PhD Thesis

Markerless Human Motion Estimation and Evaluation from Single Video

Author:
Leila Malekian

Supervisors:
Dr. Rudy Lapeer
Dr. Michal Mackiewicz

University of East Anglia
School of Computing Sciences

November 26, 2024

Declaration of Authorship

I, Leila MALEKIAN, declare that this thesis titled, "Markerless Human Motion Estimation and Evaluation from Single Video" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

Contents

List of Figures

List of Tables

Abstract

Automatic analysis and interpretation of human motion is one of the essential challenges in the field of computer vision and has been the focus of research efforts for many decades. The large quantity of research in this field is motivated by its potential applications in a wide variety of areas and make it interesting across many disciplines and research communities.

One of the main parameters of human motion research projects is the type of input data that is used for processing. Different input modalities include single-view (monocular) video cameras, Kinect depth sensors, RGB multi-camera systems and optical motion capture. The single view video camera is the most commonly used due to it being more affordable and non-specialist and its data being abundant online as compared to the other modalities. However, motions that are occluded from the single camera view, for example, due to self-occlusion, are difficult to recover. In addition to these sources of inaccuracy, the choice of the 3D human model is also important in the captured motion quality. Therefore, we adopted an adaptive shape modeling method called Skinned Multi-Person Linear Model (SMPL) which can make both joint rotations and positions available.

In my PhD research, I propose a machine learning-based method that is used in post-processing to reconstruct the incorrect motions that are caused by self-occlusion. The post-processing network is trained on a data set acquired from different subjects doing 30 different basic exercise motions that include self-occlusion. The collected data comprise single video camera footage and optical motion capture data as the ground truth. To correctly reconstruct the occluded motion, action recognition information is used to select a machine learning model that is trained on the specific motion. The performance of predictive and non-predictive networks are compared to each other and also with the state of the art in human motion estimation. The results show reduction of the overall pose error and the pose error for selected body parts with a large degree of self-occlusion.

Additionally, I have also investigated human motion evaluation which is concerned with how well a specific motion is performed. I have used both classic machine learning with feature extraction and deep learning for this purpose. An improved processing pipeline, feature selection and new machine learning models are used to improve the accuracy of the human motion evaluation compared to the state of the art and baseline methods that are using the same motion evaluation data set.

Chapter 1

Introduction

Automatic analysis and interpretation of human motion is one of the essential challenges in the field of computer vision and has been the focus of research efforts for many decades since the 1970s. The large quantity of research in this field is motivated by its potential applications in a wide variety of areas and make it interesting across many disciplines and research communities. In this chapter, a simple introduction to the project's domain and the research motivation and aims is provided.

One of the main parameters of human motion research projects is the type of input data that is used for processing. Different input modalities include: a single-view (monocular) video camera, Kinect depth sensors, RGB multi-camera system and optical motion capture. Of all these motion capture modalities, the single view video camera is the most commonly used due to it being more affordable and non-specialist and its data being abundant online as compared to the other modalities.

The extracted motion signal from any input modality can be considered in different abstraction levels such as activity, action or gestures. This motion signal is then analyzed and evaluated by assigning a number to it as a score or quality metric of that particular motion.

In the field of biomechanics, single and multiple video cameras are also called markerless motion capture methods as opposed to the traditional optical marker based motion capture methods. They are more cost-effective, offer more data versatility and provide the opportunity for the data to be used with state-of-the-art algorithms. Moreover, they don't need specific clothing and additional attire, such as markers, and this has a positive effect on the quantity of potential data that can be acquired. Markerless human motion capture is useful in the area of health science, injury prevention, and rehabilitation in sports, clinical and rehabilitation applications. Despite these advantages, the accuracy of markerless methods especially in the single-video setting is significantly lower than gold standard optical motion capture methods. Many of the mentioned applications have high accuracy requirements, so it is important to identify gaps in our knowledge that are relevant to future developments in this area and develop algorithms that increase the accuracy. Selecting an accurate model representation of the human body and correct measurement of the accuracy is also important.

1.1 Background

This research is focused on capturing and interpreting the 3D human motion using a single video camera. As mentioned before, the quality of captured human motion using video motion capture methods is usually significantly lower than optical motion capture. The areas of potential inaccuracies are diverse and they can be revealed especially when the input human motion becomes more complex. This is more evident in sport applications where the human subject can perform diverse motions with high speed or unusual poses.

