriminator to distinguish between real and fake poses, which are usually the result of a complex scene or occlusions. The discriminator learns the structure of the stick figure, and can thus decide whether a pose is real (reasonable as a body shape) or fake. The discriminator results are then used to further train the model for pose estimation. Chu et al. [18] use a multi-context attention mechanism that focuses on the global consistency of the entire human body and the description of different body parts. In addition, they introduce a novel Hourglass Residual Unit to increase the receptive field of the network. Martinez et al. [53] introduce a basis for 3D estimation of human poses that uses an hourglass network to predict 2D keypoints, which are then fed into a simple feed-forward network that provides a prediction of 3D keypoints.

2.2. The Multi-Person Approach

The multi-person approach is more complex because the number and positions of people in the image are not given. Therefore, the system must recognise keypoints and assemble an unknown number of people. Two pipelines have been proposed to deal with this task: a top-down pipeline and a bottom-up pipeline.

2.2.1. Top-Down Pipeline

The top-down pipeline starts by detecting all persons within an image and creates bounding boxes around them. The next step is to use each of the detected bounding boxes and perform a single-person approach for each of them. The single-person approach creates keypoints for each detected person, after which the pipeline may include additional post-processing steps and enhancement of the final results, as described in Figure 5.

The top-down method was first proposed in the study by Toshev and Szegedy [88], where a face detector-based model was used to determine the bounding box of the human body. In the next step, a multilevel DNN-based cascade regressor was used to estimate

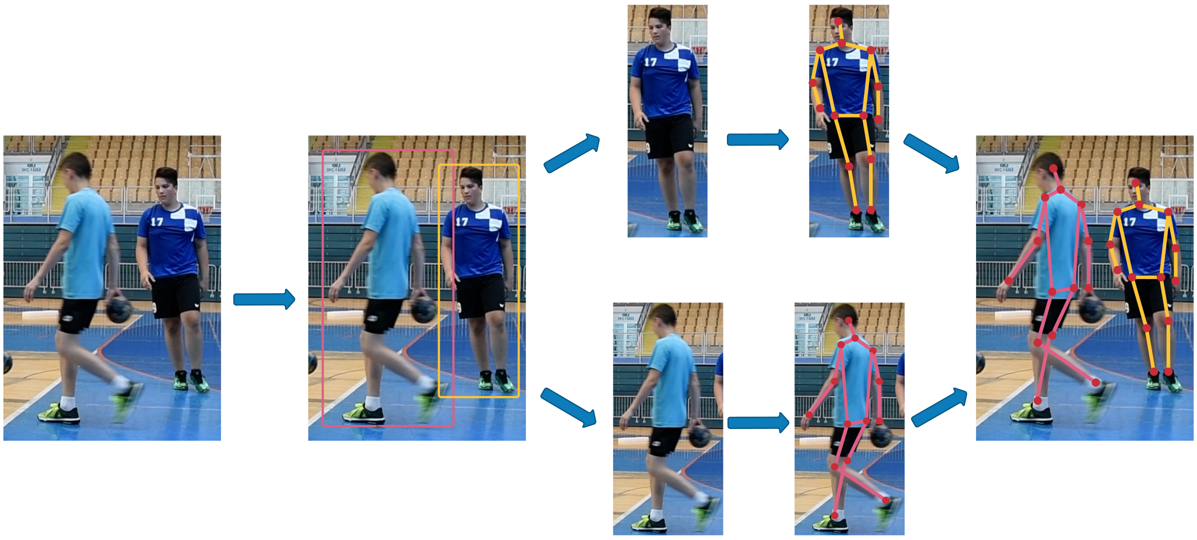


Figure 5: The top-down pipeline in multi-person approach for pose estimation. It starts by detecting all persons within an image and producing bounding boxes, on which a single-person approach is applied. The results are keypoints for each detected person, after which the pipeline may involve additional post-processing steps and improving the final results.

the joint coordinates.

He et al. [27] developed a segmentation model as an extension of the Faster Region-Based Convolutional Neural Network (R-CNN) [73] model by adding a branch to predict object masks. The robustness and better results of the proposed model were improved by using a human pose estimation model. The Mask R-CNN simultaneously predicts the human bounding box and the human keypoints, which speeds up the recognition by sharing the features between the models.

Radosavovic et al. [70] used omni-supervised learning with the Mask R-CNN detector for challenging real-world data. Self-learning techniques were applied so that the predictions of the Mask R-CNN detector on unlabelled data were used as additional training data.

Fant et al. [23] used the sensitivity of a single-person pose estimation to bounding box detection. The authors developed a method to handle inaccurate bounding boxes and redundant detections by using a Symmetric Spatial Transformer Network (SSTN) and a Pose-Guided Proposals Generator (PGPG). Moreover, PGPG is used to greatly augment the training data by learning the conditional distribution of bounding box proposals for a given human pose. This adapts the single-pose estimator to handle human localization errors due to SSTN and parallel use of the single-pose estimator.

2.2.2. Bottom-Up Pipeline

The bottom-up pipeline works like a reverse top-down pipeline and starts by detecting all keypoints in the image, which are then associated with human instances, as shown in Figure 6. Compared to the top-down pipeline, the bottom-up pipeline is likely to be faster because it does not detect human bounding boxes and does not perform pose estimation separately for each person detection.

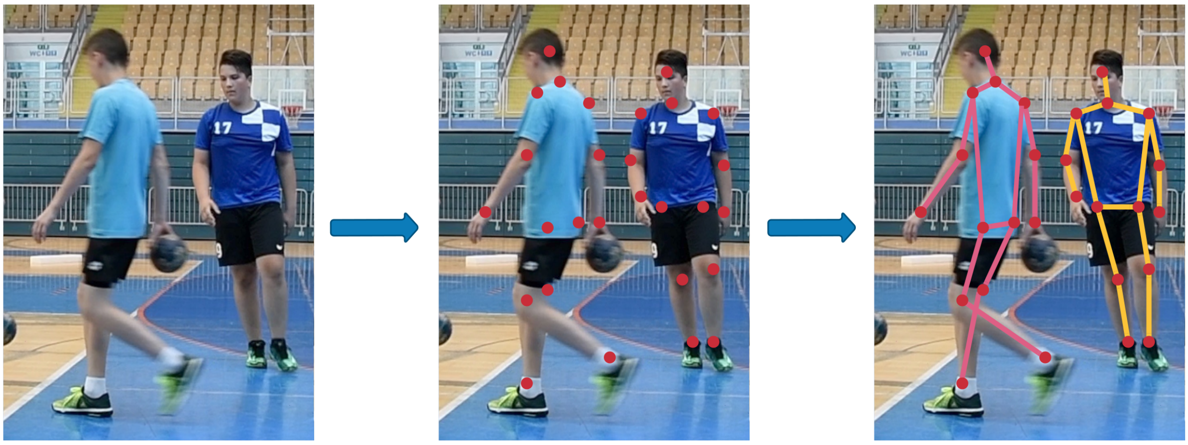


Figure 6: The bottom-up pipeline in multi-person approach for pose estimation. It starts by detecting all the keypoints in the image, which are then associated with human instances.

The bottom-up multi-person pipeline for pose estimation was first proposed by Pishchulin et al. [66]. They formulated it as a joint problem of partitioning and labeling subsets. The model jointly determines the number of persons, their poses, spatial proximity, and occlusions at the part level. Their formulation implicitly performs non-maximum suppression on the set of keypoint candidates and groups them to form body part configurations that account for geometric and visual constraints. Insafutdinov et al. [33] improved the performance of the previously described method [23] in complex scenes by using a deeper neural network for better recognition of body parts and introducing new image-conditioned pairwise terms to achieve faster pose estimation.

Insafutdinov et al. made another improvement [32] by simplifying and reducing the body part relation graph, using current methods for faster inference, and shifting much of the inference about body part association to a feed-forward convolutional architecture.

Next, Cao et al. [11] proposed a non-parametric representation called Part Affinity Fields (PAFs) to learn the association of body parts to people in the image. Their model generates a set of confidence maps for body part positions and a set of vector fields of part affinities, which are finally parsed by greedy inference to output keypoints.

Newell et al. proposed associative embeddings [57], which is a method that simultaneously outputs detection and group assignment and outperforms bottom-up methods such as in Refs. [66, 32, 11], as well as a top-down method proposed in Ref. [23]. The embeddings serve as tags that encode a grouping: detection with similar tags should be grouped, i.e., body joints with similar tags should be grouped into one person.

Huang et al. [30] took a different direction in their search for performance benefits in a pose estimation task. They focused on the data processing problems arising from complex biased coordinate system transformations and keypoint format transformation methods. Therefore, they proposed Unbiased Data Processing (UDP), which consists of two techniques: an unbiased coordinate system transformation (achieved with elementary operations such as cropping, resizing, rotating, and flipping) and an unbiased keypoint format transformation (achieved by an improved keypoint format transformation between heat maps and keypoint coordinates).

A summary of the key differences between described methods is shown in Table 1.

Table 1: Comparison of the key differences between methods for 2D pose estimation. A checkmark in column *Structure-aware* represents the methods' ability to ensure the validity of the human skeleton structure. A checkmark in the column *Use of temporal data* represents whether the method uses previous predictions or other temporal information.

Method	Approach	Human Prediction	Structure-Aware	Use of Temporal Data	Prediction	Type
Toshev and Szegedy [88]	Top-down	Single			Joint	Regression
Carreira et al. [12]	Top-down	Single	✓		Joint	Regression
Luvizon et al. [52]	Top-down	Single			Joint	Regression
Sun et al. [85]	Top-down	Single	✓		Bone	Regression
Chen and Yuille [14]	Top-down	Single	✓		Joint	Heatmap
Newell et al. [58]	Top-down	Single			Joint	Heatmap
Chen et al. [16]	Top-down	Single	✓		Joint	Heatmap
Chu et al. [18]	Top-down	Single	✓		Joint	Heatmap
He et al. [27]	Top-down	Single			Joint	Heatmap
Radosavovic et al. [70]	Top-down	Single			Joint	Heatmap
Fant et al. [23]	Top-down	Single			Joint	Heatmap
Pishchulin et al. [66]	Bottom-up	Multi			Joint	Regression
Insafutdinov et al. [33, 32]	Bottom-up	Multi	✓	✓	Joint	Heatmap
Cao et al. [11]	Bottom-up	Multi	✓		Joint	Heatmap
Newell et al. [57]	Bottom-up	Multi	✓		Joint	Heatmap
Huang et al. [30]	Bottom-up	Multi	✓		Joint	Heatmap

2.3. 3D Pose Estimation

The 3D pose estimation aims to provide a complete and accurate 3D reconstruction of a person's motion from a monocular camera or, more commonly, from 2D position keypoints.

Early studies focused on predicting 3D poses directly from images. Li and Chan first introduced the concept of predicting 3D poses using Deep Learning in Ref. [46] by constructing a convolutional neural network trained to regress 3D keypoints directly from the image. Their simple approach outperformed previous approaches that do not impose constraints on the definition of correlation between body parts. Tekin et al. [86] build on a similar idea to Ref. [46], but take advantage of using auto-encoders in latent space for 3D pose representation. First, they trained the auto-encoder to reconstruct a 3D pose given as input to the network and generated a pose representation in latent space (middle layer in the network). A CNN network was then trained to generate pose representations directly from images, rather than regressing keypoints directly, as was done in previous work. The resulting pose representation from the CNN network is then fed into the decoder network to generate a 3D pose. In addition, this approach enforces an auto-encoder that implicitly learns constraints about the human body, improving pose consistency and correlation between body parts without being explicitly trained. Pavlakos et al. [64] formulate 3D pose estimation as a 3D keypoint localization problem in a voxel space using a convolutional network to create keypoints heatmaps. The input to the network is a single image, and the output is a dense 3D volume with separate probabilities per voxel for each joint. To handle the high dimensionality and enable iterative processing, they incorporated a coarse-to-fine supervision scheme instead of using a single component with a single output.

Splitting the task of 3D pose estimation into two steps proved to be a better approach than directly predicting 3D poses from images and is more commonly used in recent studies.

Martinez et al. [53] presented a simple deep feed-forward network that ”lifts” 2D joint positions into 3D space, outperforming all previous methods. They analysed errors in previous approaches that predicted 3D keypoints directly from images and concluded that one of the main causes of errors stems from 2D pose estimation, which propagates errors further into later steps. By separating the two tasks, overall accuracy is improved because each step can be evaluated and improved separately.

Recent studies have focused primarily on improving estimation performance by evaluating temporal pose information across multiple images or frames. Hossain and Little [72] used the temporal information about a sequence of 2D joint position to estimate a sequence of 3D poses by using a sequence-to-sequence network of layer-normalized LSTM units. The proposed seq2seq network uses only the previous frames to understand the temporal context and produces predictions with errors uniformly distributed over the sequence. In Ref. [65], Pavllo et al. proposed a simple and effective approach for 3D human pose estimation based on dilated temporal convolution of 2D keypoint trajectories and a semi-supervised approach that exploits unlabelled video to improve performance when there is limited data. Their convolutional network achieves similar results to more complex LSTM sequence-to-sequence models and solves the problem of pose drift over long sequences of seq2seq models. In Ref. [17], Chen et al. solved the problem of missing information due to occlusions, out-of-frame targets, and inaccurate person detection by proposing a framework that integrates graph convolutional networks (GCNs) and temporal networks (TCNs). They proposed a human-bone GCN that models bone connections and a human-joint GCN based on a directed graph. By using the two GCNs, they can robustly estimate the spatial frame-wise 3D poses enough to work with occluded or missing information about human parts. In addition, a joint TCN was used to estimate the person-centred 3D poses across multiple frames and a velocity TCN was used to estimate the velocity of the 3D joints to ensure the consistency of the 3D pose estimation in successive frames. By using the two TCNs, 3D pose estimation can be performed without requiring camera parameters. Li et al. proposed a novel augmentation method [45] that is scalable to synthesize a large amount of training data for training 2D-to-3D networks, which can effectively reduce the bias of datasets. The proposed data evolution strategy extends an existing dataset through mutations and crosses of selected poses to synthesize novel human skeletons to expand the dataset in the order of 10^7. In addition, they proposed a novel 2D-to-3D network that contains a cascaded 3D coordinate regression model and where each cascade is a feed-forward neural network.

A summary of the key differences between described methods is shown in Table 2.

Table 2: Comparison of the key differences between methods for 3D pose estimation. A checkmark in column *Structure-aware* represents the methods' ability to ensure the validity of the human skeleton structure. A checkmark in the column *Use of temporal data* represents whether the method uses previous predictions or other temporal information. Image in column *Input* means that the model predicts directly from the image, while *2D keypoints* means that the model "lifts" the 2D keypoint to the 3D space.

Method	Input	Human Prediction	Structure-Aware	Use of Temporal Data	Prediction	Type
Li and Chan [46]	Image	Single	✓		Joint	Regression
Tekin et al. [86]	Image	Single	✓		Joint	Regression
Pavlakos et al. [64]	Image	Single			Joint	Heatmap
Martinez et al. [53]	2D keypoints	Single			Joint	Regression
Hossain and Little [72]	2D keypoints	Single		✓	Joint	Regression
Pavllo et al. [65]	2D keypoints	Single		✓	Joint	Regression
Chen et al. [17]	2D keypoints	Single		✓	Joint	Regression
Li et al. [45]	2D keypoints	Single			Joint	Regression

2.4. Occlusion

Occlusion is the predominant problem in estimating human posture, and a number of papers have attempted to solve this problem. Iqbal and Gall [35] considered multiple person pose estimation as an association problem between two persons, and used linear programming to solve the association problem anew for each person. Chen et al. proposed a novel network structure called Cascaded Pyramid Network (CPN) [15], which includes GlobalNet and RefineNet. The GlobalNet is used to locate visible keypoints, while the RefineNet is used to handle keypoints that are difficult to see or hidden. Fang et al. [23] used Non-Maximum Suppression to solve the occlusion problem and eliminate redundant poses, the problem caused by redundant detections. A similar approach was implemented in Ref. [63] to eliminate redundant detections.

2.5. Metrics

In the early works, frequently used metric was the Percentage of Correctly estimated body Parts (PCP) [25]. In PCP, a limb is considered to be detected and to be a correct part if the distance between the predicted and true joint position is less than the bone length multiplied by a chosen factor. The true joint position of the limb is at most half the length (PCP at 0.5), as shown in Equation (1). Another widely used metric is PCK (Percentage of Correct Keypoints) [95] and its variant PCKh, shown in Equation (2). In both metrics, a joint is considered detected and correct if it is within a certain number of pixels from the ground truth joint, determined by the height and width of the person bounding box (or person's head in the case of PCKh). More recent metrics are Percentage of Detected Joints (PDJ) [88], shown in Equation (3), and Object Keypoint Similarity (OKS) [75], shown in Equation (4). PDJ considers a joint to be correctly detected if the distance between the predicted joint and the true joint is within a certain fraction of the diagonal of the bounding box. OKS is calculated from the distance between the predicted points and the ground truth points normalized by the person's scale. The OKS metrics show how close the predicted keypoint is to the ground truth, with a value from 0 to 1. The final performance calculation usually involves thresholding the OKS metrics and calculating the Average Precision (AP) and Average Recall (AR), as shown in Equation (5). Mean Per Joint Position Error (MPJPE) is the most commonly used metric. MPJPE, Equation (6), calculates the Euclidean distance between the estimated 3D joint and the ground truth position, and the final score is calculated by averaging the distances across all frames. A common addition in the evaluation process is to align the poses before calculating the metrics. The most widely used alignment method is Procrustes alignment, which relies on Procrustes analysis to compare the two poses and align them on all axes. Metrics that use Procrustes alignment are usually marked with the prefix PA (e.g., PA-PCK, PA-MPJPE).

PCP, PCKh, and PDJ metrics are calculated as follows:

$$PCP = \frac{\sum_{i=0}^{n} bool(d_i < 0.5 * limb\_length_i)}{n} \tag{1}$$

$$PCKh = \frac{\sum_{i=0}^{n} bool(d_i < 0.5 * height\_of\_the\_head)}{n} \tag{2}$$

$$PDJ = \frac{\sum_{i=1}^{n} bool(d_i < 0.05 * diagonal)}{n} \tag{3}$$

where d_i is the Euclidean distance between the ground truth keypoint and predicted keypoint, $bool(condition$ is a function that returns 1 if the condition is true and 0 if it is false, n is the number of keypoints on the image. $limb\_length$, $head\_height$, and $diagonal$ are expressed as the number of pixels per the model predictions expressed in pixels as well.

In PDJ, the diagonal is calculated from the bounding box using the Pythagorean theorem, i.e., $diagonal = \sqrt{(height^2 + width^2)}$.

The OKS metric is calculated as follows:

$$OKS = exp(-\frac{d_i^2}{2s^2k_i^2}) \tag{4}$$

where d_i is the Euclidean distance between the ground truth keypoint and predicted keypoint, s is the square root of the object segment area (scale), and k is a per-keypoint constant that controls fall off.

AP and AR metrics with Precision and Recall formulas are calculated as follows:

$$\begin{aligned} Precision &= \frac{TP}{TP + FP} \\ Recall &= \frac{TP}{TP + FN} \\ AP &= \sum_{i=0}^{n} Precision_i \\ AR &= \sum_{i=0}^{n} Recall_i \end{aligned} \tag{5}$$

where n is the number of keypoints, TP represents True Positives, FP represents False Positives, and FN represents False Negatives.

The MPJPE metric is calculated as follows:

$$E_{\text{MPJPE}}(f, \varphi) = \frac{1}{N_\varphi} \sum_{i=1}^{N_\varphi} \left\| P_{f,\varphi}^{(f)}(i) - P_{gt,\varphi}^{(f)}(i) \right\|_2 \tag{6}$$

where f denotes a frame and φ denotes the corresponding skeleton. $P_{f,\varphi}^{(f)}(i)$ is the estimated position of joint i and $P_{gt,\varphi}^{(f)}(i)$ is the corresponding ground truth position. N_φ represents the number of joints.

2.6. Standardization — Spatial Alignment, Normalization, and Retargeting

Because images may be of different sizes, a person may appear in a different part of the image, requiring a preprocessing step to allow consistent calculation of accuracy metrics. In addition, the use of accelerometers or motion capture sensors such as mocaps complicates the task of accurately evaluating pose estimation methods. In these cases, it is suggested to apply pose transformations to remove potential errors caused by inappropriate preprocessing.

A simple solution is to normalize the resulting keypoint coordinates by treating them as an L2-normalized vector array. In addition, the poses can be aligned by a selected pose point (e.g., a point between the hips [77]) or by Procrustes analysis, as in Refs. [71, 87, 13, 99]. In our experiments, we defined and applied a simple normalization procedure where the person is scaled so that the height of the person is 1, and we will here refer to it as the h-norm. The H-norm assumes that there is at least one frame in the sequence in which the person is stretched, finds that frame, and then scales the person based on this frame. The height of the person is calculated as the distance between the nose and the foot keypoints, taking into account the foot that is further away from the nose. Finally, the h-norm sets the height of the person in the selected "stretched" image to 1 and scales the other images accordingly.

A more advanced solution is to use a retargeting method, as proposed in Refs. [55, 100, 62, 1], which transfers the joint angles from the predicted pose to a standardized skeleton. The result is a new pose where the limbs are always the same length, which also solves the problem of pose estimation models with small variations in keypoint detection. For example, a hand may be detected at the wrist or in the palm region, and this mismatch of detections results in an incorrect limb length.

In this experiment, we applied the simplest way of implementing a retargeting method, which is to use a direction vector. A keypoint P_t is retargeted using the root keypoint P_r by subtracting the two points to produce a direction vector $\overrightarrow{p}_t$ (the magnitude vector). Then, we rescale $\overrightarrow{p}_t$ to the distance between the targeted root point T_r and the targeted keypoint T_t to produce the direction vector $\overrightarrow{t}_t$. Finally, we add $\overrightarrow{t}_t$ to point T_r producing the retargeted keypoint T'_t. An example of the retargeted pose is shown in Figure 7.

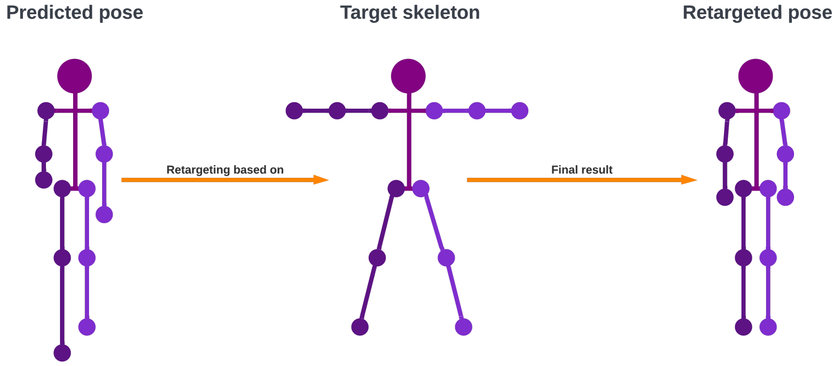


Figure 7: An example of pose retargeting where the predicted pose is retargeted based on the target skeleton. Retargeting will translate the joint angles from the predicted pose to a standardized skeleton, thus ensuring that a pose has the same lengths of limbs.

2.7. Datasets

Several publicly available datasets are provided for various image processing tasks and domains. Among the most popular datasets are COCO [49] and ImageNet [21], which contain many tagged images of various objects in the real-world conditions.

It is necessary to properly collect the data and prepare it for machine learning for various tasks such as image classification, object detection, object localization, object segmentation, object tracking, etc. For image classification, the images are annotated with a label corresponding to the class of the object that exists on the scene; for detection, the objects in the scene are surrounded by a bounding box, or the image area corresponding to the object is segmented. Finally, the skeleton of the object should be labelled for pose estimation.

The most well-known dataset in the field of pose estimation is the Human3.6M dataset [34]. It consists of 3.6 million human poses and corresponding images captured with a motion capture system. The dataset contains 11 actors performing 17 activities (discussing, smoking, taking pictures, talking on the phone, etc.). Examples from the dataset are shown in Figure 8a.

There are also appropriate datasets with images or video sequences that are specific to a particular domain. For example, in the sports domain, data on Olympic sports

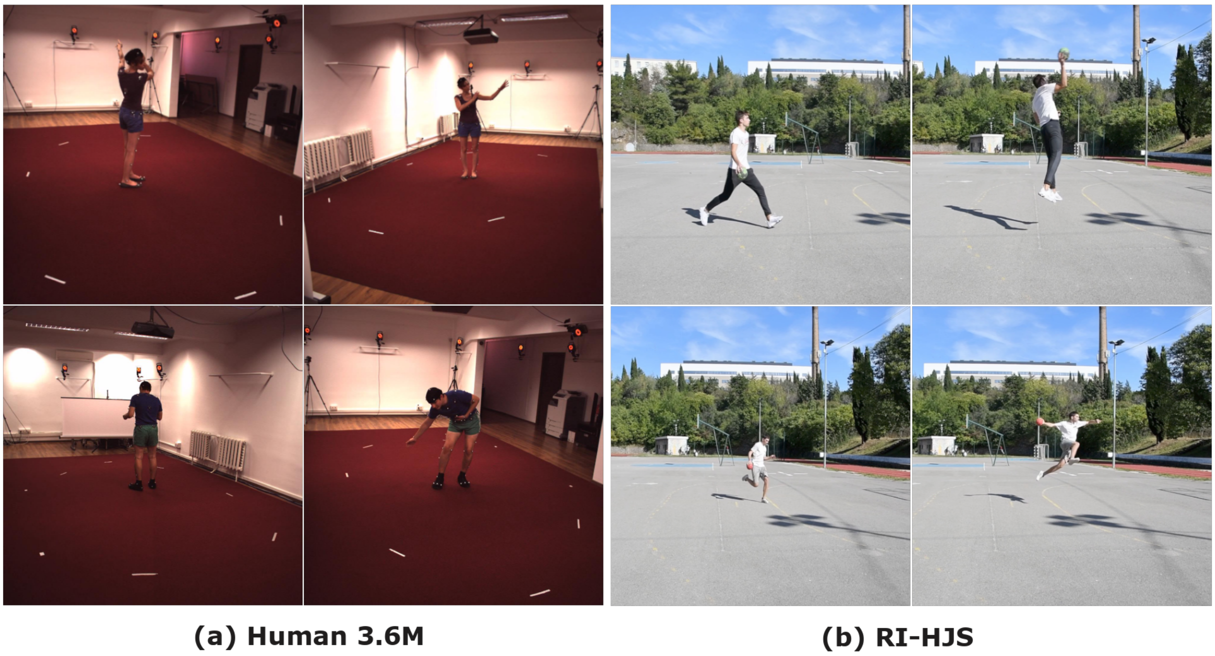


Figure 8: Examples from the Human3.6 dataset (**a**) and the RI-HJS dataset (**b**) for 3D pose estimation.

are very popular for classifying sports scenes and are collected in the Olympic Sports Dataset [59], and SVW [76] contains short sequences of actions related to 16 and 30 sports, respectively. There are also specialized databases of videos related to specific sports, such as UNIRIHBD [37], for researching the performance of athletes in handball, basketball [82], and volleyball [31].

Older publicly available image datasets such as KTH [78] and Weizmann [9] were filmed under controlled conditions with fewer actors, while on the other hand, datasets such as HACS [98] and Kinetics 700-2020 [79] were filmed under real-world conditions and contain many more classes and data. Kinetics, for example, is a large dataset (with 400 to 700 classes corresponding to different human activities depending on the version) that contains manually tagged videos downloaded from YouTube. Other popular datasets in the sports domain are UCF Sports Action Data Set [81] and Sports-1M [41].

In the experimental part of this work, we prepared and used our own dataset of handball scenes collected in Rijeka (RI-HJS). Handball is an Olympic team sport played with a ball and is very popular in Europe, but is not represented in the aforementioned databases for training models for sports scenes. RI-HJS contains 21 short clips with an average length of 9 s, in which 2 different players perform several handball jump shots. Both players were equipped with Wear-Notch motion capture sensors to capture the ground truth positions of the joints. The documentation states that the static accuracy of the Wear-Notch sensors is approximately 1–2° yaw/tilt/roll. We used a single still camera with 1920 × 1080 resolution positioned on the tripod 1.5 m from the ground, while the players were about 7–10 m away from the camera. Examples from the dataset are shown in Figure 8b.

3. Tracking

Multiple object tracking (MOT) in videos is an actively researched area in computer vision, and in this paper we present the main methods to achieve the best performance. The main goal of multiple object tracking (MOT) is to track the position and identity of multiple objects so that each object is assigned the same unique ID in each frame in which it appears in the video.

Tracking produces the best results when objects are moving uniformly, in the same direction, and without occlusions. Examples where tracking works well include runners chasing each other on the edge of a playground, or cars moving in the same direction and at the same speed on the road without being obscured by objects. However, this is usually not a realistic scenario, especially in team sports where many players change the direction of movement, speed, distance from the camera, position, and activity performed. They also frequently enter and exit the camera's field of view, so they are visible in some shots and not in others. They also stand very close to each other to interfere with the opponent and prevent him from taking appropriate action. They often occlude each other, and because of the obscuring, it is difficult to detect their whole body. The players of the team wear the same jerseys, so they can be identified only by the number on the jersey or some details such as hair colour or sneakers. In dynamic scenes, tracking more objects is still a big challenge. However, thanks to the improved performance of object and skeleton detectors, even in crowded scenes, and improved computer performance, tracking by detection has become the leading paradigm for MOT.

In tracking by detection, the tracking algorithm relies on the results of the object detectors in each frame and combines the information

In general, multiple object tracking is about detecting bounding boxes of an object in successive frames and a method to map them between image sequences, thus creating object trajectories. The taxonomy of tracking methods described in this paper is shown in Figure 9, while an example of tracking on an image is shown in Figure 10.

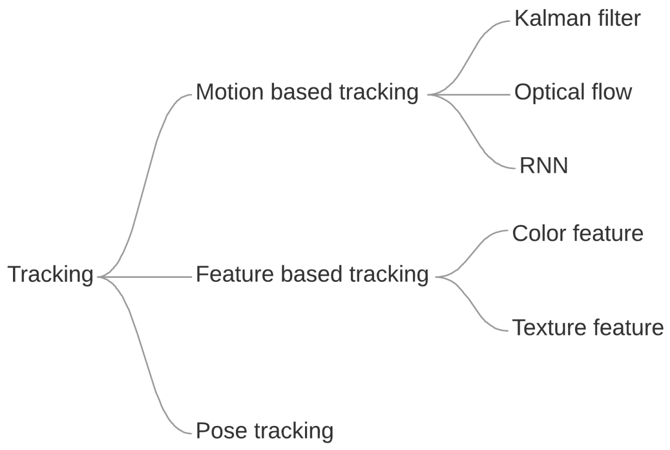


Figure 9: Taxonomy of the tracking methods.

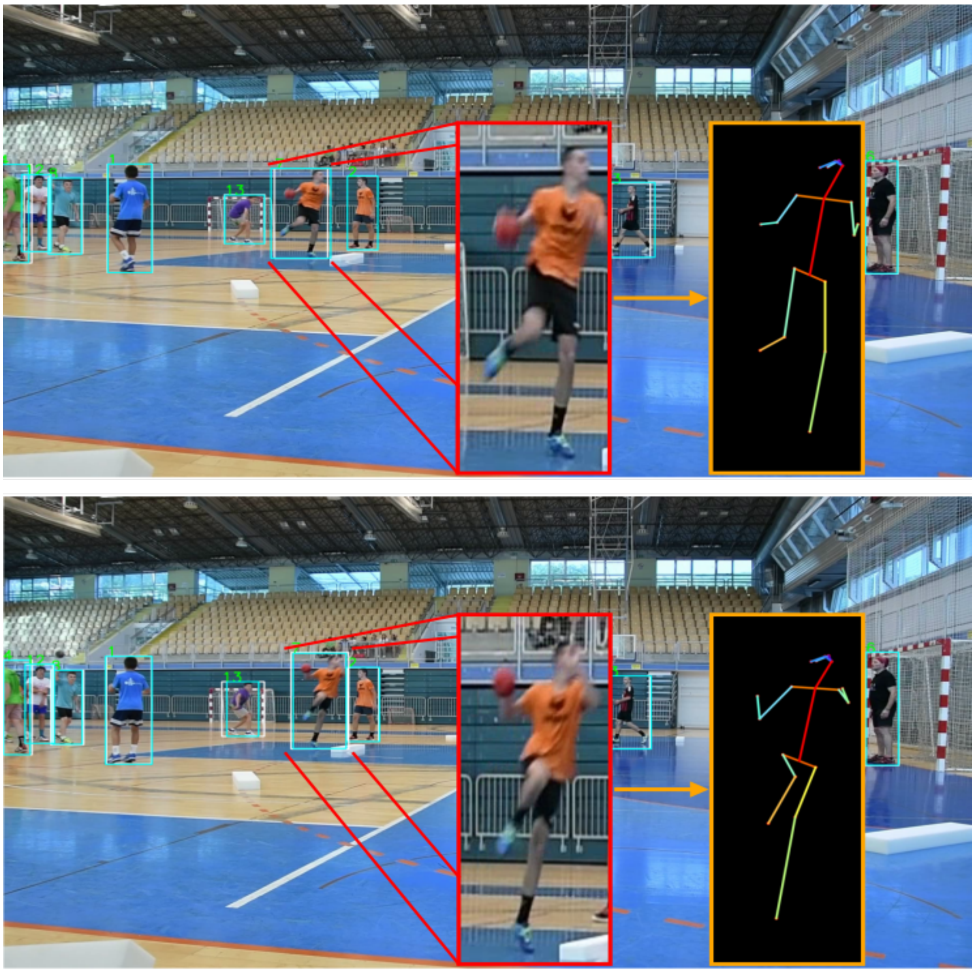


Figure 10: Two frames of a tracked player executing a jump shop where poses are estimated and performed necessary transformation. Blue bounding boxes visualize the detectors' outputs, while white bounding boxes visualize the tracking algorithm bounding box prediction. To standardize pose sizes because players can be further away or closer to the camera, we perform transformations to the pose (i.e., standardization).

3.1. Motion-Based Tracking

Motion-based detection methods mainly consist of background subtraction and difference between adjacent frames. Motion models are often robust and computationally light, but their performance is heavily affected by noise and depends heavily on frame registration, so even small errors in frame registration or illumination changes can lead to large errors in motion-based object detection. A typical traditional approach has been background modelling using Gaussian Mixtures (GMM). However, since these are not capable of detecting objects in the scene, several methods have been proposed that combine background distribution estimation with numerous filters for video post-processing and object detection.

Motion-based tracking involves recording the motion of an object in a source video clip, then analysing its motion and trajectory, and using this motion behaviour to predict a target object in a sequence of video clips. A well-known example is the use of the Kalman filter [47, 56] to estimate the position of a linear system, assuming that the errors are Gaussian. The Kalman filter [39] is an algorithm that uses a series of measurements observed over time, including noise and other inaccuracies. It provides estimates of target variables that are usually more accurate than estimates based on a single measurement. The Kalman filter is usually combined with various techniques to represent object features or to improve the estimate of the target position [10, 26, 24]. One of the most popular tracking systems that use the Kalman filter is Simple Online and Realtime Tracking (SORT) [8], a system based on state estimation techniques designed for online tracking where only previous and current frames are available. SORT uses the Kalman filter to predict object position in the current frame based on the previous frames, i.e., object movement across previous frames, along with the Hungarian matching algorithm [44] to perform data mapping and assignment on the same track (connecting bounding boxes across frames). The Hungarian algorithm searches for the optimal bounding box that best matches a given bounding box in the previous frame, given a cost allocation function that depends only on the parameters of the bounding box. The parameters used to assign objects on the track are the Euclidean distance of each detected object from the predicted centre of the last object on the track and the difference in bounding box size between the detected object and the last assigned object on the same track. This algorithm does not consider visual features and similarities between objects in successive frames. An object is assigned to a track if the reliability of the detector is higher than the set threshold. If the number of detected objects exceeds the number of currently active tracks, new tracks are created and initialized with the new object.

In some works [4, 38, 29, 93, 67, 61], optical flow was used for object tracking by separating the moving foreground objects from the background and generating an optical flow field vector for the moving object. Optical flow is a low-level feature determined from the time-varying image intensity between subsequent frames. The moving point in the image plane estimated from successive video frames, e.g., by using the Lucas–Canada method [51], generates a 2D path $x(t) \equiv (x(t), y(t))^T$ with coordinates at the centre of the camera and the current direction of motion described by the velocity vector $dx(t)/dt$. The 2D velocities of all visible points in the image form a 2D vector field of motion, where the magnitude corresponds to the velocity of motion and the angle represents the direction of motion.

Other works, such as Refs. [60, 96, 50, 40], use Recurrent Neural Network (RNN) to learn the motion behaviour of objects and use them for object tracking, usually applying them to bounding box coordinates. RNNs have connections that feed activations from an input in a previous time step back into the network, called memory cell units, which affect the output for the current input. These activations from the previous time step can be held in the internal state of the network to model long-range dependencies, so that the temporal context of the network is not limited to a fixed window and the network can model sequences such as video images in action recognition.

3.2. Feature-Based Tracking

Feature-based tracking is a method in which objects (features) in the data are first segmented, and then these segmented objects are tracked (correlated) in successive time steps based on the representation of their appearance, i.e., colour, texture, shape, and so on. Wojke et al. [91] improved the method proposed in Ref. [8] SORT by introducing a deep association metric. This is achieved by capturing object features within the bounding box to enable object tracking through longer occlusion periods, thus reducing the number of identity switches. Subsequent work, such as Refs. [84, 42, 28], has focused on improving object associations between frames using different methods or constructing a single model to perform object tracking and association. Further improvements were made by segmenting objects within the detected bounding box to eliminate unnecessary information (background, other objects, etc.), as proposed in Ref. [89], and subsequent improvements to the new approach [80, 68, 97].

3.3. Pose Tracking

Iqbal et al. [36] first formulated the problem of pose estimation and tracking for multiple persons and presented a sophisticated "Multi-Person PoseTrack" dataset. The authors proposed a method to solve this problem by representing the joint body detection with a spatiotemporal graph and solving an integer linear program to partition the graph into subgraphs corresponding to the plausible body pose trajectories for each person. Xiu et al. proposed a PoseFlow method [92], which consists of two techniques, namely, Pose Flow Builder (PF-builder) and Pose Flow non-maximum suppression (PF-NMS). PF-Builder is used to associate the cross-frame poses pointing to the same person by iteratively constructing a pose flow using a sliding window, where PF-NMS uses the pose flow as a single unit in NMS processing to stabilize tracking. Doering et al. [22] proposed a temporal model that predicts temporal flow fields, i.e., vector fields that indicate the direction in which each body joint will move between two successive frames. Raaj et al. [69] built on the Part Affinity Fields (PAF) [11] representation and proposed an architecture that can encode and predict Spatio-Temporal Affinity Fields (STAF). Their model encodes changes in the position and orientation of keypoints over time in a recurrent manner, i.e., the network takes STAF heatmaps from previous frames and estimates them for the current frame. Bao et al. [5] proposed a framework for pose-aware tracking-by-detection that combines pose information with methods for detecting people in videos and associating people. The system uses prediction of the location of people in the detection phase, and thus uses temporal information to fill in the missing detections. In addition, the authors propose a Pose-guided Graph Convolutional Network (PoseGCN) for person association, a modelling task that uses the structural relationships between person and the global features of a person.

In Ref. [6], Bazarevsky et al. focused on developing a lightweight method for estimating and tracking the single-person pose. They followed the top-down pipeline and used a face detector and certain computations to determine the width and height of a person's bounding box, which made the detection fast. For the pose estimation step, the authors chose a combined heatmap, offset, and regression approach, using heatmaps and offset losses only during training. Kong et al. [43] proposed a framework consisting of the Posebased Triple Stream Network (PTSN) and an online multi-state matching algorithm. PTSN is responsible for computing the similarity values between the historical tracklets and the candidate detection in the current frame. The values come from three network streams that model three pose cues, i.e., pose-based appearance, movements, and athlete interactions. An example of a tracked 2D pose sequence over 80 frames is shown in Figure 11.

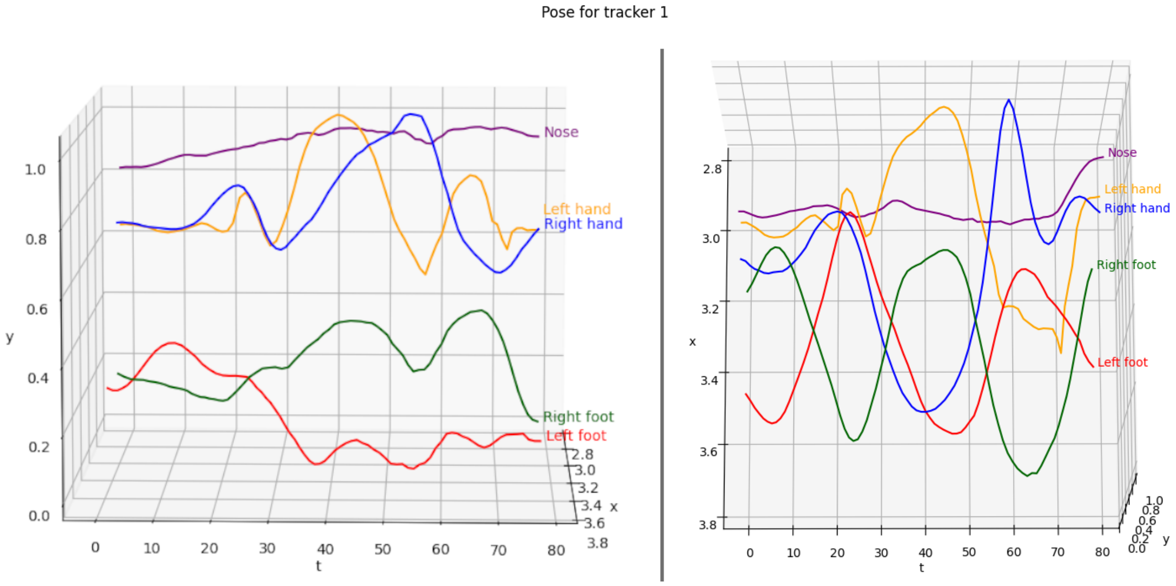


Figure 11: A 3D plot visualization of the 2D sequence joints in space and time when executing a jump shot, showing a side and top view of the plot.

3.4. Metrics

The evaluation of tracking algorithms usually involves a number of metrics. The most basic metric is the number of ID switches (IDsw) [94], which counts how many times an algorithm has switched (or lost) an object ID, as shown in Equation (7). An improvement on the IDsw metric is the IDF1 metric [74], which is computed as the ratio of correctly identified detections to the average number of detections based on ground truth and computed detections. ID precision and ID recall provide information about tracking tradeoffs. At the same time, the IDF1 score allows all trackers to be ranked on a single scale that balances identification precision and recall by their harmonic mean (see Equation (8)).

The most widely used metric is Multiple Object Tracking Accuracy (MOTA) [7], which combines three sources of error: false positives, missed targets, and identity switches into a single number, as shown in Equation (9). Another popular metric is Multiple Object Tracking Precision (MOTP) [7], which calculates the offset between the annotated and predicted bounding boxes, as shown in Equation (10). Finally, the Mostly Tracked targets (MT) [48] metric measures tracking completeness by calculating the ratio of trajectories covered by a track hypothesis to at least 80% of their respective lifetimes. The metric ML (Mostly Lost Targets [48]) is a complement to MT, which computes the ratio of trajectories covered by a track hypothesis during at most 20% of their respective lifetimes.