As mentioned before, the advantage of having a single video camera as input is its wide availability. This means diverse motion videos are available to download from the internet and can be tested on different human pose estimation methods and algorithms. Although these diverse data are widely available for testing, this is not the case for the available training datasets of human pose estimation. The training datasets provide the video as well as corresponding ground truth data of accurate human joints positions and orientations captured by other devices such as IMUs or Optical Motion Capture. Capturing a dataset with 3D ground truth is especially restricted to a specific environment or setting and eventually doesn't allow the diversity that is available in the case of online videos. This means there is a demand for new datasets with 3D ground truth and more diverse or application-specific motions.

Choosing sport applications as a potential challenging area with complex motions, we first focused on cases where human motion is challenging for example handstand, or other different gymnastics motions or yoga poses. We tested several state-of-the-art methods on gymnastics videos downloaded from Youtube and other freely available online resources. These preliminary experiments show the main complexities of human motion estimation in monocular videos. Problems such as inaccurate capture of the occluded body parts (self-occlusion), unnatural poses such as handstand and motion blur due to speed of the execution were among the most evident issues.

For overcoming mentioned problems, the preliminary research on various motion videos showed that using extra knowledge about the motion such as 3D key points and the actions that is taken, can help to improve the predicted motion. Therefore, the main research focus in this part is that further processing the motion can result in a better human 3D model prediction in the self-occlusion and unnatural poses scenarios. The result is presented as the position and rotation error of the estimated SMPL human model compared to the ground truth data computed from motion capture.

The wide problems of the video based 3D human pose estimation methods with challenging sport motions (e.g. gymnastics) prompted us to seek out similar yet less complex motions such as those observed in martial arts. For resolving the problem of self-occlusion we have created a 3D dataset with special emphasis on self-occlusion.

In terms of human motion evaluation, supervised learning which is using annotated datasets is used as the preferred method. The purpose of human motion evaluation is assigning a score number to each motion representing how well an specific motion is performed. In supervised learning methods, the annotated scored videos from the specific motions of interest should be available. Use of small available datasets is possible when using classic machine learning while deep learning methods usually require larger amount of data. In reality, having large dataset of expert annotated 3D motions might not be possible, so improving the methods that can utilize the existing limited data is valuable.

The error measurement in the human motion evaluation is the RMSE error and correlation between the annotation and the predicted values. From the previous work on the martial art motions, it can be seen that the use of different types of more complicated machine learning models and combining the different types of features together did not work better for reducing the error. Therefore, the research focus in this part is using simpler, non-combined features along with applying better machine learning methods that can result in a better prediction of scores with high accuracy. In the second part of human motion evaluation, the motions of the large dataset of SMPL models are automatically annotated and the motion without any feature extraction is evaluated with both deep learning and classical methods to compare their performances.

1.2 Main Objective

The **main objective of this dissertation** is to estimate and evaluate complex human motions from a single-view, monocular video camera with the highest possible accuracy. These motions will not include handheld aids and particular attention will be given to occluded body parts. The two main challenges to meet this objective are accurate human motion estimation and evaluation. There are two main parts in this research i.e. human motion estimation and human motion analysis and the main aim in both parts is to reduce the error between the ground truth annotations and the prediction compared to the current state-of-the-art.

1.2.1 Human Motion Estimation

Human motion can be represented as a sequence of data that can be found by putting multiple consecutive estimated human pose from video frames next to each other. Therefore, the problem of human motion estimation (several frames) and pose estimation (single frame) are closely related to each other.

In order to improve the accuracy in estimated human motion we have focused on reducing the error human motion estimation in challenging scenarios such as self-occlusion and unnatural poses. We have also chosen the more accurate human model (SMPL) which has both joint position and orientation information.

1.2.2 Human Motion Evaluation

Human motion evaluation is analyzing the human motion data with the purpose of finding out how well an specific motion is performed. Therefore, human motion evaluation is a subgroup of human motion analysis methods.

In order to improve the accuracy of the human motion evaluation, we have focused on an specific martial arts dataset and tried to improve the prediction error of human motion evaluation method. In order to consolidate this research with the previous part which is estimating SMPL human models, we have also evaluated the motions of a SMPL dataset of diverse motions.

1.3 Research Contributions

1.3.1 Human Motion Estimation

The main contributions in terms of increasing the accuracy of human motion estimation is as follows:

- For self-occluded motion recovery, a combination of action-specific modeling, predictive modeling, and inverse kinematics (IK) enables a tailored and effective approach. By leveraging prior knowledge of the specific action being performed, action-specific models are selected to aid in reconstructing occluded body parts with high accuracy. Predictive modeling further refines this process by using past motion data to forecast future movements, which complements action-specific modeling. When unnatural poses result in incorrect limb predictions, inverse kinematics techniques utilize predicted 3D key point positions to adjust limbs to more accurate, natural positions. Together, these techniques provide a comprehensive solution for robust motion reconstruction, even in challenging scenarios with self-occlusion and atypical poses.
- New rotation-based error metric: Adding the rotation error provides a more suitable evaluation of 3D human pose accuracy.
- A new dataset for 3D human pose estimation with SMPL based ground truth. The dataset contain a set of exercise actions performed by different subjects, with special emphasis on self-occluded motions.