The IDsw metric is calculated as follows:

$$IDSW_t = \sum_{i=0}^{n} bool(ID(o_i)_{t-1} \neq ID(o_i)_t) \tag{7}$$

where t is the frame index, n is a number of objects in the frame, o is the tracked object, and $bool(condition)$ is a function that returns 1 if the condition is true and 0 if it is false.

The IDF1 metric is calculated as follows:

$$IDF_1 = \frac{2\ IDTP}{2\ IDTP + IDFP + IDFN} \tag{8}$$

where $IDTP$ represents the number of correctly identified objects, $IDFP$ represents the number of falsely identified objects, while $IDFN$ represents the number of objects detections that fall outside the valid region of its corresponding ground truth.

The MOTA metric is calculated as follows:

$$MOTA = 1 - \frac{\sum_t FN_t + FP_t + IDSW_t}{\sum_t GT_t} \tag{9}$$

where t is the frame index, and GT is the number of ground truth objects.

The MOTP metric is calculated as follows:

$$MOTP = \frac{\sum_{t,i} d_{t,i}}{\sum_t c_t} \tag{10}$$

where c_t denotes the number of matches in frame t and $d_{t,i}$ is the bounding box overlap of target i with its assigned ground truth object.

3.5. Datasets

Currently, the most popular and widely used dataset is Multiple Online Tracking (MOT) [54], which contains seven different indoor and outdoor scenes of public places with pedestrians as the objects of interest. The dataset provides detections of objects in the video frames using three detectors: SDP, Faster-RCNN, and DPM.

Datasets where most people have similar appearance, such as sports and dance datasets, can greatly affect methods that rely on appearance as a tracking feature, and it is important to evaluate models in such scenarios. This type of dataset includes DanceTrack [83] and SportsMOT [90]. DanceTrack consists of 100 videos with a total of more than 105 annotated frames and contains dancers in static scenes with uniform appearance, various movements, and frequent transitions. SportsMOT consists of 240 videos with a total of more than 10^5 annotated frames and contain 3 types of scenarios: Basketball, Football, and Volleyball, covering outdoor and indoor scenes as well as different camera angles.

Datasets such as TAO [20] and GMOT [3] aim to evaluate the generality of tracking models and encourage tracking methods to generalize to different scenarios and objects, rather than overfitting to a single scenario and benchmark. TAO contains 2907 videos taken in different environments where multiple object categories are annotated (e.g., car, truck, animal, etc.), and is currently the most diverse tracking dataset with 833 different object categories annotated for tracking. GMOT contains 40 videos of complex scenes evenly distributed across 10 object categories. Some of the main features of the dataset are: over 80 objects of the same class can appear in 1 frame and annotations are done manually with careful inspection in each frame, occlusion, target entry or exit, motion, deformation, etc.

In this paper, for model testing purposes we use our custom dataset of 20 players practicing different handball actions during training sessions in Rijeka (RI-HB-PT) to test the pose estimation models. RI-HB-PT contains 2 videos with a total of 22,676 frames and 216,601 bounding box annotations. The training is very dynamic, and there is a lot of occlusion as players pass each other very often. We used a single still camera (1920 x 1080) positioned on the tripod 1.5 m from the ground, while the player was about 5–10 m away. Examples from some data sets are shown in Figure 12.

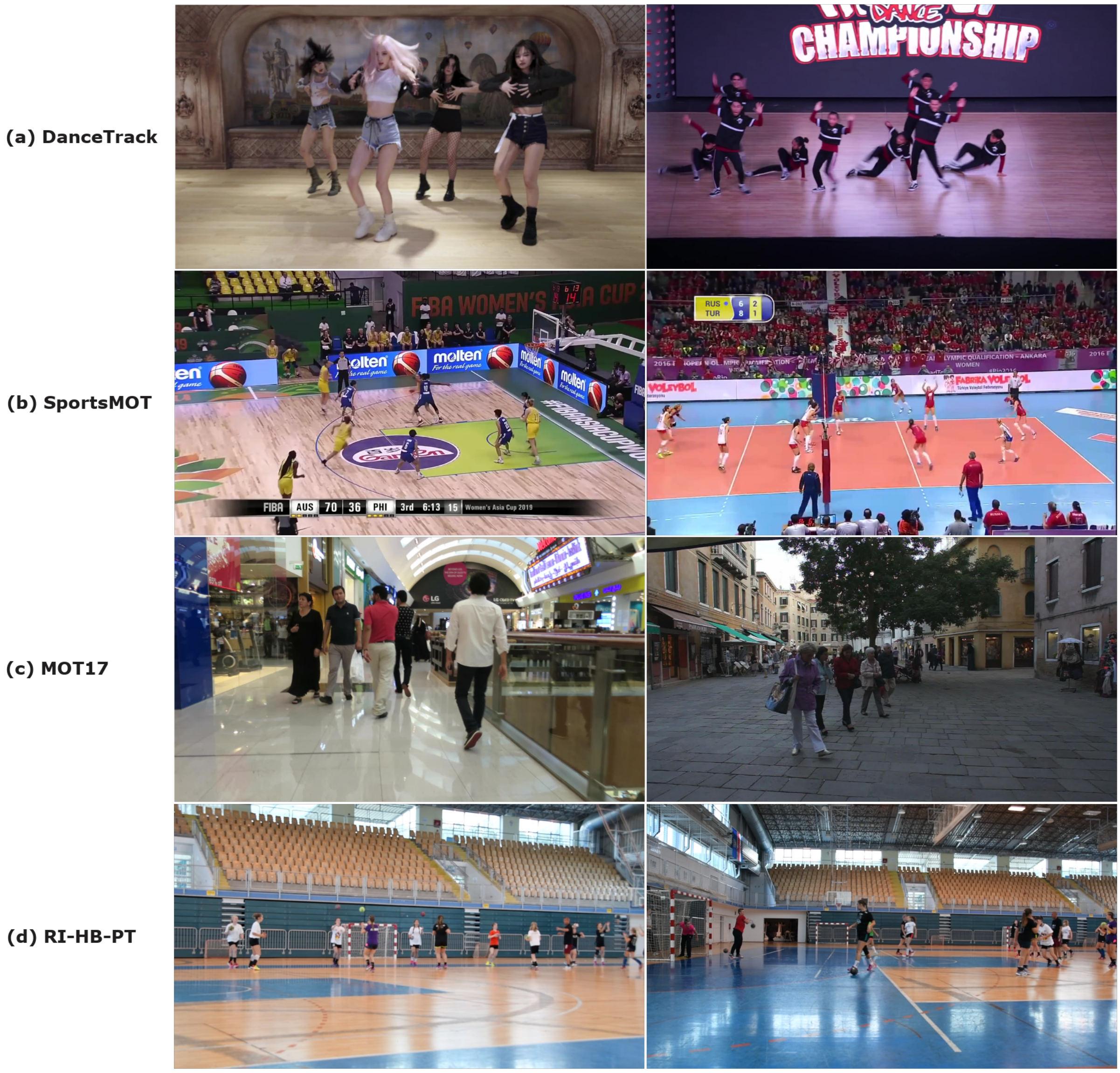


Figure 12: Examples from the tracking datasets DanceTrack (**a**), SportsMOT (**b**), MOT17 (**c**), and RI-HB-PT (**d**).

4. Evaluation of the 3D Pose Estimation Methods

We selected some well-known and well-performing methods for 2D and 3D pose estimation to evaluate on the Human3.6M dataset [34] and on our own RI-HJS dataset of handball players performing a jump shot. The Human3.6M dataset was selected because it is considered the benchmark dataset in the field of pose estimation and contains 3.6 million human poses commonly used for training pose estimation models.

RI-HJS is a customised dataset of handball scenes. We used this dataset to evaluate the robustness of models learned on a large number of standard poses from Human3.6M, and to estimate the level of generality that can be achieved on new examples from handball for which they have not been trained. Important for testing the models is the fact that the handball examples used are from the new domain, but have similar indoor conditions as other indoor sports or ordinary activities, with artificial lighting, with the player moving quickly on the field and performing different techniques and actions with the ball.

The goal of this experiment is to find a combination of models that provides the best overall results in an unseen sports environment. We considered 2D pose estimation methods: PoseRegression (https://github.com/dluvizon/pose-regression, accessed on 1 February 2022) [52], ArtTrack (https://github.com/eldar/pose-tensorflow, accessed on 1 February 2022) [32], Mask R-CNN (https://github.com/facebookresearch/detectron2, accessed on 1 February 2022) [27], and UDP-Pose (https://github.com/HuangJunJie2017/UDP-Pose, accessed on 1 February 2022) [30], and 3D pose estimation models: GnTCN (https://github.com/3dpose/GnTCN, accessed on 1 March 2022) [17], EvoSkeleton (https://github.com/Nicholasli1995/EvoSkeleton, accessed on 1 March 2022) [45], and VideoPose3D (https://github.com/facebookresearch/VideoPose3D, accessed on 1 March 2022) [65]. All 3D pose estimation models make predictions based on the results of 2D estimation models. Thus, there are 12 possible combinations of models for a final 3D prediction from an image. The 2D models PoseRegression and ArtTrack were trained using the MPII [2] training dataset, while the Mask R-CNN and UDP-Pose models were trained using the COCO 2017 training dataset.

In addition, all three 3D models, i.e., GnTCN, EvoSkeleton, and VideoPose3D, were trained with the Human3.6 training dataset, which allows for fair evaluation and comparison. Further details of the training are given in Table 3. Model combinations were evaluated using the Human3.6M validation dataset and our custom dataset of handball players executing a jump shot collected in Rijeka (RI-HJS), as described in Section 2.6. Experiments that used top-down methods were given ground truth bounding boxes to eliminate object detector errors, i.e., to evaluate only the accuracy of the pose estimation methods.

Table 3: Training details of the evaluated 2D and 3D pose estimation models. GT in the *Bounding box* column means that the models used ground truth bounding boxes in the training process. The column *2D keypoints* show the 2D pose estimation model, which produced inputs for the training of the 3D pose estimation model.

Model	Dataset	Optimizer	Learning Rate	Epoch	Bounding Box	2D Keypoints
PoseRegression	MPII	RMSProp	0.001	120	GT	-
ArtTrack	MPII	SGD	0.002	20	GT	-
Mask R-CNN	COCO	SGD	0.01	37	GT	-
UDP-Pose	COCO	Adam	0.001	210	GT	-
GnTCN	Human3.6M	Adam	0.001	100	GT	HRNet
EvoSkeleton	Human3.6M	Adam	0.001	200	GT	HRNet
VideoPose3D	Human3.6M	Adam	0.001	80	Mask R-CNN	CPN

The final results are shown in Table 4 with respect to the metrics PA-PCK and PA-MPJPE described in Section 2.5. The best results for PA-PCK is when it scores 100%, and for PA-MPJPE when is 0. The KSM and KSM + RET columns in Table 4 show the improvement in the performance of the metrics when the proposed Kalman smoothing is applied to the predicted sequence and pose retargeting. KSM means Kalman smoothing is applied to the predicted sequence to remove the noise and oscillations of the keypoints generated by the HPE method. Kalman is applied separately to all axes of the coordinate system (XYZ) for each keypoint to independently smooth the time series of keypoints across the sequence. KSM + RET means that a Kalman filter is applied to the predicted sequence for smoothing, while retargeting is applied to both the predicted and the ground truth sequence.

Table 4: Results of model combinations for 3D pose estimations on the Human3.6M dataset and the custom dataset of players performing handball jump-shot (RI-HJS). The best results are marked in bold. Metrics are computed on the normalized poses using the h-norm described in Sections 2.5 and 2.6 on 13 keypoints. KSM in the table is shorthand for Kalman smoother, which is applied on the predicted sequence before evaluation, while KSM + RET is shorthand for Kalman smoother applied on the predicted sequence and Retargeting applied both on predicted and ground truth sequences.

Dataset	Models	▲PA-PCK$_{0.15}$	▲PA-PCK$_{0.15}$ + KSM	▲PA-PCK$_{0.15}$ + KSM + RET	▼PA-MPJPE	▼PA-MPJPE + KSM	▼PA-MPJPE + KSM + RET
Human3.6M	PoseRegression + GnTCN	67.742	+1.554	+9.269	0.131	-0.003	-0.021
	PoseRegression + EvoSkeleton	68.257	+0.431	+8.370	0.130	-0.001	-0.019
	PoseRegression + VideoPose3D	69.703	+0.236	+10.065	0.127	0.000	-0.020
	ArtTrack + GnTCN	91.015	+0.769	+3.291	0.067	-0.002	-0.010
	ArtTrack + EvoSkeleton	86.698	+1.888	+6.069	0.079	-0.003	-0.015
	ArtTrack + VideoPose3D	93.056	+0.167	+1.905	0.061	0.000	-0.006
	Mask R-CNN + GnTCN	96.896	+0.210	+0.654	0.049	-0.001	-0.003
	Mask R-CNN + EvoSkeleton	96.275	+0.528	+1.325	0.054	-0.002	-0.007
	Mask R-CNN + VideoPose3D	97.935	-0.068	+0.386	0.045	0.000	-0.002
	UDP-Pose + GnTCN	97.790	+0.140	+0.446	0.045	0.000	-0.003
	UDP-Pose + EvoSkeleton	97.645	+0.271	+0.773	0.049	-0.001	-0.005
	UDP-Pose + VideoPose3D	**98.023**	-0.075	+0.345	**0.044**	0.000	-0.002
RI-HJS	PoseRegression + GnTCN	60.381	+0.283	+2.057	0.150	-0.001	-0.008
	PoseRegression + EvoSkeleton	62.475	+0.357	+2.158	0.144	-0.001	-0.006
	PoseRegression + VideoPose3D	58.784	+0.290	+5.029	0.154	0.000	-0.013
	ArtTrack + GnTCN	80.310	+0.864	+1.398	0.106	-0.002	-0.006
	ArtTrack + EvoSkeleton	80.549	+2.079	+2.501	0.107	-0.006	-0.010
	ArtTrack + VideoPose3D	59.736	-0.027	+8.668	0.151	0.000	-0.020
	Mask R-CNN + GnTCN	84.545	+0.771	+1.135	0.098	-0.002	-0.005
	Mask R-CNN + EvoSkeleton	86.485	+2.132	+2.415	0.094	-0.006	-0.010
	Mask R-CNN + VideoPose3D	73.718	-0.152	+3.084	0.124	0.000	-0.008
	UDP-Pose + GnTCN	90.797	+0.551	+1.370	0.083	-0.001	-0.004
	UDP-Pose + EvoSkeleton	**94.436**	+0.695	+1.009	**0.074**	-0.002	-0.005
	UDP-Pose + VideoPose3D	76.357	-0.232	+2.916	0.117	-0.000	-0.007

4.1. Discussion of the Pose Estimation Results

Table 4 shows that the tested models scored much better on the Human3.6M dataset than in the custom RI-HJS datasets in PA-PCK and PA-MPJPE metrics (Figures 13 and 14). Better results on the Human3.6M dataset than on the custom dataset have been expected, given that all 3D models were pretrained on the training set of the Human3.6M dataset.

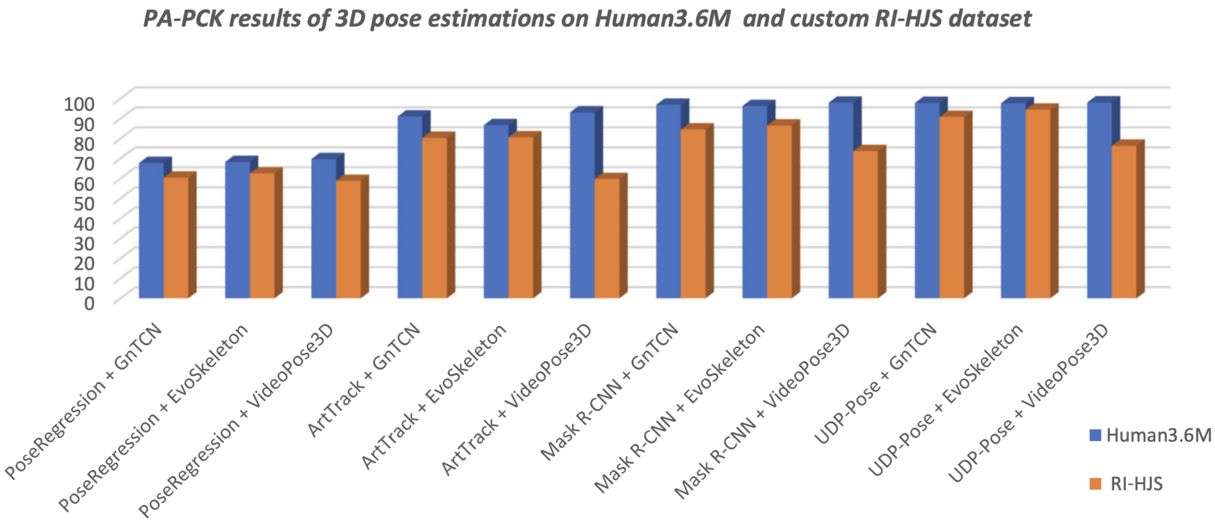


Figure 13: Comparison of the 3D pose estimation model results in terms of PA-PCK on Human 3.6M and custom RI-HJS datasets (higher is better). All models experienced a significant drop in performance on the RI-HJS dataset, except the two-step model UDP-Pose + EvoSkeleton, which retained high accuracy, showing robustness in an unseen environment. It is interesting to note that all two-step models that use VideoPose3D experienced the largest performance drop compared to other models.

The lower performance values for the user-defined dataset indicate that the tested models are not robust enough to be applied to new environments without retraining. Figure 15 shows the differences between the results on the Human 3.6M test set and the results on the user-defined set obtained by the models trained on the Human 3.6M training set. It is interesting to note that the difference in the performance drop ranges from 3% to more than 33%. It should be noted that the UDP-Pose + EvoSkeleton model achieved almost the same high level of performance in the new custom set. In other words, all the tested models had the lowest performance drop when combined with the EvoSkeleton model, which ranged from 3% to a maximum of 10%, suggesting its robustness and its ability to be used in new sports scenes such as the handball scenes

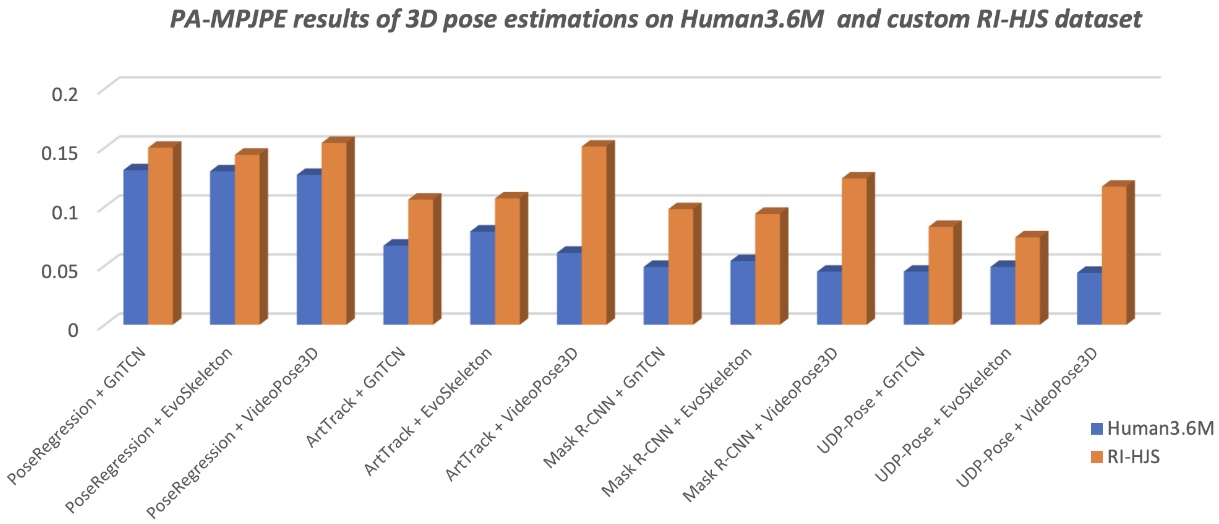


Figure 14: Comparison of the 3D pose estimation model results in terms of PA-MPJPE on Human 3.6M and custom RI-HJS datasets (lower is better). The comparison shows a significant drop in performance on the RI-HJS dataset, which is not surprising given that the models have never seen uncommon poses such as the handball jump-shot from the RI-HJS dataset. Two-step models that use VideoPose3D are more prone to errors due to unseen data, as they have the largest performance drop.

tested. The videoPose3D model, on the other hand, had the largest drop in performance regardless of which model it was used with; more specifically, it had a significant drop in performance of over 20% with all models except PoseRegression, where the drop was also significant but only half as large (about 11%).

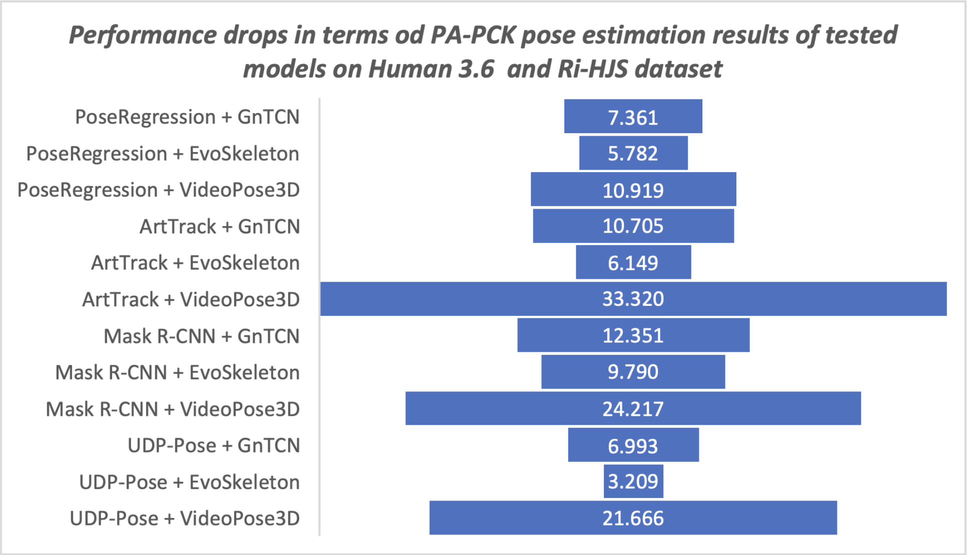


Figure 15: The robustness of the tested 3D models trained on the Human3.6M dataset shown as a difference of obtained results and performance drops between PA-PCK pose estimation results on Human 3.6M and custom RI-HJS datasets.

Overall, the models using UDP-Pose for 2D pose estimation were found to perform better, which is not surprising since the authors reported better results than Mask R-CNN.

Using a method to smooth predicted 3D sequences proved beneficial in most cases, except in the case of VideoPose3D, where it does not seem to improve the predicted sequence, but rather looks like the sequence has already been smoothed directly in VideoPose3D.

Figure 16 shows the improvements in pose prediction sequence after applying Kalman smoothing with respect to PA-PCK on RI-HJS datasets, and Figure 17 shows the improvements after applying Kalman smoothing and retargeting with respect to PA-MPJPE on the Human3.6M datasets.

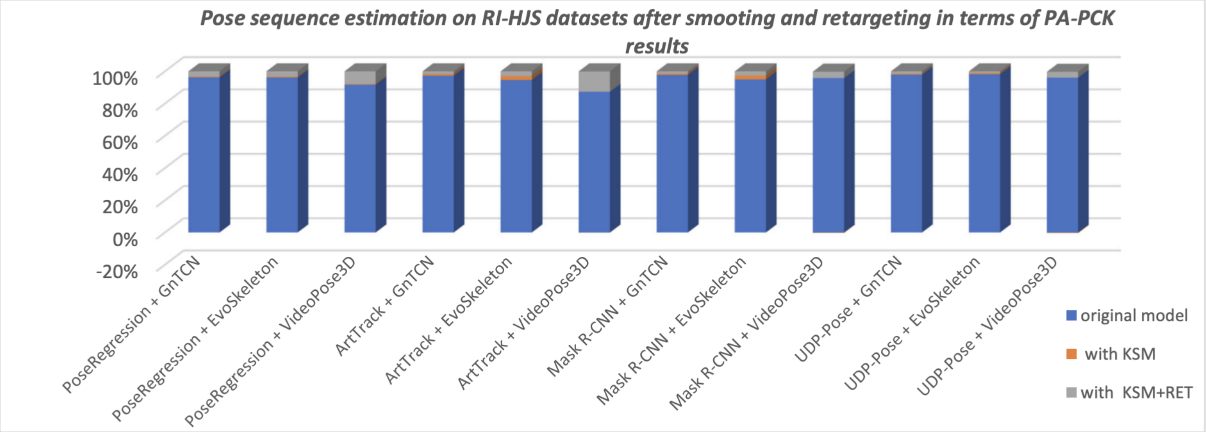


Figure 16: Comparison of pose sequence estimation in terms of PA-PCK on custom RI-HJS datasets (higher score is better). Two-step models that use EvoSkeleton show a significant improvement when using smoothing on the sequence of poses, showing the lack of consistency in the process of "lifting" 2D keypoints to 3D space. When using retargeting on the ground truth and smoothed predicted sequence, the results are significantly improved, indicating that all models lack an understanding of the human skeleton structure, which is especially true in the case of VideoPose3D.

The average improvement using 3D predicted sequence smoothing (KSM) is 0.7% for the PCK metric (i.e., 0.57% on Human3.6M and 0.84% on RI-HJS) and 1.4% for the MPJPE metric (i.e., 1.26% on Human3.6M and 1.52% on RI-HJS). Retargeting to standardize both predicted sequence bone lengths and ground truth improved the overall result in all cases. The average improvement using retargeting and smoothing (KSM + RET) is 3.87% for the PCK metric (i.e., 4.04% on Human3.6M and 3.71% on RI-HJS) and 10.1% for the MPJPE metric (i.e., 12.95% on Human3.6M and 7.36% on RI-HJS). Interestingly, retargeting improved EvoSkeleton's overall score the most on the Human3.6M dataset, but improved VideoPose3D's overall score the most when evaluated

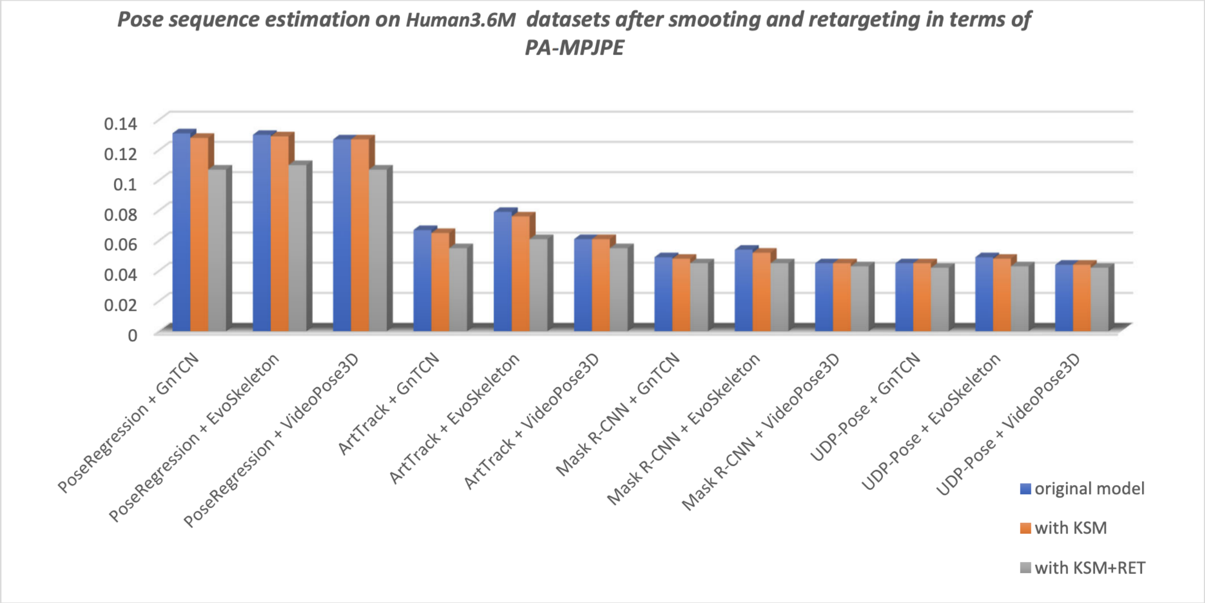


Figure 17: Comparison of pose sequence estimation in terms of PA-MPJPE measure on Human3.6M datasets (lower is better). All models show a slight improvement when using smoothing on the sequence of poses, showing the lack of consistency in the detection location of keypoints and ”lifting” 2D keypoints to the 3D space. An exception to this conclusion is the VideoPose3D model, which constructed a smooth sequence of poses by utilizing temporal information. When using retargeting on the ground truth and smoothed predicted sequence, results are significantly improved, which indicates that all models lack an understanding of the human skeleton structure.

on a custom dataset. This suggests that the models have the potential for performance improvement in constructing valid and consistent poses. Based on these results, we can conclude that of the two-stage models evaluated, the UDP pose + EvoSkeleton proved to be the most robust and stable, achieving the highest overall score on the datasets evaluated.

4.2. Analysis of the Pose Estimation Errors

Analysing the 3D pose estimation images and predictions, we find that the errors are mainly propagated due to poor 2D pose estimation. Poor predictions occur mainly when one or more joints are occluded. Then the 2D pose estimation model usually assigns the position of the invisible joint to the position of the visible joint on the opposite side. Examples of this problem are shown in Figure 18. Another common problem that is not easily understood or explained is the detection of keypoints on the opposite side of the player. The result is usually a valid human structure but rotated 180 degrees, i.e., the left side is swapped with the right and opposite sides, as shown in Figure 19.

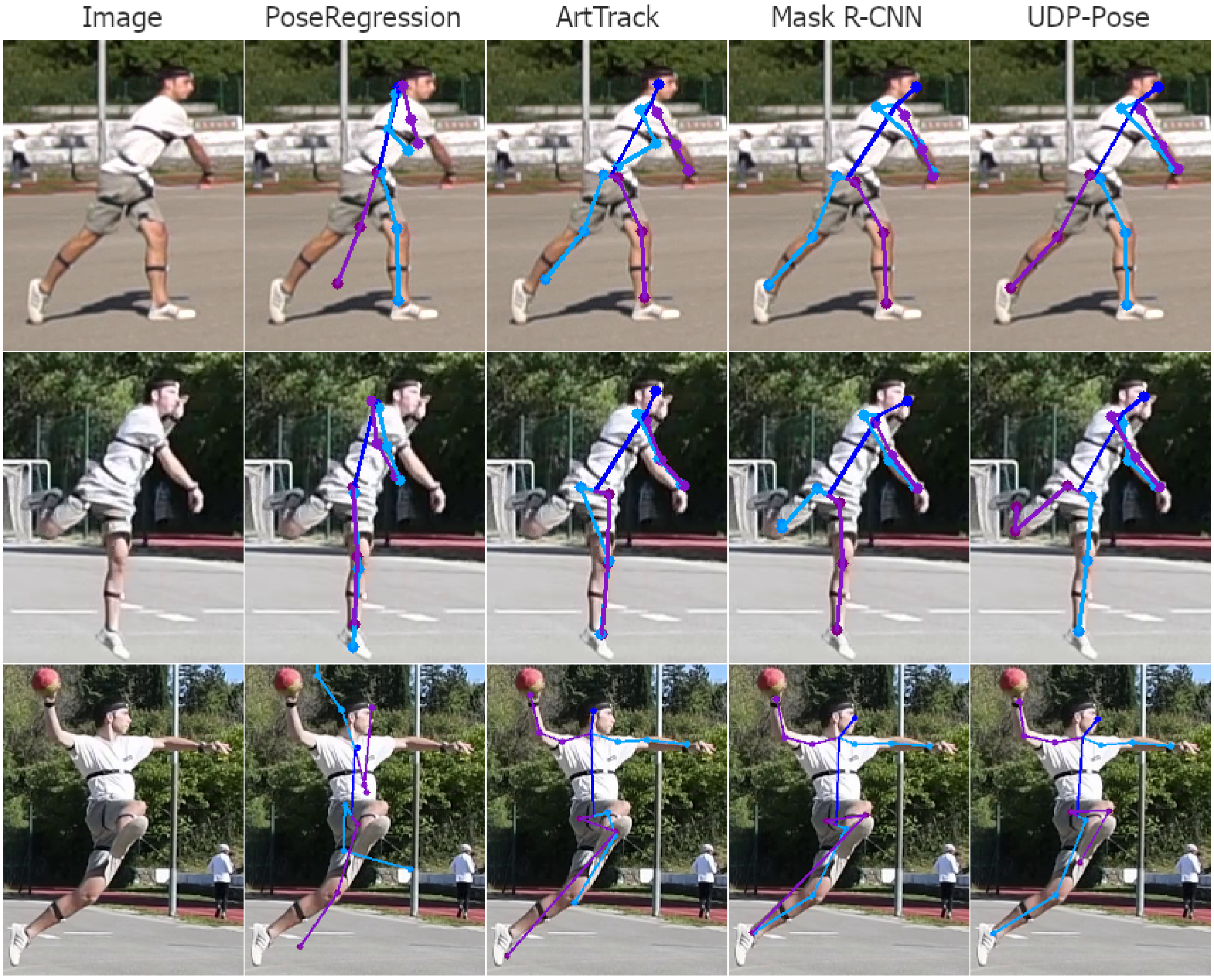


Figure 18: Examples of poor detection of keypoint location that happens mostly because the true keypoint location is occluded or less clear. The right side of the player is coloured purple while the keypoint location is occluded or less clear. The right side of the player is coloured purple while the left side of the person is coloured blue. In the first row, where the left elbow and hand are not visible, methods PoseRegression and ArtTrack incorrectly assume the location, while Mask R-CNN and UDP-Pose placed the left elbow and hand on the right elbow and hand of the player. The second row shows a situation where parts are visible but less clear, where all methods fail to detect the left hand, which is close to the head, while methods ArtTrack and Mask R-CNN miss the right foot. The third row shows situations where methods ArtTrack and Mask R-CNN produced invalid human structures by detecting the right foot on the location of the left foot, while the UDP-Pose almost correctly detected the keypoints. PoseRegression generally did not perform well on uncommon poses such as the handball jump-shot.

With the goal of reducing errors due to missed detection of visible joints, false detection of visible joints, and invalid pose rotation when switching left and right sides, we trained Mask R-CNN and UDP-Pose on the dataset RI-HJS. Both models were trained on 227

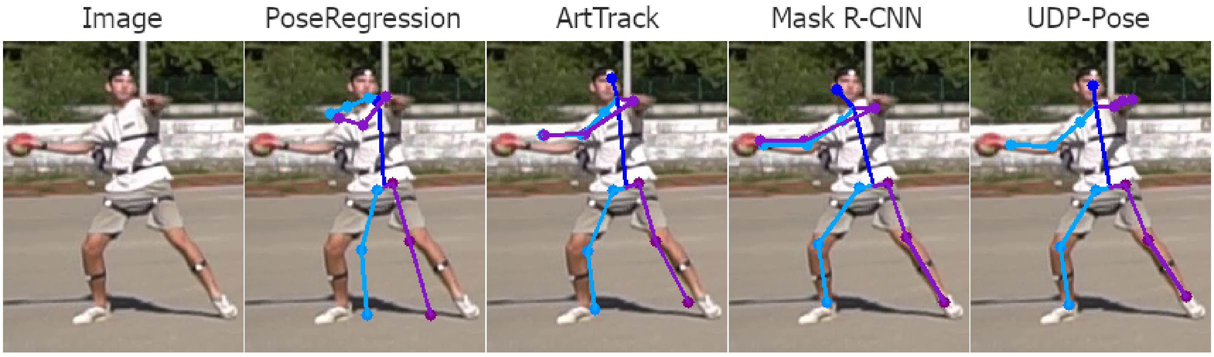


Figure 19: Examples of wrong player side keypoint detection, with an unclear reason for this occurrence. The right side of the player is coloured purple while the left side of the person is coloured blue. While all methods detected almost all keypoints correctly, all methods switched sides of the player, producing an invalid pose. Occurrences of this problem can also be observed on a few keypoints in Figure 18.

images while evaluation was performed on the rest of the dataset. Both models were trained with a learning rate of 0.001 and an Adam optimizer with 30 epochs. We trained only the Mask R-CNN and UDP-Pose because the PoseRegression and ArtTrack models performed poorly in the previous evaluation in Table 4. The results of the evaluation performed on the test set of the RI-HJS dataset are shown in Table 5.

Table 5 shows the results on the test part of the dataset RI-HJS after training the Mask R-CNN and UDP-Pose models on 227 images from the training part of the dataset RI-HJS. The evaluation shows that all two-stage models using models trained on the RIHJS dataset perform significantly better than models not trained on this dataset. Models using Mask R-CNN show an average improvement of 2.79 on the PA-PCK metric and an average improvement of 0.007 on PA-MPJPE. Models using UDP-Pose show an average improvement of 1.06 on the PA-PCK metric and an average improvement of 0.002 on PA-MPJPE. Even with training, the Mask R-CNN model did not achieve the accuracy of UDP-Pose. In addition, EvoSkeleton appears to be the most robust of the 3D models, providing the best results when paired with both 2D models. A graphical representation of the comparative results before and after training the most successful models Mask R-CNN and UDP-Pose on the training set of RI-HJS is given in Figure 20.

Examples of detections after training the Mask R-CNN and UDP Pose models are shown in Figures 21 and 22. Further comparisons between the trained and untrained 2D models are shown in Supplementary Figures S1–S4.

Table 5: Results of model combinations for 3D pose estimations on the custom dataset of players performing handball jump-shot (RI-HJS). The best results are marked in bold. Models Mask R-CNN and UDP-Pose were trained on 227 images from the RI-HJS dataset, while evaluation was performed on the rest of the dataset. Metrics are computed on the normalized poses using the h-norm described in Sections 2.5 and 2.6 on 13 keypoints. KSM in the table is shorthand for Kalman smoother that is applied on the predicted sequence before evaluation, while KSM + RET is shorthand for Kalman smoother applied on the predicted sequence and Retargeting applied both on predicted and ground truth sequences.

Dataset	Models	▲PA-PCK$_{0.15}$	▲PA-PCK$_{0.15}$ + KSM	▲PA-PCK$_{0.15}$ + KSM + RET	▼PA-MPJPE$_{0.15}$	▼PA-MPJPE$_{0.15}$ + KSM	▼PA-MPJPE$_{0.15}$ + KSM + RET
RI-HJS	Mask R-CNN + GnTCN	87.574	+0.644	+0.970	0.090	-0.002	-0.004
	Mask R-CNN + EvoSkeleton	89.562	+1.875	+2.004	0.086	-0.006	-0.009
	Mask R-CNN + VideoPose3D	76.970	-0.196	+2.258	0.118	0.000	-0.007
	UDP-Pose + GnTCN	92.205	+0.486	+0.974	0.080	-0.001	-0.003
	UDP-Pose + EvoSkeleton	**94.462**	+0.537	+0.303	**0.073**	-0.002	-0.003
	UDP-Pose + VideoPose3D	78.122	-0.267	+2.053	0.115	-0.000	-0.006

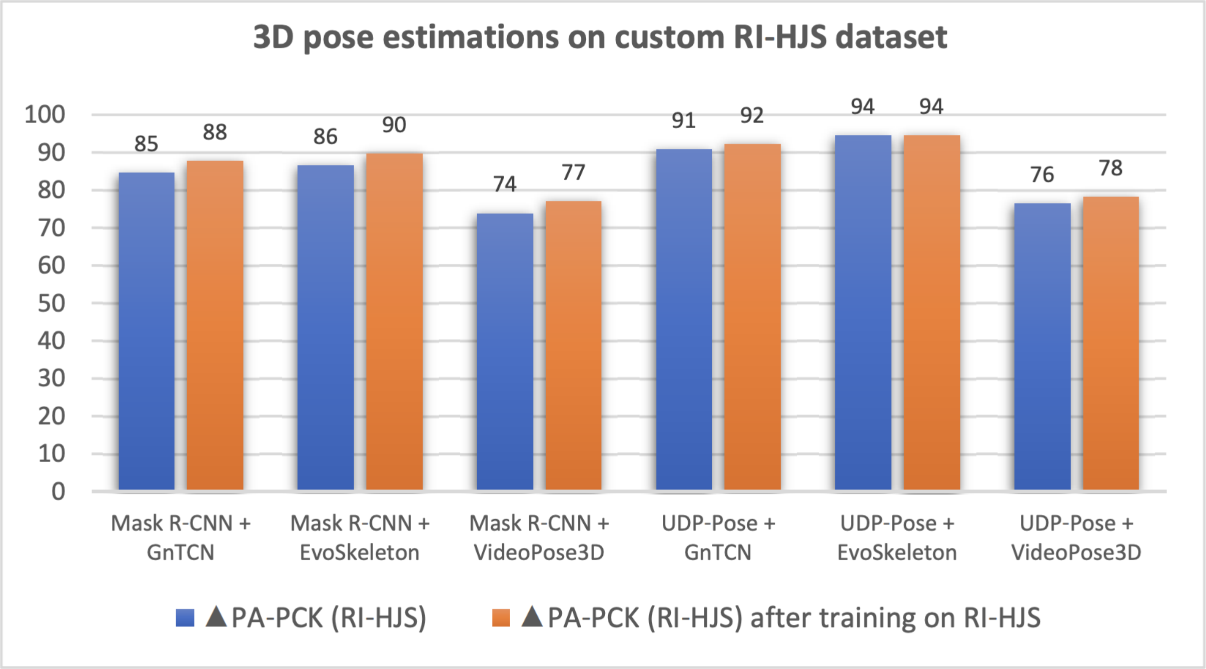


Figure 20: Comparison of pose sequence estimation in terms of PA-PCK on custom RI-HJS datasets before and after additional training of the Mask R-CNN and UDP-Pose models on training part on RI-HJS dataset (higher score is better).