1.3.2 Human Motion Evaluation

The main contributions in terms of increasing the accuracy of human motion estimation is as follows:

- Motion evaluation using classic machine learning methods with handcrafted features: This approach combines handcrafted features with traditional machine learning techniques, such as random forests and regression, tailored to small, annotated datasets. By carefully selecting features and classifier types, this method enables effective human motion evaluation even with limited data.

- Demonstrating potential motion evaluation of SMPL human models using deep learning and automatic labeling of a large SMPL dataset.

1.3.3 Publications

Malekian, Leila, Rudy Lapeer. ″Self-Occluded Human Pose Recovery in Monocular Video Motion Capture.″ 2024 14th International Conference on Pattern Recognition Systems (ICPRS). IEEE, 2024.

1.4 Thesis Structure

1.4.1 Chapter 2

This chapter provides a general overview of two main research areas covered, namely human motion estimation (commonly known as human pose estimation) and human motion evaluation. The former is about inferring accurate (2D or 3D) human motion data from the input, which is mainly a monocular video in this project. The latter does an assessment of the inferred 3D motion sequence and assign a number related to how well the motion is performed. This number is refereed to as the evaluation score.

Two main categories of human pose estimation techniques are introduced and the general method of extracting human pose data including possible features is listed. Since solving the human pose estimation problem also involves various other extra sub-tasks, some of the most important ones are explained. In human motion estimation, different types of human models that can be inferred using pose estimation methods are introduced.

Regarding human motion evaluation, the techniques are categorized based on the type of input. The input usually is the extracted human model (e.g. kinematic skeleton, adaptive shape model) or the input RGB data itself. In this thesis, we have extracted the 3D human model from the video and performed the motion analysis using the joints data of model. The machine learning techniques can be also different in terms of using classic methods with formulized handcrafted features or they can use deep learning methods. A list of some previous works in the area of motion evaluation in different applications is also explained in this chapter.

1.4.2 Chapter 3

This chapter explains the methods that are introduced to improve human motion estimation. After human pose estimation, a post-processing step consisting of machine learning or Inverse Kinematics method are used to resolve the problems such as self-occlusion and unnatural poses. The results are compared to the baseline and the state of the art methods to show the improvement.

1.4.3 Chapter 4

In the second part two main human motion evaluation projects using classic machine learning and deep learning methods are explained. The former is using a small dataset of martial arts gestures and 3D kinematic skeleton data and the latter is using a larger dataset of 3D SMPL human models that have body shape and pose parameters. The results are compared to the existing work on the same martial arts dataset to show the improvement.

1.4.4 Chapter 5

This chapter discuss the research research summary and findings. It also reviews the limitations of the project and propose ideas for future research.

Chapter 2

Related work

2.1 Introduction

This chapter reviews the previous work regarding the 3D human motion estimation and evaluation. Human motion first should be estimated from an input modality (in our case video data) with use of a 3D human model and then the motion of the resulting model can be processed. Main HPE methods are discussed along with the image features that are mostly used. Different ways of modeling the human body are discussed. In previous work on human motion estimation, topics such as data calibration, foreground segmentation, human detection and tracking and body part parsing are mentioned.

In human motion evaluation, classic machine learning methods and related areas such as human motion features, motion feature processing methods and machine learning methods as well as deep learning based methods are discussed. Preprocessing methods for human motion data is also mentioned. In terms of evaluation of motion, previous research in applications such as physical rehabilitation, skill training and sport activity scoring are reviewed.

2.2 Human Motion Estimation

A typical task in computer vision is identifying a specific object in the image and finding its position and orientation relative to some coordinate system. The pose of an object is defined as a combination of position and orientation of that object. Articulated body pose estimation is finding the pose of an articulated body that consists of joints and rigid parts from input image observation.