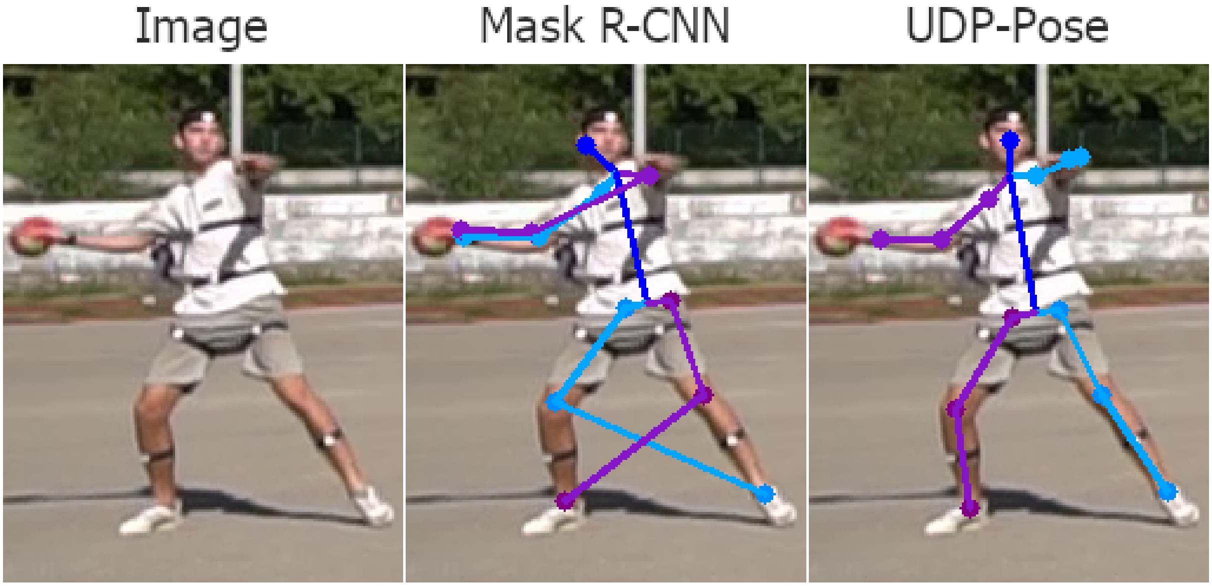


Figure 21: Examples of detection after training on the 227 images of the RI-HJS dataset. The right side of the player is coloured purple while the left side of the person is coloured blue. Untrained models made a mistake and switched the players' sides, shown in Figure 19. After training, UDP-Pose successfully detected keypoints on the correct sides, while Mask R-CNN did not manage to detect all keypoint sides correctly.

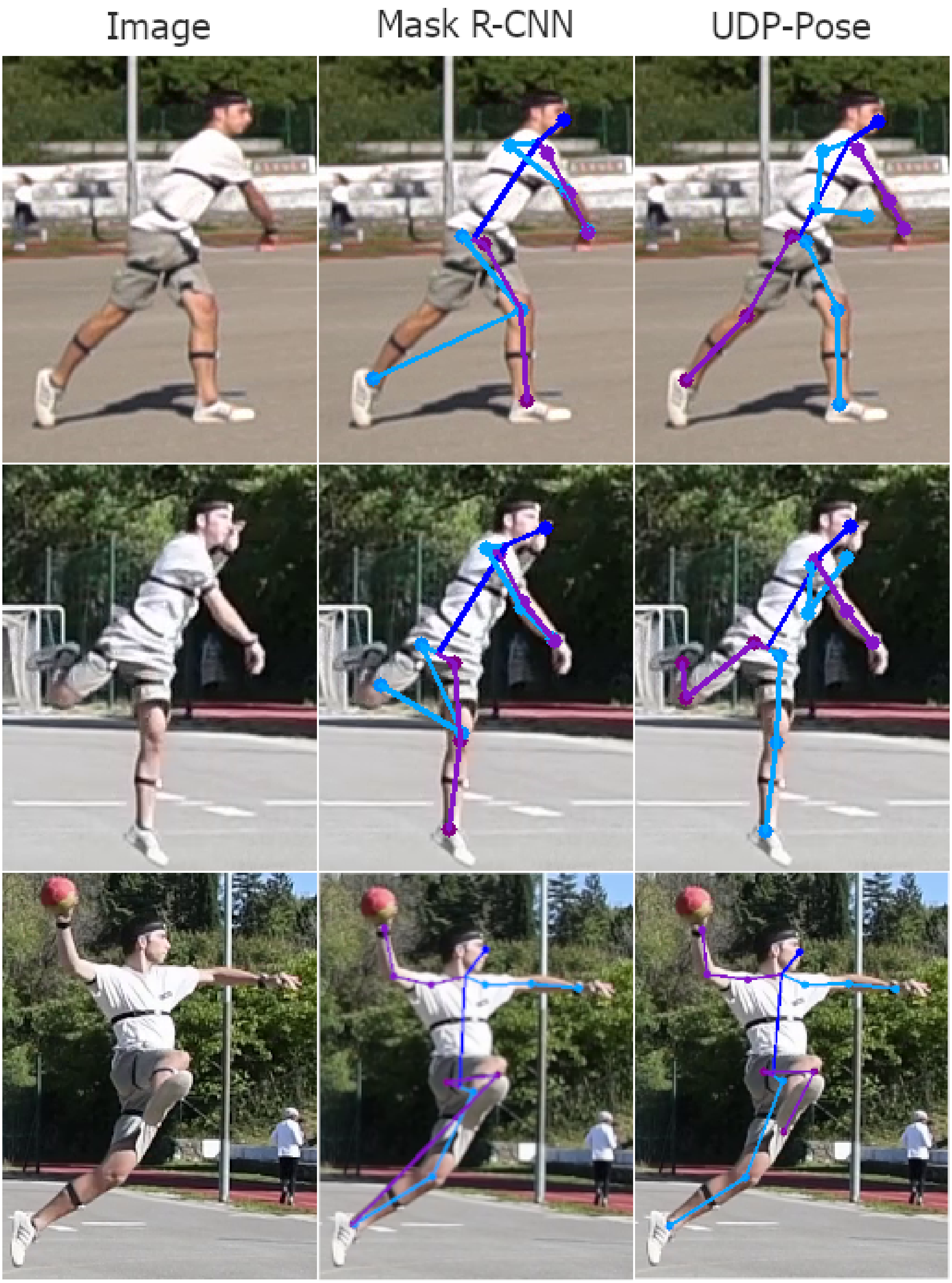


Figure 22: Examples of detection after training on the 227 images of the RI-HJS dataset. The right side of the player is coloured purple while the left side of the person is coloured blue. Untrained models missed detection when the left hand was hidden or less clear, as shown in Figure 18. After training, UDP-Pose successfully detected the left hand on the second row, while on the first row, it made a reasonable guess of the hand position. Mask R-CNN performed worse on both examples after training, wrongly detecting the right knee location on the left knee.

5. Evaluation of Tracking Methods

We chose some well-known and well-performing methods for tracking and at least one method for each methodology. Tracking methods were evaluated against the following datasets: DanceTrack [83], SportsMOT [90], MOT17 [54], and our custom dataset RI-HB-PT.

The goal of this experiment is to evaluate the tracking performance of the methods that provide the best overall results in an unseen environment. We considered the tracking methods CentroidKF (`https://github.com/adipandas/multi-object-tracker`, accessed on 1 April 2022) [47] (Kalman filter motion tracking), SORT (`https://github.com/adipandas/multi-object-tracker`, accessed on 1 April 2022) [8] (Kalman filter motion tracking), DeepSORT (`https://github.com/abhyantrika/nanonets_object_tracking`, accessed on 1 April 2022) [91] (motion and feature tracking), FlowTracker (`https://github.com/hitottiez/sdoftracker`, accessed on 1 April 2022) [61] (optical flow motion tracking), and Tracktor++ (`https://github.com/phil-bergmann/tracking_wo_bnw`, accessed on 1 April 2022) [28] (motion and feature tracking). None of the selected trackers have seen previously tested datasets, so the evaluation was performed on all unseen datasets. The evaluation was performed on the training part of the datasets, since the annotations and ground truths are only available for this part. The final results are shown in Table 6, using the metrics described in Section 3.4.

5.1. Discussion of Tracking Results

Table 6 shows that there is no clear winner that performs best in all datasets or metrics. DeepSORT performs best on the re-identification task and significantly outperforms the other methods on all datasets except RI-HB-PT in terms of IDF1, Identity Switching (IDsw), and Mostly Tracked targets (MT). For the MOTP metric, Tracktor++ and SORT achieve the best overall results, but based on the MOTA metrics, Tracktor++ and DeepSORT share first place. FlowTracker, surprisingly, scores significantly lower than the compared methods, but performs better in datasets where the camera is still and there are fewer entrances and exits from the scene. The CentroidKF and SORT methods rely on the Kalman filter and are very simple, but perform well in certain scenarios. It appears that camera motion strongly influences these methods, as they perform significantly better when the camera is stationary than when it is moving. Table 7 shows the averaged results for all datasets and confirms the observations described earlier. Examples where the models performed poorly are shown in Supplementary Figures S5–S9.

Table 6: Results of tracking methods on DanceTrack, SportsMOT, MOT17, and a custom RI-HB-PT dataset of players practicing various handball actions during a practice session. The best results according each metrics are marked in bold. Metrics are computed as described in Section 3.4.

Dataset	Models	▲MOTA	▲MOTP	▲IDF1	▼IDsw	▲Recall	▲Precision	▲MT	▼ML
DanceTrack	CentroidKF	**76.9**	0.201	7.9	47,550	95.3	95.3	409	**0**
	SORT	27.5	**0.313**	11.7	16,052	66.1	66.1	20	**0**
	DeepSORT	68.0	0.160	**43.0**	**4717**	**97.6**	77.5	**418**	**0**
	FlowTracker	38.1	0.262	13.5	10,448	66.4	72.4	59	1
	Tracktor++	67.4	0.262	29.4	18, 255	75.1	**96.9**	166	**0**
SportsMOT	CentroidKF	15.4	0.247	5.9	43,469	64.7	64.7	133	**0**
	SORT	15.6	0.254	6.2	48,186	65.5	65.5	4	**0**
	DeepSORT	**79.9**	0.149	**63.7**	**2939**	**99.2**	84.4	**635**	**0**
	FlowTracker	25.4	0.281	12.2	9873	62.6	64.7	16	3
	Tracktor++	64.6	**0.297**	42.9	7949	78.0	**89.8**	298	**0**
MOT17	CentroidKF	60.7	0.140	49.0	11,200	83.1	83.1	366	26
	SORT	56.5	0.159	52.4	9909	80.7	80.7	220	**2**
	DeepSORT	**71.0**	0.062	**70.9**	**1159**	**90.5**	90.5	**664**	24
	FlowTracker	37.1	0.156	38.2	2093	47.4	83.6	162	310
	Tracktor++	64.8	**0.258**	64.8	3263	73.4	**91.3**	356	115
RI-HB-PT	CentroidKF	69.4	0.231	19.3	9912	87.0	87.0	310	15
	SORT	49.0	**0.261**	21.4	6299	76.0	76.0	225	16
	DeepSORT	70.2	0.063	38.7	2276	**99.7**	77.8	**353**	**4**
	FlowTracker	68.0	0.222	16.6	3770	84.6	85.1	125	145
	Tracktor++	**92.4**	0.202	**49.6**	**1346**	94.8	**98.1**	176	137

Table 7: Averaged results of trackers across all datasets (DanceTrack, SportsMOT, MOT17, RI-HB-PT), i.e., averaged results from Table 6. The best results according each metrics are marked in bold.

Models	▲MOTA	▲MOTP	▲IDF1	▼IDsw	▲Recall	▲Precision	▲MT	▼ML
CentroidKF	55.60	0.204	20.52	28,057	82.52	82.52	304	10
SORT	37.15	0.246	22.92	20,111	72.07	72.07	117	**4**
DeepSORT	72.15	0.108	**54.07**	**2772**	**96.75**	80.60	**517**	7
FlowTracker	42.15	0.230	20.12	6546	65.25	76.45	88	115
Tracktor++	**72.30**	**0.254**	46.67	7703	80.32	**93.52**	249	63

6. Conclusions

In this work, we evaluated 12 selected 2-stage models for 3D pose estimation and methods for smoothing and retargeting the sequences. We reported the results and concluded that the application of smoothing and retargeting methods significantly improves the performance of the models. We also evaluated the performance of the two-stage model on a custom dataset to assess its robustness in different/unknown environments. The results of this evaluation are surprising in that most pipelines showed significant performance degradation; only pipelines based on EvoSkeleton had the smallest degradation. However, the UDP-Pose + EvoSkeleton and UDP-Pose + GnTCN models were able to achieve equally high values in both familiar and unfamiliar environments. They achieved an accuracy of correctly estimated body parts of over 90% and a mean joint position error of less than 0.08%, which undoubtedly enables the use of these models for pose estimation in dynamic scenes, such as handball sports.

The greatest performance gain for the models appears to be in constructing good and consistent poses, as smoothing the time series of keypoints over the predicted sequence and retargeting the poses consistently improved the overall score. The improvement in results from smoothing the predicted 3D sequence was seen in the accuracy of the estimated body parts (according to the PCK metric, 0.57% in the Human3.6M dataset and 0.84% in the RI-HJS dataset) and in the reduction of error in the joint position estimation (according to the MPJPE metric, by 1.26% in the Human3.6M dataset and 1.52% in the RI-HJS dataset).

In addition, the retargeting procedure used to normalize the data using the standardized bone length improved the overall score by approximately 4% in all cases in terms of the accuracy of the estimated body parts (i.e., PCK metric 4.04% on Human3.6M and 3.71% on RI-HJS). In addition, the mean error of joint position was reduced by 10% (i.e., according to MPJPE metric 12.95% on Human3.6M and 7.36% on RI-HJS). It is important to note that the performance of top-down pose estimation methods can be affected by object detector performance (e.g., by generating invalid bounding boxes). The detailed performance effects of different object detectors on pose estimation methods are beyond the scope of this work, but may be investigated in future work.

To track the poses of one athlete while acting, poses must be collected throughout the video and mapped on consecutive frames with an algorithm. We selected five stateof-the-art tracking methods and evaluated them against public and user-defined datasets. The main finding after the evaluation is that there is no particular method that performs best in all tested scenarios. However, the DeepSORT method outperforms the rest of the methods in most of the datasets, except for our custom dataset RI-HB-PT, especially in terms of IDF1, Identity switch (IDsw), and Mostly Tracked targets (MT). On the other hand, camera motion seems to strongly affect methods based on the Kalman filter or optical flow, where feature-based tracking methods show their strength. Based on the averaged overall results, we conclude that Tracktor++ and DeepSORT methods provide promising results for tracking people represented by skeletons in dynamic sports scenarios. Therefore, these methods should be considered in the definition of athlete action recognition models.

The experiment has shown that existing state-of-the-art methods for pose estimation already perform satisfactorily and can be used for estimating the poses of a single athlete in individual or team sports. However, for more complex tasks such as tracking more athletes in team sports and comparing athletes' performances or actions, where multi-object tracking methods are to be used, further research and development of methods are needed to successfully use them in dynamic environments such as sports scenes.

Supplementary Materials: The following are available online at `https://www.mdpi.com/article/10.3390/jimaging8110308/s1`, Figure S1: Example of 3D sequences produced by three 3D HPE models and Mask R-CNN, Figure S2: Example of 3D sequences produced by three 3D HPE models and trained Mask R-CNN on RI-HJS dataset, Figure S3: Example of 3D sequences produced by three 3D HPE models and UDP-Pose, Figure S4: Example of 3D sequences produced by three 3D HPE models and trained UDP-Pose on RI-HJS dataset, Figure S5: Example of scenario where CentroidKF tracker performs poorly, Figure S6: Example of scenario where SORT tracker performs poorly, Figure S7: Example of scenario where FlowTracker tracker performs poorly, Figure S8: Example of scenario where DeepSORT tracker performs poorly, Figure S9: Example of scenario where Tracktor++ tracker performs poorly.

Author Contributions: Conceptualization, R.Š. and M.I.-K.; methodology, R.Š. and M.I.-K.; software, R.Š.; validation, R.Š. and M.I.-K.; formal analysis, R.Š. and M.I.-K.;

investigation, R.Š.; resources, R.Š. and M.I.-K.; data curation, R.Š.; writing—original draft preparation, R.Š.; writing—review and editing, M.I.-K.; visualization, R.Š. and M.I.-K.; supervision, M.I.-K.; project administration M.I.-K.; funding acquisition, M.I.-K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Croatian Science Foundation under the project IP-2016-06-8345, "Automatic recognition of actions and activities in multimedia content from the sports domain" (RAASS), and by the University of Rijeka (project number 18-222).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy restrictions.

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] Kfir Aberman et al. "Skeleton-aware networks for deep motion retargeting". In: *ACM Transactions on Graphics (TOG)* 39.4 (2020), pp. 62–1.

[2] Mykhaylo Andriluka et al. "2d human pose estimation: New benchmark and state of the art analysis". In: *Proceedings of the IEEE Conference on computer Vision and Pattern Recognition*. 2014, pp. 3686–3693.

[3] Hexin Bai et al. "GMOT-40: A benchmark for generic multiple object tracking". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, pp. 6719–6728.

[4] A Balasundaram, S Ashok Kumar, and S Magesh Kumar. “Optical flow based object movement tracking”. In: *Int. J. Eng. Adv. Technol.(IJERT)* 9 (2019), pp. 3913–3916.

[5] Qian Bao et al. “Pose-guided tracking-by-detection: Robust multi-person pose tracking”. In: *IEEE Transactions on Multimedia* 23 (2020), pp. 161–175.

[6] Valentin Bazarevsky et al. “BlazePose: On-device Real-time Body Pose tracking”. In: *arXiv preprint arXiv:2006.10204* (2020).

[7] Keni Bernardin and Rainer Stiefelhagen. “Evaluating multiple object tracking performance: the clear mot metrics”. In: *EURASIP Journal on Image and Video Processing* 2008 (2008), pp. 1–10.

[8] Alex Bewley et al. “Simple online and realtime tracking”. In: *2016 IEEE international conference on image processing (ICIP)*. IEEE. 2016, pp. 3464–3468.

[9] Moshe Blank et al. “Actions as space-time shapes”. In: *Tenth IEEE International Conference on Computer Vision (ICCV’05) Volume 1*. Vol. 2. IEEE. 2005, pp. 1395–1402.

[10] Matija Buric, Marina Ivasic-Kos, and Miran Pobar. “Player tracking in sports videos”. In: *2019 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*. IEEE. 2019, pp. 334–340.

[11] Zhe Cao et al. “Realtime multi-person 2d pose estimation using part affinity fields”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 7291–7299.

[12] Joao Carreira et al. “Human pose estimation with iterative error feedback”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 4733–4742.

[13] Ching-Hang Chen and Deva Ramanan. “3d human pose estimation = 2d pose estimation + matching”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 7035–7043.

[14] Xianjie Chen and Alan Yuille. “Articulated pose estimation by a graphical model with image dependent pairwise relations”. In: *arXiv preprint arXiv:1407.3399* (2014).

[15] Yilun Chen et al. “Cascaded pyramid network for multi-person pose estimation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 7103–7112.

[16] Yu Chen et al. “Adversarial posenet: A structure-aware convolutional network for human pose estimation”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 1212–1221.

[17] Yu Cheng et al. “Graph and Temporal Convolutional Networks for 3D Multi-person Pose Estimation in Monocular Videos”. In: *Proceedings of the AAAI Conference on Artificial Intelligence* 35.2 (May 2021), pp. 1157–1165.

[18] Xiao Chu et al. “Multi-context attention for human pose estimation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 1831–1840.

[19] Qi Dang et al. “Deep learning based 2d human pose estimation: A survey”. In: *Tsinghua Science and Technology* 24.6 (2019), pp. 663–676.

[20] Achal Dave et al. “Tao: A large-scale benchmark for tracking any object”. In: *European conference on computer vision*. Springer. 2020, pp. 436–454.

[21] Jia Deng et al. “Imagenet: A large-scale hierarchical image database”. In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee. 2009, pp. 248–255.

[22] Andreas Doering, Umar Iqbal, and Juergen Gall. “Joint flow: Temporal flow fields for multi person tracking”. In: *arXiv preprint arXiv:1805.04596* (2018).

[23] Hao-Shu Fang et al. “Rmpe: Regional multi-person pose estimation”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 2334–2343.

[24] Fahime Farahi and Hadi Sadoghi Yazdi. “Probabilistic Kalman filter for moving object tracking”. In: *Signal Processing: Image Communication* 82 (2020), p. 115751.

[25] Vittorio Ferrari, Manuel Marin-Jimenez, and Andrew Zisserman. “Progressive search space reduction for human pose estimation”. In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE. 2008, pp. 1–8.

[26] Pramod R Gunjal et al. “Moving object tracking using kalman filter”. In: *2018 International Conference On Advances in Communication and Computing Technology (ICACCT)*. IEEE. 2018, pp. 544–547.

[27] Kaiming He et al. “Mask r-cnn”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2961–2969.

[28] Kristina Host, Marina Ivasic-Kos, and Miran Pobar. “Tracking Handball Players with the DeepSORT Algorithm.” In: *Proceedings of the International Conference on Pattern Recognition Applications and Methods, ICPRAM*. 2020, pp. 593–599.

[29] Junjie Huang et al. “Optical flow based real-time moving object detection in unconstrained scenes”. In: *arXiv preprint arXiv:1807.04890* (2018).

[30] Junjie Huang et al. “The devil is in the details: Delving into unbiased data processing for human pose estimation”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 5700–5709.

[31] Mostafa S Ibrahim et al. “A hierarchical deep temporal model for group activity recognition”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 1971–1980.

[32] Eldar Insafutdinov et al. “Arttrack: Articulated multi-person tracking in the wild”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 6457–6465.

[33] Eldar Insafutdinov et al. “Deepercut: A deeper, stronger, and faster multi-person pose estimation model”. In: *European Conference on Computer Vision*. Springer. 2016, pp. 34–50.

[34] Catalin Ionescu et al. “Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments”. In: *IEEE transactions on pattern analysis and machine intelligence* 36.7 (2013), pp. 1325–1339.

[35] Umar Iqbal and Juergen Gall. “Multi-person pose estimation with local joint-to-person associations”. In: *European Conference on Computer Vision*. Springer. 2016, pp. 627–642.

[36] Umar Iqbal, Anton Milan, and Juergen Gall. “Posetrack: Joint multi-person pose estimation and tracking”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 2011–2020.

[37] Marina Ivasic-Kos and Miran Pobar. “Building a labeled dataset for recognition of handball actions using mask R-CNN and STIPS”. In: *2018 7th European Workshop on Visual Information Processing (EUVIP)*. IEEE. 2018, pp. 1–6.

[38] Kiran Kale, Sushant Pawar, and Pravin Dhulekar. “Moving object tracking using optical flow and motion vector estimation”. In: *2015 4th international conference on reliability, infocom technologies and optimization (ICRITO)(trends and future directions)*. IEEE. 2015, pp. 1–6.