There are many applications of technology that can benefit from pose estimation which makes it one of the key problems in computer vision. Some of them are:

- Human-Computer Interaction: human gesture can be used as an input to control a computer interface (e.g. in sign language recognition)

- Human-Robot Interaction: the perception of the human body poses by a robot can build easier and more intuitive communication with robots (e.g. in home robotics or rehabilitation and assisted living)

- Video Surveillance: activity and behaviour of humans in an outdoor or indoor environment are monitored for safety, security or managing or directing people
- Gaming: introduction of the Microsoft Kinect sensor popularized the use of depth sensors for full body control in video games and virtual reality (VR) environment
- Sport Performance Analysis: analysing the athlete's actions which require accurate pose estimation
- Scene understanding: obtaining semantic knowledge of a given scene image containing humans

Traditionally, marker-based systems are used for accurate pose estimation. However, they are not suitable for real-life non-invasive applications. Vision based systems are a cheaper and more practical alternative that uses input from cameras. The input images can be an RGB or grayscale image, depth image or infrared image. In infrared imaging devices, the sensor is sensitive to infrared light which make it useful for night vision. Depth images contain information about the distance of the object from the camera. Depth imaging devices are not expensive and the commercial products like Microsoft Kinect [1], [2] or Leap Motion [3] can be easily used in ordinary projects settings, but the distance of the object from the depth sensor is limited (around eight meters) and they can be only used in indoor environments. Furthermore, most of the standard datasets available online are RGB or grayscale images.

Despite longstanding research in human pose estimation, various problems remain due to challenges such as variability in visual appearance and physique, variability in lightning, partial visibility and self or other types of occlusions, human skeletal structure complexity, high dimensionality of pose and problems with pose estimation in 3D space including information loss from 2D images. The human body has a high degree of freedom leading to a high dimensional solution space. Usually, the solution is also required to be able to successfully deal with illumination changes, shading problems and viewpoint variations.

Pose estimation algorithms are expected to work under time, memory and processing power constraints. Because pose estimation is usually at the beginning of a longer pipeline of algorithms (for example action recognition or human computer interaction), its accuracy and time efficiency are important. If it is required to implement pose estimation on embedded systems and mobile devices, also resource constraints should be considered.

2.2.1 Previous work

Pose estimation techniques can be categorized based on their body structure interpretation method into a model based and model free approaches (Figure 2.1). The estimation process is minimizing the error between observations and a human body model (model based), a projection function (learning based) or an example set (example based).

2.2.1.1 Human Pose Estimation Methods

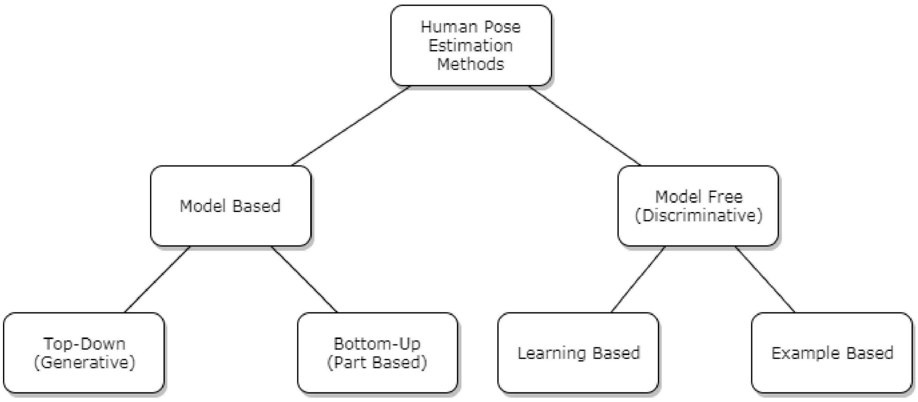


FIGURE 2.1: Human pose estimation methods

2.2.1.1.1 Model based approaches In a model based **top-down** pose estimation method, a priori information from motion or kinematic properties of a human body is employed to generate a parametric shape model [4]–[8]. For example, in [9], [10] the model is based on a priori information on specific motion and context, respectively. Top-down methods match the model with the observation. It is done by performing a local search around initial pose estimates. Due to the high dimensionality of the pose space, instead of computationally expensive brute-force local searches, an optimization approach is often used to find *a posteriori* pose estimates.

Top-down approaches consist of two stages: modelling and estimation [11]. In the modelling stage, a likelihood function is generated using the problem description (camera model, human body model, known constraints and image descriptors) and in the estimation stage this likelihood function and input image are used for predicting the most likely poses. The optimization procedure of this method is computationally expensive and needs to be initialized with accurate parameter values (problem descriptors). In top-down approaches, accuracy of the shape model is an important factor in the precision of pose estimation.

One of the disadvantages of the top-down approach is the requirement for manual initialization of the first frame, because the initial estimate of the algorithm is obtained from the previous frame. Another drawback is that an initial basic model based on physical properties of test subjects is made by taking neutral poses. This is a limitation of the top-down approach which makes it difficult for some scenarios in which there is no access to co-operative users. Forward rendering of the human body model and distance calculation between rendered model and image observation is also computationally expensive. Top-down methods often have a problem with the (self) occlusion scenarios and can cause inaccurate estimation when the error is propagated through the kinematic chain (for example error in estimating of the head position can cause errors in estimating other body parts lower in the kinematic chain).

Model based **bottom-up** methods (also called part-based approaches) [12]–[16], find body parts in the input image and represent a human skeleton as a set of connected parts restricted by joints. The estimated body parts are used in hypothesis generation about body configuration and some of these configurations can exhibit unrealistic kinematics of the human body. However, bottom-up methods are usually fast, and this disadvantage can be compensated by combining it with a tracking algorithm to ensure accuracy and consistency between frames.

There are two stages in bottom-up approaches: finding the body parts and assembling them into a human body. In the assembly stage, physical constraints (e.g., body part proximity) is usually used. To deal with the occlusion problem, motion constraints can be defined at this stage. Another advantage is that no manual initialization is needed, and bottom-up approaches can also be employed for initialization of top-down algorithms. There are also hybrid approaches which combine both top-down and bottom-up approaches. Body parts are usually considered as 2D templates, and this can produce false detections for limb-like structures in the image. Another drawback is the necessity to have a part detector for all the body parts.

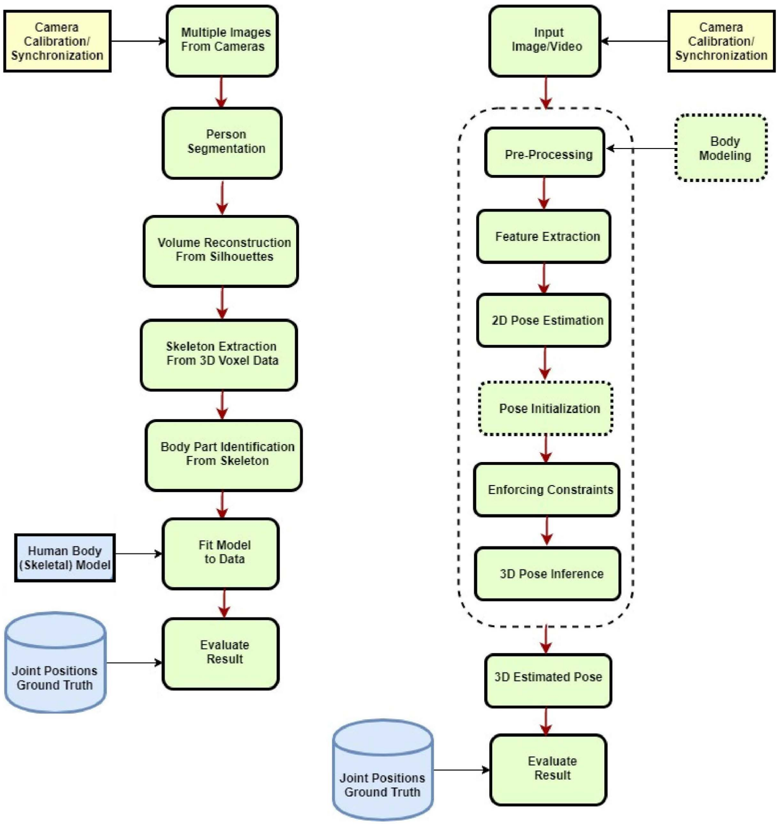

Camera Calibration/ Synchronization
Multiple Images From Cameras
Person Segmentation
Volume Reconstruction From Silhouettes
Skeleton Extraction From 3D Voxel Data
Body Part Identification From Skeleton
Human Body (Skeletal) Model
Fit Model to Data
Joint Positions Ground Truth
Evaluate Result
Input Image/Video
Camera Calibration/ Synchronization
Pre-Processing
Body Modeling
Feature Extraction
2D Pose Estimation
Pose Initialization
Enforcing Constraints
3D Pose Inference
3D Estimated Pose
Joint Positions Ground Truth
Evaluate Result

FIGURE 2.2: Example pipelines of common model-based 3D pose estimation systems (Left: bottom-up method, Right: top-down method)

2.2.1.1.2 Model free approaches Model free (or discriminative) approaches, establish a direct relation between human pose and observations without using a human body model. They can be classified into learning based and example-based methods. **Learning based** approaches use training data for learning a mapping function from the image (observation) space to pose space [17]–[19]. In **example based** methods, instead of learning this mapping, a set of example image observations and their corresponding poses are stored in a database [20]–[22]. A similarity search in the dataset for a given input observation is performed to choose some candidate poses. The final pose is estimated by interpolating the selected candidates.

As the set of feasible human body configurations is always smaller than a set of geometrically possible configurations, discriminative approaches are usually fast and robust. They also have high precision because of the ability to generalize well, and complexity in body configuration or appearance can be handled by them. In the learning-based approaches, construction of the training data is an important stage. Because of the high nonlinearity of the mapping between image space and pose space, the pose space in the training data must be densely sampled. Because of variations in body configuration, appearance and size and viewpoints, the training data need to generalize well over invariant parameters and distinguish well between variant ones.