[39] Rudolph Emil Kalman. “A new approach to linear filtering and prediction problems”. In: (1960).

[40] Wenjing Kang et al. “Online Multiple Object Tracking with Recurrent Neural Networks and Appearance Model”. In: *2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. IEEE. 2020, pp. 34–38.

[41] Andrej Karpathy et al. “Large-scale video classification with convolutional neural networks”. In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2014, pp. 1725–1732.

[42] Shyamgopal Karthik, Ameya Prabhu, and Vineet Gandhi. “Simple unsupervised multi-object tracking”. In: *arXiv preprint arXiv:2006.02609* (2020).

[43] Longteng Kong et al. “Online Multiple Athlete Tracking with Pose-Based Long-Term Temporal Dependencies”. In: *Sensors* 21.1 (2021), p. 197.

[44] Harold W Kuhn. “The Hungarian method for the assignment problem”. In: *Naval research logistics quarterly* 2.1-2 (1955), pp. 83–97.

[45] Shichao Li et al. “Cascaded Deep Monocular 3D Human Pose Estimation With Evolutionary Training Data”. In: *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020.

[46] Sijin Li and Antoni B Chan. "3d human pose estimation from monocular images with deep convolutional neural network". In: *Asian Conference on Computer Vision*. Springer. 2014, pp. 332–347.

[47] Xin Li et al. "A multiple object tracking method using Kalman filter". In: *The 2010 IEEE international conference on information and automation*. IEEE. 2010, pp. 1862–1866.

[48] Yuan Li, Chang Huang, and Ram Nevatia. "Learning to associate: Hybridboosted multi-target tracker for crowded scene". In: *2009 IEEE conference on computer vision and pattern recognition*. IEEE. 2009, pp. 2953–2960.

[49] Tsung-Yi Lin et al. "Microsoft coco: Common objects in context". In: *European conference on computer vision*. Springer. 2014, pp. 740–755.

[50] F Lotfi, V Ajallooeian, and HD Taghirad. "Robust object tracking based on recurrent neural networks". In: *2018 6th RSI International Conference on Robotics and Mechatronics (IcRoM)*. IEEE. 2018, pp. 507–511.

[51] Bruce D Lucas, Takeo Kanade, et al. "An iterative image registration technique with an application to stereo vision". In: Vancouver. 1981.

[52] Diogo C Luvizon, Hedi Tabia, and David Picard. "Human pose regression by combining indirect part detection and contextual information". In: *Computers & Graphics* 85 (2019), pp. 15–22.

[53] Julieta Martinez et al. "A simple yet effective baseline for 3d human pose estimation". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 2640–2649.

[54] Anton Milan et al. "MOT16: A benchmark for multi-object tracking". In: *arXiv preprint arXiv:1603.00831* (2016).

[55] Jean-Sébastien Monzani et al. "Using an intermediate skeleton and inverse kinematics for motion retargeting". In: *Computer Graphics Forum*. Vol. 19. 3. Wiley Online Library. 2000, pp. 11–19.

[56] Nima Najafzadeh, Mehran Fotouhi, and Shohreh Kasaei. "Object tracking using Kalman filter with adaptive sampled histogram". In: *2015 23rd Iranian Conference on Electrical Engineering*. IEEE. 2015, pp. 781–786.

[57] Alejandro Newell, Zhiao Huang, and Jia Deng. "Associative embedding: End-to-end learning for joint detection and grouping". In: *arXiv preprint arXiv:1611.05424* (2016).

[58] Alejandro Newell, Kaiyu Yang, and Jia Deng. "Stacked hourglass networks for human pose estimation". In: *European conference on computer vision*. Springer. 2016, pp. 483–499.

[59] Juan Carlos Niebles, Chih-Wei Chen, and Li Fei-Fei. "Modeling temporal structure of decomposable motion segments for activity classification". In: *European conference on computer vision*. Springer. 2010, pp. 392–405.

[60] Guanghan Ning et al. "Spatially supervised recurrent convolutional neural networks for visual object tracking". In: *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE. 2017, pp. 1–4.

[61] Hitoshi Nishimura et al. "SDOF-Tracker: fast and accurate multiple human tracking by skipped-detection and optical-flow". In: *arXiv preprint arXiv:2106.14259* (2021).

[62] Hua-wei Pan et al. "A method of real-time human motion retargeting for 3D terrain adaption". In: *IEEE Conference Anthology*. IEEE. 2013, pp. 1–5.

[63] George Papandreou et al. "Towards accurate multi-person pose estimation in the wild". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 4903–4911.

[64] Georgios Pavlakos et al. "Coarse-to-fine volumetric prediction for single-image 3D human pose". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 7025–7034.

[65] Dario Pavllo et al. "3d human pose estimation in video with temporal convolutions and semi-supervised training". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 7753–7762.

[66] Leonid Pishchulin et al. "Deepcut: Joint subset partition and labeling for multi person pose estimation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 4929–4937.

[67] Miran Pobar and Marina Ivasic-Kos. “Mask R-CNN and Optical flow based method for detection and marking of handball actions”. In: *2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. IEEE. 2018, pp. 1–6.

[68] Lorenzo Porzi et al. “Learning multi-object tracking and segmentation from automatic annotations”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 6846–6855.

[69] Yaadhav Raaj et al. “Efficient online multi-person 2d pose tracking with recurrent spatio-temporal affinity fields”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 4620–4628.

[70] Ilija Radosavovic et al. “Data distillation: Towards omni-supervised learning”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 4119–4128.

[71] Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. “Reconstructing 3d human pose from 2d image landmarks”. In: *European conference on computer vision*. Springer. 2012, pp. 573–586.

[72] Mir Rayat Imtiaz Hossain and James J Little. “Exploiting temporal information for 3D pose estimation”. In: *arXiv e-prints* (2017), arXiv–1711.

[73] Shaoqing Ren et al. “Faster r-cnn: Towards real-time object detection with region proposal networks”. In: *arXiv preprint arXiv:1506.01497* (2015).

[74] Ergys Ristani et al. “Performance measures and a data set for multi-target, multi-camera tracking”. In: *European conference on computer vision*. Springer. 2016, pp. 17–35.

[75] Matteo Ruggero Ronchi and Pietro Perona. “Benchmarking and error diagnosis in multi-instance pose estimation”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 369–378.

[76] Seyed Morteza Safdarnejad et al. “Sports videos in the wild (svw): A video dataset for sports analysis”. In: *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*. Vol. 1. IEEE. 2015, pp. 1–7.

[77] Faegheh Sardari, Adeline Paiement, and Majid Mirmehdi. "View-invariant pose analysis for human movement assessment from rgb data". In: *International Conference on Image Analysis and Processing*. Springer. 2019, pp. 237–248.

[78] Christian Schuldt, Ivan Laptev, and Barbara Caputo. "Recognizing human actions: a local SVM approach". In: *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.* Vol. 3. IEEE. 2004, pp. 32–36.

[79] Lucas Smaira et al. "A short note on the kinetics-700-2020 human action dataset". In: *arXiv preprint arXiv:2010.10864* (2020).

[80] Young-min Song and Moongu Jeon. "Online Multi-Object Tracking and Segmentation with GMPHD Filter and Simple Affinity Fusion". In: *arXiv preprint arXiv:2009.00100* (2020).

[81] Khurram Soomro and Amir R Zamir. "Action recognition in realistic sports videos". In: *Computer vision in sports*. Springer, 2014, pp. 181–208.

[82] S Sudhakar et al. "Efficacy of 6 week Plyometric training on agility performance in collegiate male basketball players". In: *Indian Journal of Physiotherapy & Occupational Therapy* 2.2 (2016), pp. 1–8.

[83] Peize Sun et al. "Dancetrack: Multi-object tracking in uniform appearance and diverse motion". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 20993–21002.

[84] ShiJie Sun et al. "Deep affinity network for multiple object tracking". In: *IEEE transactions on pattern analysis and machine intelligence* 43.1 (2019), pp. 104–119.

[85] Xiao Sun et al. "Compositional human pose regression". In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 2602–2611.

[86] Bugra Tekin et al. "Structured prediction of 3d human pose with deep neural networks". In: *arXiv preprint arXiv:1605.05180* (2016).

[87] Hüseyin Temiz, Berk Gökberk, and Lale Akarun. "Multi-view reconstruction of 3D human pose with procrustes analysis". In: *2019 Ninth International Conference on Image Processing Theory, Tools and Applications (IPTA)*. IEEE. 2019, pp. 1–5.

[88] Alexander Toshev and Christian Szegedy. “Deeppose: Human pose estimation via deep neural networks”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014, pp. 1653–1660.

[89] Paul Voigtlaender et al. “Mots: Multi-object tracking and segmentation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 7942–7951.

[90] Limin Wang et al. *SportsMOT*. `https://github.com/MCG-NJU/SportsMOT`. 2022.

[91] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. “Simple online and realtime tracking with a deep association metric”. In: *2017 IEEE international conference on image processing (ICIP)*. IEEE. 2017, pp. 3645–3649.

[92] Yuliang Xiu et al. “Pose flow: Efficient online pose tracking”. In: *arXiv preprint arXiv:1802.00977* (2018).

[93] Qingwen Xu et al. “An Optical Flow Based Multi-Object Tracking Approach Using Sequential Convex Programming”. In: *2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV)*. IEEE. 2020, pp. 1216–1221.

[94] Kota Yamaguchi et al. “Who are you with and where are you going?” In: *CVPR 2011*. IEEE. 2011, pp. 1345–1352.

[95] Yi Yang and Deva Ramanan. “Articulated human detection with flexible mixtures of parts”. In: *IEEE transactions on pattern analysis and machine intelligence* 35.12 (2012), pp. 2878–2890.

[96] Yashu Zhang, Yue Ming, and Runqing Zhang. “Object detection and tracking based on recurrent neural networks”. In: *2018 14th IEEE International Conference on Signal Processing (ICSP)*. IEEE. 2018, pp. 338–343.

[97] Yifu Zhang et al. “FairMOT: On the fairness of detection and re-identification in multiple object tracking”. In: *arXiv e-prints* (2020), arXiv–2004.

[98] Hang Zhao et al. “Hacs: Human action clips and segments dataset for recognition and temporal localization”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 8668–8678.

[99] Xiaowei Zhou et al. “Monocap: Monocular human motion capture using a cnn coupled with a geometric prior”. In: *IEEE transactions on pattern analysis and machine intelligence* 41.4 (2018), pp. 901–914.

[100] Youding Zhu, Behzad Dariush, and Kikuo Fujimura. “Kinematic self retargeting: A framework for human pose estimation”. In: *Computer vision and image understanding* 114.12 (2010), pp. 1362–1375.

D. Analysis of Multi-Person Pose Forecasting Models on Handball Actions

This work was published as: Romeo Šajina, Marina Ivašić-Kos, 2024. Analysis of Multi-Person Pose Forecasting Models on Handball Actions. 2024 8th International Conference on Computer, Software and Modeling (ICCSM), ISBN:979-8-3503-6714-0, DOI: 10.1109/ICCSM63823.2024.00018

https://ieeexplore.ieee.org/document/10638569

1. Introduction

Pose forecasting is a task in computer vision that involves predicting future poses based on a sequence of previous pose observations. The goal is to anticipate a person's movement or behavior over time, which has applications in action recognition, motion analysis, and human-computer interaction. Earlier research in pose forecasting primarily focused on single-person scenarios, where the objective is to predict the future poses of a single individual given the sequence of previous poses. This task, though challenging, laid the groundwork for subsequent advancements in more complex scenarios.

Applications of pose forecasting in sports analytics, surveillance, and social robotics demanded more comprehensive analyses of human interactions, shifting the focus toward multi-person pose forecasting. This presents a more challenging problem, as it requires models to capture intricate dependencies and interactions between multiple individuals in a scene. Multi-person pose forecasting involves predicting the future poses of multiple individuals simultaneously, considering their spatial and temporal relationships.

This paper focuses on the challenges and advancements in multi-person pose forecasting, emphasizing the necessity of capturing interaction dependencies to achieve more precise and robust predictions in dynamic environments. Through empirical evaluation and analysis, we demonstrate the effectiveness of state-of-the-art models when pre-trained on public datasets or when fine-tuned on custom dataset HBS (Handball Shot). The analysis presented in this paper aims to assess the practical applicability of pose forecasting models in real-world scenarios, specifically demonstrated on a custom HBS dataset.
In short, our contributions are:

1. evaluation of pre-trained state-of-the-art models for multi-person pose forecasting on a custom HBS test dataset to evaluate their practical applicability in real-world scenarios
2. analyzing the performance of these models on the HBS test dataset after fine-tuning them on the HBS training dataset, thereby examining their adaptability to new domains
3. introducing a novel custom dataset named HBS, featuring scenarios where two players execute action handball shots, enriching the available resources for pose forecasting research

2. Related work

The task of single-person pose forecasting involves modeling temporal dependencies within a sequence of observed poses to accurately predict the subsequent poses. Early works such as [11, 5, 4] have explored various approaches for single-person pose forecasting and demonstrated the feasibility and potential of pose forecasting for understanding human motion dynamics.

In recent years, advanced deep learning architectures and attention mechanisms have been leveraged to address the complexities of multi-person pose forecasting. Models such as [12, 9, 8, 13, 6, 7] have been developed specifically to capture interaction dynamics and spatial dependencies among multiple individuals for accurate pose forecasting. The proposed models explored various architectures for pose forecasting, each tailored to address specific challenges in capturing temporal dependencies and spatial interactions within articulated sequences. Attention mechanisms and transformer-based models have gained significant attention due to their ability to model long-range dependencies efficiently. SoMoFormer [9], TBiFormer [6], MRT [12], and JRTransformer [13] are notable examples that leverage self-attention mechanisms to capture complex dependencies between poses over time. Graph Convolutional Networks (GCNs) have also proved to be a powerful architecture for multi-person pose forecasting, where the skeleton structure is represented as a graph and spatial dependencies are learned through graph convolutions. Approaches like Future Motion [11], LTD [4], and SocialTGCN [7] have demonstrated success in modeling dynamic interactions among multiple individuals by leveraging graph-based representations.

Additionally, traditional architectures such as Multilayer Perceptrons (MLPs) [8] and Long-Short Term Memory (LSTM) networks [5] have also been applied to pose forecasting tasks, demonstrating performance comparable to state-of-the-art models.

These diverse paradigms reflect the ongoing research and adaptation of neural network architectures to effectively tackle the challenges of multi-person pose forecasting tasks.

3. Problem description

The multi-person pose forecasting task aims to predict the future movements of multiple individuals within a scene based on their observed pose sequences. Each person's pose is characterized by key anatomical joints like elbows, knees, and shoulders, defining their spatial configuration. The task involves predicting the trajectories of these joints over a specified future timeframe, denoted by T time steps. To accomplish this, the model is provided with a sequence of historical poses for each individual, detailing the positions of their joints in a three-dimensional space relative to a global coordinate system. For each individual indexed as n, these historical poses form a chronological series $X^n_{1:t}$, capturing the evolution of their poses up to the present moment. The length of the input sequence t dictates the depth of historical data used for prediction. The primary challenge is to generate future pose sequence $X^n_{t+1:T}$ for each individual, where T denotes the number of future time steps that the model aims to forecast.

4. HBS dataset description

The Handball Shot (HBS) dataset is our new dataset comprising 10 training session scenes where two players execute handball shot actions in parallel. Each scene was recorded using Wear-Notch motion capture sensors attached to both players, enabling precise capture of their movements. The sensors record data at 40 frames per second (FPS), resulting in sequences of 5-second duration for each scene, yielding approximately 4000 unique poses across the dataset. To facilitate model training and evaluation, we divided the dataset into train and test sets. The train set includes data from 7 scenes with two players, while the test set comprises data from 3 scenes with two players. Notably, the HBS dataset differs from other publicly available datasets commonly used for training, such as 3D Poses in the Wild (3DPW) [10] and Archive of motion capture as surface shapes (AMASS) [3] in that handball shot actions exhibit rapid, dynamic, and complex motion, contrasting with the more conventional actions seen in existing datasets. Recently, datasets like The Extreme Pose Interaction (ExPI) [2] have been introduced to create environments with more extreme motion dynamics featuring two couples performing 16 distinct actions. The HBS dataset aligns with this trend of creating more challenging datasets for multi-person pose forecasting, focusing on capturing the intricacies of handball shot actions for advanced model evaluation and development. An example of scene sequence is shown in Figure 1

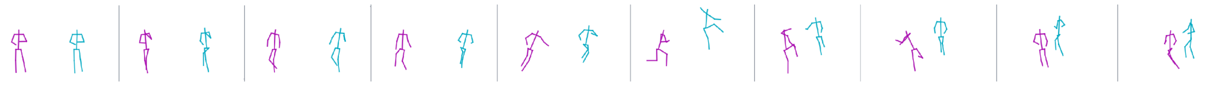


Figure 1: An example of a scene from the HBS dataset where two players parallelly execute a handball shot during a training session.

5. Experiments

In our experiments, we aim to assess the performance of pose forecasting models under different training conditions. Specifically, in the first experiment, we will evaluate the effectiveness of models pre-trained on 3DPW and AMASS datasets on our custom Handball Shot (HBS) test dataset. This experimental design allows us to gauge how well the models generalize from generic pose data to domain-specific handball shot actions. In the subsequent experiment, we will assess the performance of models on the HBS test dataset after conducting fine-tuning specifically on the training part of this dataset. This approach provides valuable insights into the transferability and adaptability of pre-trained models in dynamic and specialized scenarios.

5.1. Metrics

A commonly used metric for pose forecasting is Mean Per Joint Position Error (MPJPE). This metric calculates the average Euclidean distance between predicted joint positions and corresponding ground truth positions across all joints. A lower MPJPE value indicates a closer alignment between predicted poses and ground truth data. The MPJPE metric is calculated as follows:

$$E_{\text{MPJPE}}(\hat{y}, y, \varphi) = \frac{1}{J_\varphi} \sum_{j=1}^{J_\varphi} \left\| P^{(f)}_{\hat{y},\varphi}(j) - P^{(f)}_{y,\varphi}(j) \right\|_2 \tag{1}$$

where f denotes a time step and φ denotes the corresponding skeleton. $P^{(f)}_{\hat{y},\varphi}(j)$ is the estimated position of joint j and $P^{(f)}_{y,\varphi}(j)$ is the corresponding ground truth position. J_φ represents the number of joints. $\|\cdot\|_2$ denotes the Euclidean distance (L2 norm), and $\frac{1}{J_\varphi} \sum_{j=1}^{J_\varphi}$ represents the mean distance across all joints.

Recently, a more popular metric is the Visibility-Ignored Metric (VIM) [1]. VIM assesses the mean distance between predicted and ground truth joint positions at the last pose (T). This metric involves flattening joint positions and coordinates into a unified vector representation of dimensionality $3J$, where J denotes the number of joints. The

VIM score is computed as follows:

$$E_{\text{VIM}}(\hat{y}, y, \varphi) = \frac{1}{3J_\varphi} \sum_{j=1}^{3J_\varphi} \left\| P_{\hat{y},\varphi}^{(j)} - P_{y,\varphi}^{(j)} \right\|_2 \tag{2}$$

where J represents the number of joints, $P_{y,\varphi}^{(i)}$ is the ground-truth position of the i-th joint (flattened), $P_{\hat{y},\varphi}^{(i)}$ is the predicted position of the i-th joint (flattened), $\|\cdot\|_2$ denotes the Euclidean distance (L2 norm), and $\frac{1}{3J_\varphi} \sum_{j=1}^{3J_\varphi}$ represents the mean distance across all joints.

5.2. Training

This section outlines the training strategies and hyperparameters used for pre-training on the 3DPW and AMASS datasets and subsequent fine-tuning on the Handball Shot (HBS) train dataset. Initially, all models were trained on the 3DPW and AMASS datasets based on the methodologies and hyperparameter configurations specified in their respective papers. For the second experiment, all pre-trained models underwent fine-tuning on the HBS train dataset for 50 epochs with a fixed learning rate of 0.0001. The objective of fine-tuning was to adapt the models to the unique characteristics of handball shot actions captured in the HBS dataset.

The sequence configuration comprises 16 input poses (t) corresponding to an output of 14 poses (T), aligning with the setup of the SoMoF Benchmark [1] upon which many of the models were developed. For the 3DPW dataset, poses are sampled at a frequency of 2, translating to input poses covering 1070 milliseconds, and output poses spanning 930 milliseconds. Similarly, the HBS dataset is also sampled at a frequency of 2. Given its recording rate of 40 FPS, the input sequence from the HBS dataset represents 775 milliseconds, while the output sequence covers 675 milliseconds. This setup standardizes the sequence lengths across datasets to enable a second experiment of fine-tuning.

5.3. Results on HBS dataset

The results presented in Table 1 offer insights into the performance of multi-person pose forecasting models evaluated on the Handball Shot (HBS) test dataset, leveraging pre-training on the 3DPW and AMASS datasets. The analysis unveils notable variations in performance across different time intervals and evaluation metrics. The VIM metric emphasizes the accuracy of individual poses at certain time intervals, and models such as SoMoFormer, Future Motion, MPFSIR, and SocialTGCN demonstrate competitive performance, with Future Motion achieving slightly superior overall scores compared to others. Interestingly, while SocialTGCN performs well in long-term forecasting scenarios, it surprisingly exhibits suboptimal results in short-term forecasting tasks, while other models generally perform better in short-term forecasting. The MPJPE metric, which emphasizes the accuracy of pose sequences over individual poses, highlights the standout performance of MPFSIR. This model consistently achieves the lowest MPJPE scores across various time intervals, indicating its superior ability to accurately forecast multi-person pose sequences. Furthermore, models like Future Motion and SoMoFormer also demonstrate competitive performance, particularly excelling at shorter time intervals. However, the results suggest that the overall effectiveness of these models may vary depending on the specific forecasting requirements, with MPFSIR showcasing robust performance across different time horizons.