2.2.1.2 Accuracy comparison

The advent of the HumanEva standardized dataset [23] has enabled quantitative evaluation of different human pose estimation algorithms and performance comparison with the same metric. The proposed error metric is based on a sparse set of virtual markers that corresponds to the location of joints or the endpoint of limbs. If the pose of the body is represented by M virtual markers, then the state of the body can be written as $X = \{x_1, x_2, \ldots, x_M\}$, where $x_m \in \mathbb{R}^3$ (or $x_m \in \mathbb{R}^2$ in 2D pose estimation) is the location of the marker m in the world coordinate system. The error of the estimated pose $\hat{X}$ to the ground truth pose X is expressed as the average of absolute distance between marker locations:

$$D(X, \hat{X}) = \sum_{m=1}^{M} \frac{\|x_m - \hat{x}_m\|}{M} \tag{2.1}$$

In terms of accuracy, model free (discriminative) methods tend to be more accurate than model-based methods for 3D pose tracking in monocular (single view) images. For example, in Bo et al. [24], [25], [26] most of the errors are in the range of 40-50 mm. In model free algorithms, the choice of inference form or features have less effect on the performance. However, generalizing complex motions that are not observed before is difficult. Part-based (bottom-up) methods are a general approach for human pose estimation in images, but they lag in accuracy from other methods. Bergtholdt et al. [27] suggests that they are used for initializing generative (top-down) methods. Using generative (top-down) approaches for monocular tracking is a challenge and requires very strong subject specific and motion specific priors. For example, in [28] an error of 33 mm is reported. Using physics-based models is an alternative to subject specific models [29] which is more general but have lower accuracy.

In multi-view settings, the most accurate result for pose estimation and tracking is employing a large number (>10) of cameras in controlled static environment and tight clothes (an error of 15 mm is reported in [30]). The related methods do background subtraction and recover the volumetric representation of the body. The volumetric data and the 3D body model are then fitted for tracking. Generative methods can be used for fewer camera views and weaker motion priors (32-45 mm error for a 4-camera setting in [31]), but good performance requires careful design such as choosing the likelihood function and the inference method. The most used inference technique is particle filtering. Different implementations resulted in 106.9 mm [32], 68 mm [23] and 29.9 mm [31] errors, respectively. Tracking using weak likelihood even in the multiview scenario is challenging and the methods usually reduce the search space using prior motion models [33].

A general pipeline for 3D pose estimation in images/videos is shown in Figure 2.2. A 3D pose estimation system may use an a priori body model if it is model based. Pre-processing techniques are also different among the approaches, many of them using background subtraction at this stage. The important features are then extracted from the image and used as an input for the pose estimation algorithm. Some of the methods use additional information from 2D pose estimation results or evaluate the result of projecting 3D estimated pose on 2D image. Then an initial 3D pose should be computed or obtained (e.g., Top-down methods need pose initialization). Applying the constraints will help in eliminating physically unrealistic poses. The 3D poses are then inferred, and the result will be evaluated using the motion capture ground truth available from the dataset.

2.2.1.3 Theory and formulization

Pose estimation problems are often defined probabilistically as estimating the posterior distribution $p(x|f)$, where x is the body pose and f is a set of extracted features from the image. Methodologies in pose estimation can be characterized by the different choices of these key parameters:

- x: the pose representation
- f: the feature representation/feature encoding method
- the inference method for estimating $p(x|f)$

2.2.1.3.1 Pose representation There are different ways to show the configuration of the human body. The kinematic tree is one of the common representations:
$x = \{t, \theta_t, \theta_1, \ldots, \theta_n\}$, where t is the root segment (the pelvis is usually considered

as the root segment to make the kinematic tree short), θ_t is the orientation of the root segment in the world coordinate system and $\theta_1, \ldots, \theta_n$ are other joint angles which represent the orientation of each body part with respect to its parent. Body models with different dimensionality ($2D$, $2.5D$,$3D$) can be shown with a kinematic tree representation. In $3D$ models, the θ_i dimensions depend on the joint type (i.e., Spherical joint, saddle joint or hinge joint). $2.5D$ models are extensions of $2D$ models in which we also have discrete variables (layers) representing the depth of the body parts. This representation produces a high dimensional pose vector and the alternative way is parameterizing the pose by ($2D/3D$) position of a set of the most important joints $x = \{p_1, p_2, \ldots, p_n\}$. Despite the simplicity, this representation is not invariant to body segment length and is not commonly used.

Another method of representing human body structure is modelling the body as a set of parts: $x = \{x_1, x_2, \ldots, x_M\}$, in which each part has position and orientation $x_i = \{t_i, \theta_i\}$. By defining physical and statistical constraints, body parts are connected and skeletal or image consistency is enforced. The resulting representation of part-based modelling have higher dimensions due to redundancy in parameterization, but this redundancy allows more efficient inference of the pose. Although $2D$ representation is more common in part-based models, both $2D$ and $3D$ parameterisation is possible. In $2D$ representation another scaling variable of the body part s_i is often used: $X_i = \{t_i, \theta_i, s_i\}$.

2.2.1.3.2 Image features The first step of studying human motion is the accurate feature extraction from the input which has a large effect on performance of pose estimation and any further steps. The choice of image features that represent the main points related to human pose is also important. Another significant factor is the feature encoding/feature descriptor method which is employed to describe the low–level features and reduce the size of feature space. Some methods also reduce the dimensionality of the resulting feature vector by vector quantization. For pose estimation, various features can be used for different purposes:

- **Silhouette and Contours**: silhouettes [34], [35] and contours [36], [37], [38] can be used for separating the human body from the background. The best performance in extraction is when the background is static. In other scenarios the background is considered different from human appearance. The silhouette is not affected by variation in colour, texture and lighting, but shadows or noisy background can have a negative effect on silhouette extraction performance. The silhouette also contains lots of information for 3D pose estimation [39], however, due to having no information about depth, recovering a certain degree of freedom (DOF) is very difficult.

- **Edges**: discontinuity or large changes in pixel brightness can be found by edge detection. Therefore, external or internal contours of the body are identifiable by edges. Extraction of the edges is fast and robust, and this feature is mostly insensitive to changes in lighting. Cluttered background or textured images are unsuitable for extracting edge features so the approaches that use the edges as features often extract them within a silhouette [40], [41], [42] or the projection of a human model [43].

- **Colour/Texture**: colour can be a suitable feature for modelling skin and clothing. As the body part's appearance is mostly unchanged in different human body poses, colour and texture can be used for modelling the human body. Colour histograms [44], [45] or Gaussian distribution of colours [46] are common descriptors of body part appearance.

- **Motion**: The difference between two consecutive frames when pixel brightness are assumed unchanged produce the motion data. Displacement of pixel positions which is caused by motion is called optical flow [47], [48]. Optical flow can also be combined with other features, for example for weighting the importance of edges [49].

Additional raw features include gradient information for texture modeling of body parts, along with shading and focus attributes. To enhance robustness, a combination of these features can be used within a likelihood function, where the cost functions of individual features are multiplied together. This approach results in a sharp peak in likelihood when all feature costs are low—meaning each feature produces a low cost, which signifies a strong match or high compatibility. Choosing an appropriate likelihood function is crucial, as it ensures that performance remains high even when some features may not align as expected.

To reduce the dimensionality and increase robustness to noise, the mentioned low-level features are compressed into feature descriptors. The most common feature description (feature representation) methods are:

- Scale Invariant Feature Transform (SIFT) [50], [51], [24]

- Shape Context (SC) [52], [53], [54], [39], [24]

- Appearance and Position Context (APC) [10]

- Histogram of Oriented Gradient (HOG) [55], [56], [57]

HOG has been used extensively in recent publications because of high performance in cluttered images and the ability to extract discriminative features of the image. POSEBITS [58] is a recent semantic descriptor which infers qualitative information about the 3D human pose from the image. Hierarchical and multilevel descriptors such as HMAX [59], spatial pyramids [59] and vocabulary trees [59] also can be used as a feature encoding method.

2.2.1.3.3 Inference framework Different approaches in pose estimation estimate the posterior distribution $p(x|f)$ in various ways. The most direct solution involves defining p as a parametric or non-parametric conditional distribution and using training data to learn the parameters of this distribution. This approach frames pose estimation as a probabilistic regression problem, where the goal is to learn the regression function. Consequently, this approach aligns with discriminative methods mentioned earlier, as it focuses on directly learning the mapping from features to poses. The training data (a set of labeled poses and corresponding images) can be generated using computer graphics software packages.

Parametric methods have an advantage of fixed model representation with respect to training data size, but they are not effective when there is a nonlinear relationship between poses and images. Non-parametric methods perform well in such cases, but the complexity of the model and inference is dependent on the size of the training data. Ambiguity in features is when some different poses correspond to identical features. This situation produces a multimodal distribution. Some parametric (e.g. parametric mixture models) and non-parametric solutions are introduced to deal with ambiguity problems.