Table 1: Evaluation results of multi-person pose forecasting models on the Handball Shot (HBS) test dataset when pre-trained on 3DPW and AMASS datasets

Method	VIM						MPJPE					
	75ms	175ms	375ms	475ms	675ms	Overall	75ms	175ms	375ms	475ms	675ms	Overall
Zero Velocity	68.91	127.67	238.84	293.51	382.84	222.35	133.27	212.36	360.59	435.76	581.38	344.67
LTD [4]	46.11	90.26	178.25	223.02	300.20	167.57	87.30	144.15	255.39	313.85	431.48	246.44
MRT [12]	40.09	82.53	165.42	205.49	292.87	157.28	74.41	128.96	237.56	292.73	407.81	228.29
TBIformer [6]	41.81	81.85	154.54	187.42	245.39	142.20	78.22	129.46	225.61	272.18	363.06	213.70
DViTA [5]	37.85	80.51	156.37	191.77	251.66	143.63	67.42	120.32	222.83	273.44	369.93	210.79
JRTransformer [13]	29.74	72.82	152.65	189.85	262.80	141.57	50.77	104.17	212.17	264.53	368.22	199.97
SocialTGCN [7]	33.20	72.15	149.28	181.05	**232.54**	133.64	60.61	109.14	209.62	257.78	345.65	196.56
SoMoFormer [9]	**27.71**	65.76	142.36	179.64	253.39	133.77	**48.36**	95.02	193.19	243.78	346.59	185.39
Future Motion [11]	28.22	**65.20**	136.23	169.33	239.72	**127.74**	49.60	95.59	188.73	235.01	328.81	179.55
MPFSIR [8]	28.62	66.33	**135.35**	**166.32**	259.55	131.24	50.04	**94.71**	**186.88**	**231.47**	**325.19**	**177.66**

The results presented in Table 2 highlight the impact of fine-tuning pre-trained multi-person pose forecasting models on the Handball Shot (HBS) train dataset, revealing notable enhancements in accuracy across both evaluation metrics. Notably, SoMoFormer emerges as a standout performer, achieving the lowest scores across VIM and MPJPE metrics after fine-tuning. This indicates superior precision in short-term and long-term multi-person pose forecasting tasks.

Table 2: Evaluation results of multi-person pose forecasting models on the Handball Shot (HBS) test dataset after fine-tuning

Method	VIM						MPJPE					
	75ms	175ms	375ms	475ms	675ms	Overall	75ms	175ms	375ms	475ms	675ms	Overall
Zero Velocity	68.91	127.67	238.84	293.51	382.84	222.35	133.27	212.36	360.59	435.76	581.38	344.67
DViTA [5]	37.31	76.43	138.81	167.55	217.20	127.46	68.16	116.27	203.72	244.76	323.45	191.27
TBIformer [6]	37.29	72.05	126.37	148.89	185.48	114.02	70.09	114.83	194.66	229.93	293.37	180.58
SocialTGCN [7]	33.41	68.88	123.90	148.17	186.50	112.17	61.66	106.83	185.64	222.11	288.52	172.95
JRTransformer [13]	28.76	68.50	125.15	148.69	194.49	113.12	49.13	98.16	181.45	217.98	288.36	167.01
Future Motion [11]	29.52	63.61	123.63	152.41	210.23	115.88	53.61	96.93	178.56	219.05	298.66	169.36
MRT [12]	34.65	68.22	115.83	135.10	167.65	104.29	63.99	108.21	180.22	210.96	267.67	166.21
MPFSIR [8]	29.02	65.79	123.73	146.99	199.59	113.02	50.43	96.28	178.51	215.49	287.81	165.70
LTD [4]	33.29	66.19	115.46	132.53	160.87	101.67	61.89	104.01	174.28	205.02	258.65	160.77
SoMoFormer [9]	**25.85**	**58.10**	**100.79**	**120.77**	**157.76**	**92.65**	**44.91**	**85.44**	**152.55**	**182.04**	**238.45**	**140.68**

Several other models, including LTD, MRT, TBIformer, and JRTransformer, also exhibit considerable improvements in forecasting capabilities across different time intervals following fine-tuning. These enhancements highlight the effectiveness of domain-specific adaptations in optimizing model performance for dynamic handball shot actions.

Interestingly, while models like MPFSIR, SocialTGCN, Future Motion, and DViTA may experience slight performance degradation in short-term forecasting metrics after fine-tuning, they significantly improve long-term forecasting accuracy. This suggests that fine-tuning on the HBS train dataset enhances the models' ability to capture complex interaction dependencies over extended time intervals, leading to overall performance gains across diverse forecasting scenarios.

The overall improvements following fine-tuning, as detailed in Table 3, reveal varying performance boosts across different model architectures. Transformer-based models exhibit the most substantial improvement, with enhancements ranging from 19.82% to 33.69% on the VIM metric and 15.50% to 27.20% on the MPJPE metric. In contrast, models employing GCN architecture show relatively modest performance gains, ranging

Table 3: Percentage improvements in VIM and MPJPE metrics after fine-tuning multi-person pose forecasting models on the Handball Shot (HBS) train dataset, categorized by model architecture.

Method	Model type	VIM	MPJPE
Future Motion [11]	GCN	9.28%	5.67%
MPFSIR [8]	MLP	13.88%	6.73%
DViTA [5]	LSTM	11.26%	9.26%
SocialTGCN [7]	GCN & TCN	16.07%	12.01%
TBIformer [6]	Transformer	19.82%	15.50%
JRTransformer [13]	Transformer	20.10%	16.48%
SoMoFormer [9]	Transformer	30.74%	24.12%
MRT [12]	Transformer	33.69%	27.20%
LTD [4]	GCN	39.33%	34.76%

from 9.28% to 16.07% on VIM and 5.67% to 12.01% on MPJPE after fine-tuning. Notably, the LTD model, utilizing GCN architecture, experiences a significant performance increase of 39.33% on VIM and 34.76% on MPJPE, which can be attributed to its poor performance during pre-trained evaluation. MLP and LSTM architectures demonstrate comparable improvements after fine-tuning, ranging from 11.26% to 13.88% on VIM and 6.73% to 9.26% on MPJPE. These findings underscore the efficacy of domain-specific fine-tuning in optimizing model performance for dynamic handball shot actions, with Transformer-based models proving most adaptable to domain-specific datasets.

Figure 2 illustrates the predicted poses of a test example from all models, with GT representing ground-truth poses. While Transformer-based models demonstrated the best adaptation to the new domain in terms of overall performance metrics, their predicted poses often lack movement dynamic and appear static, resembling repeated poses in different locations. An exception to this trend is SoMoFormer, which produces valid and realistic movement aligned with the model's performance on the dataset. TBIFormer, JRTransformer, and MPFSIR generate even some invalid poses with minimal movement dynamic. Interestingly, models like SocialTGCN, Future Motion, and LTD exhibit more realistic movement dynamics, suggesting that GCN-based models may excel in modeling movement dynamics, while Transformer-based models excel in global position prediction, resulting in superior VIM and MPJPE scores. These observations highlight the need for a more comprehensive metric that rewards accuracy in movement dynamics regardless of global position, serving as a potential area for future research and development.

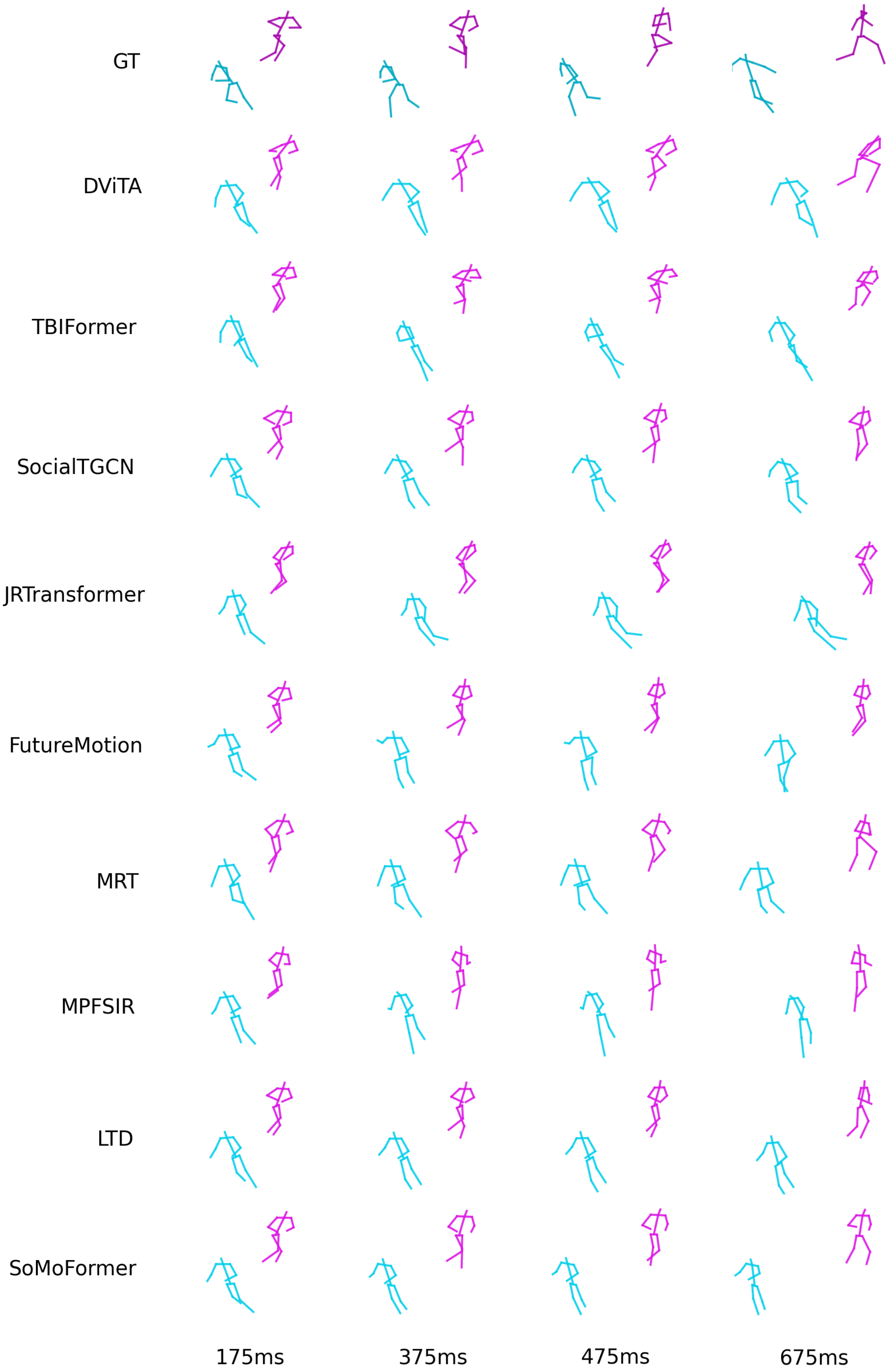


Figure 2: An example of predicted poses at different time intervals in a test scene from fine-tuned models, with GT representing ground-truth poses.

6. Conclusion

In conclusion, this study investigated the effectiveness of multi-person pose forecasting models on the Handball Shot (HBS) dataset through two experiments: evaluating pre-trained models and fine-tuning them on the HBS test dataset. The analysis revealed valuable insights into the applicability and adaptability of state-of-the-art models in real-world scenarios involving dynamic handball actions.

Firstly, the evaluation of pre-trained models showcased competitive performance across various architectures, with models like SocialTGCN, Future Motion, MPFSIR, and SoMoFormer demonstrating notable accuracy in multi-person pose forecasting on HBS. It should be noted that the MPFSIR model consistently had low MPJPE scores in this experiment.

Secondly, fine-tuning the pre-trained models on the HBS train dataset led to significant performance improvements across all models. Transformer-based models demonstrated the most significant improvements in both short-term and long-term forecasting accuracy, highlighting the efficacy of domain-specific fine-tuning.

Overall, this research underscores the importance of domain adaptation for optimizing multi-person pose forecasting models to specific action contexts like handball shots. The findings contribute to advancing the understanding of model adaptability in dynamic environments and lay a foundation for future research in the domain of action recognition and motion analysis. Further exploration could focus on refining fine-tuning strategies and exploring additional datasets to enhance model robustness and generalizability in practical applications.

References

[1] Vida Adeli et al. “Tripod: Human trajectory and pose dynamics forecasting in the wild”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 13390–13400.

[2] Wen Guo et al. “Multi-person extreme motion prediction”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 13053–13064.

[3] Naureen Mahmood et al. “AMASS: Archive of motion capture as surface shapes”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 5442–5451.

[4] Wei Mao et al. “Learning trajectory dependencies for human motion prediction”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2019, pp. 9489–9497.

[5] Behnam Parsaeifard et al. “Learning decoupled representations for human pose forecasting”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 2294–2303.

[6] Xiaogang Peng, Siyuan Mao, and Zizhao Wu. “Trajectory-aware body interaction transformer for multi-person pose forecasting”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 17121–17130.

[7] Xiaogang Peng et al. “The MI-Motion Dataset and Benchmark for 3D Multi-Person Motion Prediction”. In: *arXiv preprint arXiv:2306.13566* (2023).

[8] Romeo Šajina and Marina Ivasic-Kos. “MPFSIR: An Effective Multi-Person Pose Forecasting Model With Social Interaction Recognition”. In: *IEEE Access* 11 (2023), pp. 84822–84833. DOI: 10.1109/ACCESS.2023.3303018.

[9] Edward Vendrow et al. “SoMoFormer: Multi-Person Pose Forecasting with Transformers”. In: *arXiv preprint arXiv:2208.14023* (2022).

[10] Timo Von Marcard et al. “Recovering accurate 3d human pose in the wild using imus and a moving camera”. In: *Proceedings of the European conference on computer vision (ECCV)*. 2018, pp. 601–617.

[11] Chenxi Wang et al. “Simple baseline for single human motion forecasting”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 2260–2265.

[12] Jiashun Wang et al. “Multi-person 3D motion prediction with multi-range transformers”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 6036–6049.

[13] Qingyao Xu et al. “Joint-Relation Transformer for Multi-Person Motion Prediction”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 9816–9826.

E. Evaluacija i analiza modela dubokih neuronskih mreža za predviđanje kretanja više osoba na sceni

This work was published as: Romeo Šajina, 2024. Evaluacija i analiza modela dubokih neuronskih mreža za predviđanje kretanja više osoba na sceni. 2024 47th MIPRO ICT and Electronics Convention (MIPRO), The paper is not indexed because it was written in Croatian.

http://mipro-proceedings.com

1. Uvod

U suvremenim pristupima predviđanja kretanja više osoba na sceni, duboko učenje igra ključnu ulogu primjenjujući različite arhitekture kako bi se odgovorilo na složenost dinamičnih scena. Osim najnovijih modela koji koriste arhitekturu Transformera, primjećuje se značajna uloga graf konvolucijskih mreža (GCN) i jednostavnih mreža kao što je to višeslojni perceptron (MLP). Transformeri su pokazali uspješnost u modeliranju vremenskih zavisnosti i redoslijeda događaja. S druge strane, GCN-ovi pružaju moćan alat za modeliranje kompleksnih međusobnih zavisnosti između subjekata na sceni, uzimajući u obzir topološke odnose između njih. Integracija tih arhitektura pridonosi sveobuhvatnom pristupu predviđanja kretanja u scenama s više sudionika. Ovaj rad fokusira se na analizu i evaluaciju raznolikih modela s naglaskom na njihovu sposobnost predviđanja kretanja više osoba na sceni, uključujući pristupe temeljene na Transformer, GCN i MLP arhitekturi. Ključna karakteristika predviđanja kretanja više osoba na sceni leži i u potrebi za modeliranjem interakcija između osoba i međusobnih zavisnosti. Ovaj problem se može rješavati ručnim oblikovanjem značajki dinamike interakcije, ali također postoji potencijal da model samostalno nauči ove dinamike kroz odgovarajuće skupove podataka. Kod evaluacije modela koristiti će se metrike Mean Per Joint Position Error (MPJPE) i Visibility-Ignored Metric (VIM), kako bi se pružio dublji uvid u njihove performanse na novom MI-Motion skupu podataka. Kroz ovu analizu, rad će doprinijeti razumijevanju kompleksnosti i raznolikosti modela za predviđanje kretanja više osoba na sceni, istražujući njihovu primjenjivost i efikasnost u stvarnim scenarijima interakcije.

Prema tome, doprinose ovog rada se može sažeti na sljedeći način:

1. evaluacija najnovijih modela za predviđanje kretanja više osoba na sceni na MI-Motion skupu podataka kako bi se pružio dublji uvid u primjenjivost modela na novom skupu podataka

2. analiza najnovijih modela za predviđanje kretanja više osoba na sceni u različitim scenarijima, kao što su: park, ulica, unutarnji prostor, posebna mjesta i složene gomile.

3. analiza performansi i efikasnosti modela za predviđanje kretanja više osoba na sceni

2. Pregled literature

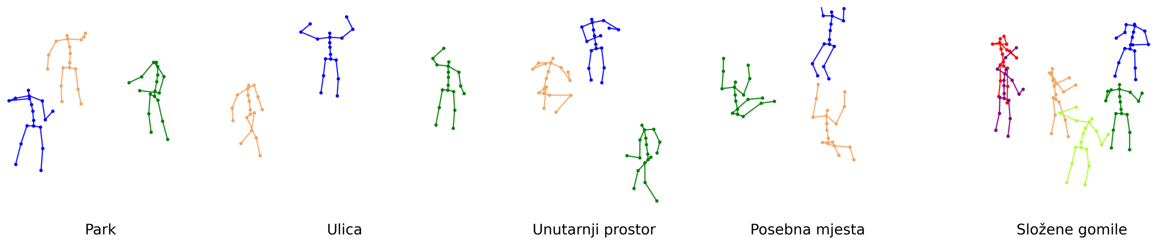
U istraživanju predviđanja kretanja više osoba razmatramo raznolike pristupe koji se koriste u relevantnim radovima. Bez obzira na raznolikost arhitektura i formulacija problema, primjetan je zajednički element - Discrete Cosine Transform (DCT), koji se često koristi za pretvaranje točaka iz koordinatnog sustava u frekvencijsku domenu, olakšavajući modelima učenje predviđanja kretanja osoba [2, 4, 7, 3, 5, 6]. Smanjenje težine zadatka postiže se i pretvaranjem sekvenci poza u pomak između slijednih poza, što se izračunava kao razlika između dviju uzastopnih poza [8]. Modeli se razlikuju u korištenim arhitekturama, obuhvaćajući konvolucijske mreže, višeslojne perceptrone, pa sve do Transformera. Graf konvolucijske mreže (GCN) se često koriste u modelima temeljenim na konvolucijskim mrežama [2, 4], dok se vremenske konvolucijske mreže (TCN) koriste za modeliranje vremenskih zavisnosti između poza u sekvenci [4]. Arhitekture Transformera postaju sveprisutne u predviđanju kretanja, pridonoseći inovacijama u blokovima i formulaciji problema [3, 6, 8]. Autori u radu [6] posebno ističu značaj modeliranja sekvenci zglobova umjesto poza, omogućavajući paralelno predviđanje cijele budućnosti za sve zglobove. U radu [8] se dodaju informacije o relacijama između zglobova u formulaciju problema kako bi se poboljšale performanse modela. Efikasnost modela naglašena je u radu [5], gdje se koriste blokovi s višeslojnim perceptronom kako bi se postigle usporedive performanse uz značajno manju veličinu modela. U kontekstu predviđanja kretanja više osoba na sceni, posebna pažnja posvećuje se modeliranju socijalnih interakcija među osobama na sceni, budući da te interakcije značajno utječu na buduća kretanja svakog pojedinca. Modeli koji uključuju ovu komponentu [4, 7, 3, 5, 6, 8] često grupno obrađuju sve osobe na sceni, omogućavajući modelu da samostalno uči o međusobnim zavisnostima između kretanja pojedinaca. Ova zajednička obrada pruža modelu potrebne informacije o socijalnim interakcijama, čime se postiže bolje razumijevanje dinamike kretanja više osoba u kompleksnom scenariju. Tablica 1 pruža sveobuhvatan pregled različitih modela korištenih u istraživanju predviđanja kretanja više osoba na sceni. Svaki model je opisan prema ključnim karakteristikama, uključujući sposobnost rada s više osoba, vrstu arhitekture koju primjenjuje, te ukupan broj parametara izražen u milijunima.

Tablica 1: Pregled karakteristika modela za predviđanje kretanja više osoba na sceni.

Model	Više osoba	Vrsta	Broj parametra (M)
MRT [7]	✓	Transformer	8,92
HRI [2]		GCN	2,83
TBIFormer [3]	✓	Transformer	8,88
SocialTGCN [4]	✓	GCN & TCN	3,37
SoMoFormer [6]	✓	Transformer	4,88
MPFSIR [5]	✓	MLP	0,36
JRTransformer [8]	✓	Transformer	3,70

3. MI-Motion skup podataka

Skup podataka MI-Motion (Multi-person Interaction Motion) [4] predstavlja obiman skup podataka s višestrukim subjektima (3-6) koji izvode različite interakcije u pet različitih scenarija svakodnevnih aktivnosti. Autori su prikupili pakete interaktivnih akcija s Unreal Engine asset store-a i snimili određene specijalizirane akcije koristeći sustav za praćenje pokreta temeljen na markerima. Umjesto nasumičnog miješanja akcija, što može rezultirati nepraktičnim podacima, autori su primijenili pažljiv pristup prilagodbe interaktivnih pokreta korištenjem ovih sekvenci akcija unutar Unreal Engine 5 game engine-a. Nadalje, poboljšali su pokrete koristeći urednik animacija kako bi stvorili prirodna interaktivna ponašanja. Ovaj metodološki pristup osigurava da generirani pokreti budu realistični i vjerojatni, pružajući precizniju reprezentaciju interaktivnih scenarija. Sintetizirani skup podataka sastoji se od 210 sekvenci s 3 do 6 subjekata, kategoriziranih prema pet različitih scenarija aktivnosti (park, ulica, unutarnji prostor, posebna mjesta i složene gomile), kako je ilustrirano na Slici 1. 3D ključne točke subjekata zabilježene su kroz ukupno 167 tisuća okvira. Kako bi osigurali pouzdanu evaluaciju, autori su uspostavili odgovarajuće podjele za učenje i testiranje.



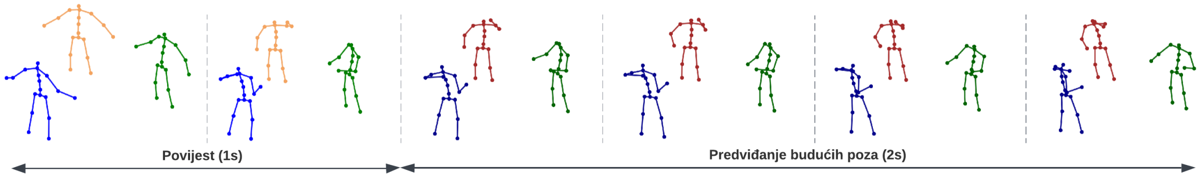
Slika 1: Prikaz primjera scena iz skupa podataka MI-Motion. Raznolikost scena uključuje prikaze iz parka, ulice, unutarnjih prostora, posebnih mjesta i složenih gomila. Svaka scena nosi svoje karakteristične značajke koje uključuju specifične uvjete, dinamiku i međusobne interakcije.

4. Evaluacija performansi modela

4.1. Piprema skupa podataka i formulacija problema

MI-Motion skup podataka je sustavno pripremljen kako bi se omogućila učinkovita evaluacija modela za predviđanje kretanja više osoba. Prvotno prikupljene sekvence, snimljene brzinom od 75 okvira u sekundi (FPS), su prilagođene za potrebe istraživanja. U procesu pripreme podataka, uzorkovanje je provedeno kako bi se smanjila frekvencija na 25 FPS, čime se osigurava učinkovitija analiza bez značajnog gubitka informacija. Od ukupno 20 dostupnih zglobova, odabrano je 18 ključnih zglobova kako bi se fokusiralo na bitne dijelove ljudskog tijela relevantne za predviđanje kretanja. Ovaj odabir zglobova omogućuje smanjenje dimenzionalnosti podataka, čime se pojednostavljuje ulazni skup za modele i olakšava učenje.

U procesu učenja i evaluacije modela, svaka ulazna sekvenca u model bit će reprezentirana s 25 okvira (1000 ms), dok će predviđanje budućeg kretanja obuhvaćati 50 okvira (2000 ms). Konačan izgled pripremljene sekvence za učenje i predviđanje je prikazan na Slici 2. Ova precizna i pomno osmišljena obrada MI-Motion skupa podataka stvara solidnu osnovu za objektivnu evaluaciju modela za predviđanje kretanja više osoba, pridonošenje dubljem razumijevanju njihove primjenjivosti i učinkovitosti u različitim scenarijima.



Slika 2: Prikaz formulacije problema predviđanja kretanja više osoba na sceni. Ulazni podaci u model obuhvaćaju 1 sekundu prethodnih kretanja osobe, dok model koristi tu informaciju kako bi predvidio buduća kretanja u razdoblju od 2 sekunde.

4.2. Evaluacijske metrike

Metrika MPJPE (Mean Per Joint Position Error) predstavlja široko prihvaćenu mjeru za evaluaciju preciznosti metoda predviđanja kretanja osoba [2, 5, 7, 6]. Ova metrika kvantificira prosječnu euklidsku udaljenost između predviđenih položaja zglobova i stvarnih položaja, obuhvaćajući sve zglobove u strukturi tijela. Niža vrijednost MPJPE označava bolje poravnanje predviđenih položaja sa stvarnim, pružajući mjeru ocjene točnosti predviđenog položaja zglobova. Metrika omogućuje analizu performansi predviđanja na razini pojedinačnih zglobova, pridonoseći dubljem razumijevanju preciznosti modela u zadatku predviđanja položaja tijela.

Formula za izračun MPJPE metrike izgleda kako slijedi:

$$E_{\text{MPJPE}}(y,\varphi) = \frac{1}{N_\varphi}\sum_{i=1}^{N_\varphi}\left\| P_{y,\varphi}^{(f)}(i) - P_{gt,\varphi}^{(f)}(i) \right\|_2 \tag{1}$$

gdje f označava vremenski korak, a φ označava odgovarajuću strukturu tijela. $P_{y,\varphi}^{(f)}(i)$ predstavlja predviđeni položaj zgloba i, dok $P_{gt,\varphi}^{(f)}(i)$ označava stvarni položaj tog zgloba. N_φ predstavlja ukupan broj zglobova u strukturi tijela.

Još jedna često korištena metrika je VIM (Visibility-Ignored Metric) [1], koja se izračunava kao srednja udaljenost između stvarnih i predviđenih položaja zglobova. Za izračun ove udaljenosti, dimenzije zglobova i koordinata spajaju se u zajednički vektor, rezultirajući pojedinačnim vektorskim prikazom i za stvarne i za predviđene položaje zglobova. Dimenzionalnost ovog vektora iznosi 3J, gdje J označava broj zglobova. Nakon što su položaji zglobova vektorizirani, izračunava se euklidska udaljenost (L2 norma) između svakog odgovarajućeg para stvarnih i predviđenih položaja zglobova. Zatim se izračunava prosječna vrijednost preko svih zglobova kako bi se dobio konačni rezultat VIM.

VIM metrika izračunava se kako slijedi:

$$E_{\mathrm{VIM}}(y, \varphi) = \frac{1}{3\mathrm{J}_{\varphi}} \sum_{i=1}^{3\mathrm{J}_{\varphi}} \left\| P_{\mathrm{gt},\varphi}^{(i)} - P_{\mathrm{y},\varphi}^{(i)} \right\|_2 \tag{2}$$

gdje J označava broj zglobova, $P_{\mathrm{gt},\varphi}^{(i)}$ predstavlja stvarni položaj i-tog zgloba (vektoriziran), $P_{\mathrm{y},\varphi}^{(i)}$ označava predviđeni položaj i-tog zgloba (vektoriziran), $\|\cdot\|_2$ označava euklidsku udaljenost (L2 norma), a $\frac{1}{3\mathrm{J}_{\varphi}} \sum i = 1^{3\mathrm{J}_{\varphi}}$ predstavlja srednju vrijednost izračunatu preko svih zglobova. Ova metrika pruža pouzdanu procjenu usklađenosti između predviđenih i stvarnih položaja zglobova, doprinoseći sveobuhvatnoj evaluaciji točnosti predviđanja položaja.

5. Rezultati evaluacije

Rezultati evaluacije, prikazani u tablicama 2 za MPJPE metriku i 3 za VIM metriku, otkrivaju značajne razlike u performansama različitih modela u zadatku predviđanja kretanja više osoba na sceni. Modeli MPFSIR i JRTransformer ističu se kao najuspješniji, dok MRT model ostvaruje najslabije rezultate. Analiza rezultata ukazuje na potrebu za postizanjem ravnoteže između kratkoročnog i dugoročnog predviđanja kretanja. Modeli poput TBIFormer, SoMoFormer i JRTransformer pokazuju bolje performanse u kratkoročnom predviđanju, no istovremeno imaju poteškoće u dugoročnim predviđanjima u usporedbi s konkurentskim modelima koji ostvaruju bolje rezultate dugoročnog predviđanja, unatoč lošijem kratkoročnom predviđanju. Iz rezultata se može uočiti da modeli s arhitekturom Transformera dominiraju u kratkoročnom predviđanju, ali su lošiji u dugoročnom predviđanju. S druge strane, modeli koji se oslanjaju na konvolucijske mreže ili višeslojne perceptrone ostvaruju bolje dugoročne rezultate, dok su im performanse u kratkoročnom predviđanju inferiornije u usporedbi s Transformer modelima. Analizom VIM metrike se može uočiti da modeli s arhitekturom Transformera uzimaju u obzir vremenske zavisnosti, gdje veća kratkoročna greška rezultira većom dugoročnom greškom. S druge strane, modeli s drugim arhitekturama pokazuju manji utjecaj kratkoročnog predviđanja na dugoročnu preciznost.

Tablica 2: Prikaz rezultata modela iskazan kroz MPJPE metriku.

Scena	Park			Ulica			Unutarnji prostor			Posebna mjesta			Složene gomile		
Vrijeme (ms)	400	1200	2000	400	1200	2000	400	1200	2000	400	1200	2000	400	1200	2000
MRT [7]	61	150	241	52	122	172	64	139	189	115	240	328	64	166	252
HRI [2]	93	131	177	65	95	119	75	116	135	138	217	262	77	130	180
TBIFormer [3]	38	104	165	36	90	132	40	98	132	92	204	268	39	112	174
SocialTGCN [4]	45	92	133	36	73	95	49	91	117	103	195	243	43	97	140
SoMoFormer [6]	37	93	144	27	75	99	36	85	111	86	184	240	30	85	118
MPFSIR [5]	32	81	125	24	**62**	**77**	32	73	97	80	**172**	**217**	27	72	101
JRTransformer [8]	**29**	**79**	**123**	**22**	**62**	86	**28**	**71**	**95**	**78**	173	226	**24**	**68**	**97**

Tablica 3: Prikaz rezultata modela iskazan kroz VIM metriku.

Scena	Park			Ulica			Unutarnji prostor			Posebna mjesta			Složene gomile		
Vrijeme (ms)	400	1200	2000	400	1200	2000	400	1200	2000	400	1200	2000	400	1200	2000
MRT [7]	56	129	210	50	91	135	56	105	136	100	177	224	57	138	198
HRI [2]	54	103	148	43	56	81	50	79	89	102	145	142	51	95	134
TBIFormer [3]	41	97	146	36	73	112	38	78	97	87	155	169	39	100	144
SocialTGCN [4]	42	80	115	33	56	72	42	64	87	88	139	145	39	80	112
SoMoFormer [6]	40	89	135	32	63	86	36	69	89	83	141	167	34	82	100
MPFSIR [5]	36	80	**113**	28	**49**	**62**	33	61	**77**	79	**124**	**135**	31	64	82
JRTransformer [8]	**34**	**78**	114	**27**	53	74	**30**	**60**	79	**78**	135	146	**29**	**62**	**79**

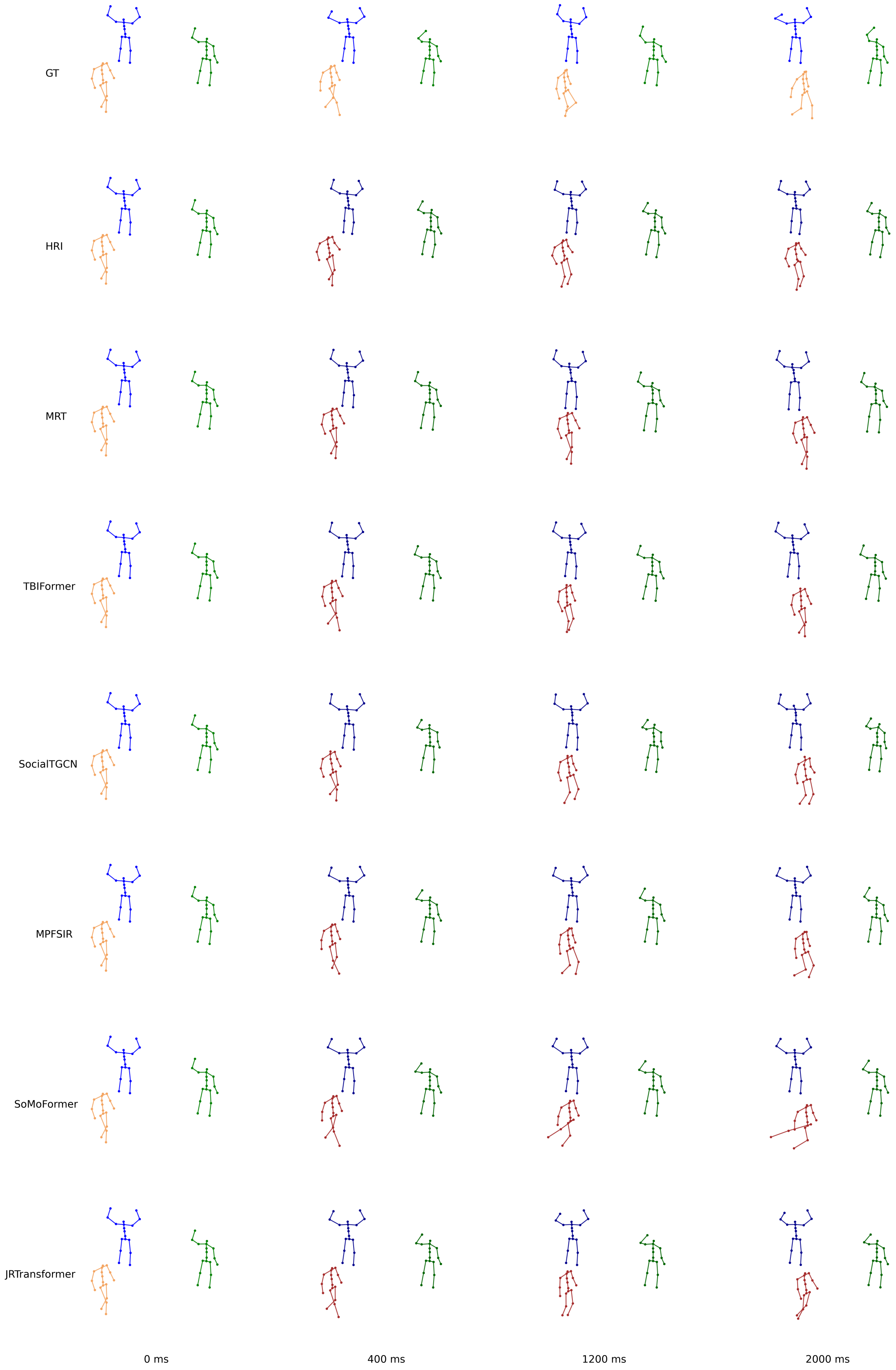
Modeli ostvaruju najmanje zadovoljavajuće rezultate na scenariju ”Posebna mjesta”, pri čemu se model MPFSIR izdvaja kao najuspješniji s iznimno dobrim rezultatima u obje evaluacijske metrike. Nasuprot tome, najbolje performanse modela bilježe se na scenariju ”Ulica”, gdje su kretanja osoba jednostavnija i uglavnom neovisna o prisutnosti drugih osoba u sceni. U scenarijima koji uključuju socijalne interakcije, modeli koji integriraju modeliranje međusobnih zavisnosti između osoba pozicioniraju se bliže vrhu rezultatske ljestvice, reflektirajući bolje razumijevanje kompleksnih socijalnih dinamika i njihov utjecaj na kretanje.

Tablica 4 prikazuje prosječne greške modela izražene kroz VIM i MPJPE metrike u različitim scenarijima. Iz tablice se može uočiti značajna varijabilnost rezultata modela s arhitekturom Transformera, naglašavajući kako različite formulacije problema i blokovi u modelu mogu izazvati bitno različite performanse. Iz analize rezultata proizlazi da modeli raznolikim strategijama uče predviđati buduće kretanje, posebno ističući se modeli HRI, SocialTGCN i MPFSIR po značajnom unaprjeđenju rezultata na VIM metrici u odnosu na MPJPE metriku. Ovi rezultati sugeriraju da navedeni modeli uspješno predviđaju poze u specifičnim vremenskim okvirima, neovisno o složenim vremenskim zavisnostima između prethodnih okvira koji vode do određene poze. Nadalje, primjećuje se da modeli mogu ispraviti eventualne pogreške u inicijalnom kratkoročnom predviđanju, postižući preciznije dugoročno predviđanje, što je posebno uočljivo u rezultatima VIM metrike. MPJPE metrika, s druge strane, striktnije kažnjava pogreške u kratkoročnom predviđanju čak i prilikom evaluacije dugoročnog kretanja. Unatoč relativno malim skupom podataka za učenje modela, najmanji model MPFSIR, ostvaruje rezultate usporedive s najboljim modelom JRTransformer, što dovodi u pitanje potrebu za dodatnom složenošću Transformer modela u kontekstu predviđanja kretanja osobe. Važno je napomenuti da ograničenje veličine skupa podataka može utjecati na performanse Transformer modela, koji inače zahtijeva obilje podataka za postizanje optimalnih rezultata.

Tablica 4: Prikaz rezultata modela iskazan kroz MPJPE i VIM metrike s prosječnim greškama kroz sve scenarije.

Metrika	MPJPE						VIM					
Vrijeme (ms)	400	800	1200	1600	2000	Prosjek	400	800	1200	1600	2000	Prosjek
MRT [7]	71	123	163	201	236	159	64	99	128	156	181	126
HRI [2]	90	117	138	156	175	135	60	81	96	111	119	93
TBIFormer [3]	49	90	122	149	174	117	48	79	101	120	134	96
SocialTGCN [4]	55	88	110	129	146	105	49	70	84	99	106	81
SoMoFormer [6]	43	80	104	125	142	99	45	74	89	105	115	86
MPFSIR [5]	39	71	92	**109**	**123**	87	41	65	**76**	**86**	**94**	**72**
JRTransformer [8]	**36**	**68**	**91**	110	125	**86**	**40**	**63**	78	91	98	74

Na slici 3 prikazana je jedna sekvenca kretanja s ulaznim podacima od 1 sekunde te predviđanjima sljedećih 2 sekunde za svaki model. Ova vizualna reprezentacija omogućuje usporedbu između predviđanja modela i stvarnih kretanja, pružajući uvid u točnost i preciznost svakog modela u predviđanju budućih kretanja osoba na sceni.



Slika 3: Prikaz jedne sekvence kretanja s ulaznim podacima od 1 sekunde i predviđanjima sljedećih 2 sekunde za svaki model. Na vrhu su prikazana stvarna kretanja kao referentni podaci (GT - ground truth).

6. Zaključak

Ovaj rad proučava i uspoređuje performanse različitih modela za predviđanje kretanja više osoba na sceni. Kroz evaluaciju temeljenu na metrikama MPJPE i VIM, analizirane su prednosti i nedostaci modela s različitim arhitekturama, pridonoseći boljem razumijevanju njihove učinkovitosti u raznolikim scenarijima. Evaluirani modeli se mogu grupirati u dvije osnovne kategorije: modeli s arhitekturom Transformera i modeli koji ne koriste arhitekturu Transformera. Modeli s arhitekturom Transformera pokazali su se učinkovitima u kratkoročnim predviđanjima, ali su istovremeno lošiji u dugoročnom predviđanju, nasuprot modelima koji ne koriste arhitekturu Transformera, a koji su pokazali suprotnu dinamiku. Rezultati sugeriraju da modeli s arhitekturom Transformera bolje usvajaju modeliranje vremenskih zavisnosti unutar sekvenci, pri čemu točnost kratkoročne greške značajnije utječe na točnost dugoročne greške u usporedbi s modelima bez arhitekture Transformera. Kao zajednički pobjednici istaknuli su se modeli MPFSIR i JRTransformer, koji su pokazali slične rezultate unatoč pripadnosti različitim skupinama modela. Ovo ukazuje na relevantnost daljnjeg istraživanja obiju skupina modela kako bi se još bolje razumjele njihove prednosti i ograničenja, te unaprijedila raznolikost pristupa u predviđanju kretanja više osoba na sceni.

Osim toga, dobiveni rezultati postavljaju pitanje opravdanosti dodatne složenosti modela arhitekture Transformera u kontekstu predviđanja kretanja osoba, s obzirom na činjenicu da model MPFSIR postiže iste rezultate koristeći znatno manje parametara i efikasniju arhitekturu. Ovo istraživanje dodatno ističe izazove povezane s primjenom Transformer modela na manjim skupovima podataka, gdje takvi modeli ne mogu potpuno iskazati svoj puni potencijal. Za sveobuhvatnu usporedbu modela, nužno je provesti evaluaciju na velikom skupu podataka kako bi se istražilo mogu li modeli sa arhitekturom Transformera značajnije nadmašiti performanse drugih modela.

Literatura

[1] Vida Adeli et al. “Tripod: Human trajectory and pose dynamics forecasting in the wild”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 13390–13400.

[2] Wei Mao, Miaomiao Liu, and Mathieu Salzmann. “History repeats itself: Human motion prediction via motion attention”. In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*. Springer. 2020, pp. 474–489.

[3] Xiaogang Peng, Siyuan Mao, and Zizhao Wu. “Trajectory-aware body interaction transformer for multi-person pose forecasting”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 17121–17130.

[4] Xiaogang Peng et al. “The MI-Motion Dataset and Benchmark for 3D Multi-Person Motion Prediction”. In: *arXiv preprint arXiv:2306.13566* (2023).

[5] Romeo Šajina and Marina Ivasic-Kos. “MPFSIR: An Effective Multi-Person Pose Forecasting Model With Social Interaction Recognition”. In: *IEEE Access* 11 (2023), pp. 84822–84833. DOI: 10.1109/ACCESS.2023.3303018.

[6] Edward Vendrow et al. “SoMoFormer: Multi-Person Pose Forecasting with Transformers”. In: *arXiv preprint arXiv:2208.14023* (2022).

[7] Jiashun Wang et al. “Multi-person 3D motion prediction with multi-range transformers”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 6036–6049.

[8] Qingyao Xu et al. “Joint-Relation Transformer for Multi-Person Motion Prediction”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 9816–9826.