In generative (top-down) methods, the posterior $p(x|f)$ is shown as a product of likelihood and a prior:

$$p(x|f) \ = p(f|x)p(x) \tag{2.2}$$

Most methods use maximum a posteriori (MAP) estimation which maximize this product and choose configurations that have high likelihood and high prior:

$$x_{MAP} = argmax \ \ p(x|f) \tag{2.3}$$

When the articulation space is high dimensional, the search for finding the mentioned configuration is very difficult and usually results in finding a local maximum. Some hierarchical search methods can solve this issue for simple skeletal configurations in an upright position and for multi-view input data. However, for single view data and more general body configurations, the success of this method is very limited.

In part based (bottom-up) methods, the size of the search space is reduced significantly because of searching each body part separately by considering only adjacent constraining body parts. Although the bottom-up method has been usually more successful than top-down methods [60] , combining top-down and bottom-up approaches can be used in situations with more complex postures and motions [61].

2.2.1.4 Human body and motion modelling

Due to the importance of the human body model in this research area, in this section we are briefly introducing 3D kinematic models, probabilistic models and adaptive shape modelling of the human body.

2.2.1.4.1 3D kinematic skeleton models One of the effective factors on human tracking and modelling is the accuracy of the 3D model of human shape and motion. The kinematic model or kinematic chain is often considered as the underlying structure in 3D computer animation applications such as games and special effect. In a kinematic model, length of each limb (body segment) and 2D/3D rotation angle between the limbs is usually known. It should be also noted that using a known position value of a visible surface point, it is possible to find the joint angle values through inverse kinematics (IK) which is widely used in the computer graphics field. Figure 2.3 shows the 2D kinematic model from Openpose computer vision library. Figure 2.4 shows 3D Kinematic model measured from MoCap data in h36m dataset. Figure 2.5, shows 3D Kinematic model from Kinect v1 and v2.

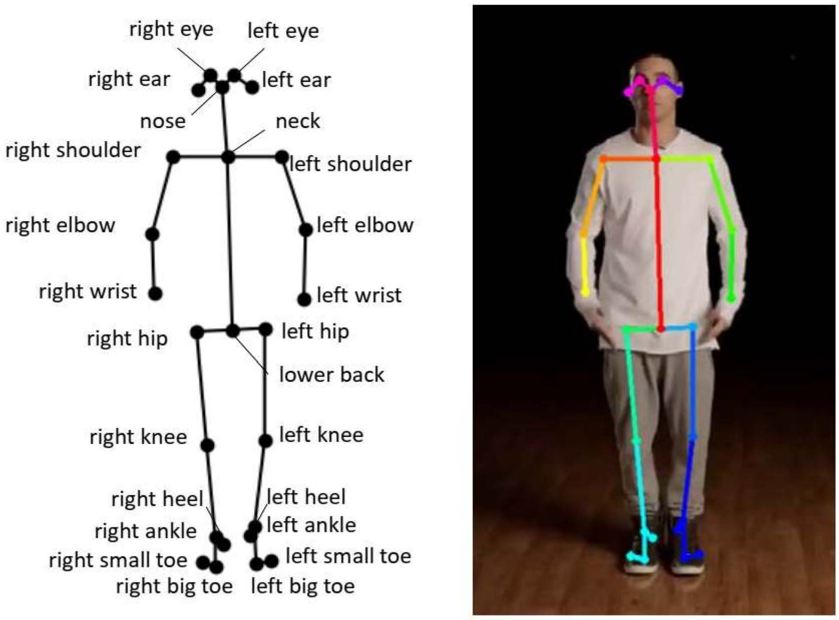


FIGURE 2.3: 2D Kinematic model from the Openpose [13] library

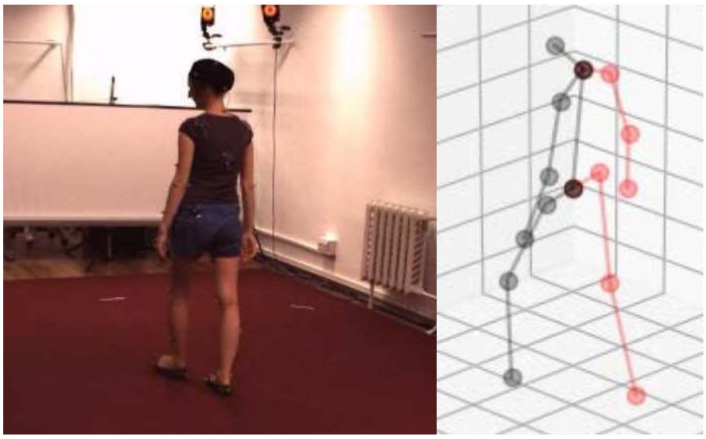


FIGURE 2.4: 3D Kinematic model measured from MoCap data in h36m dataset [62]

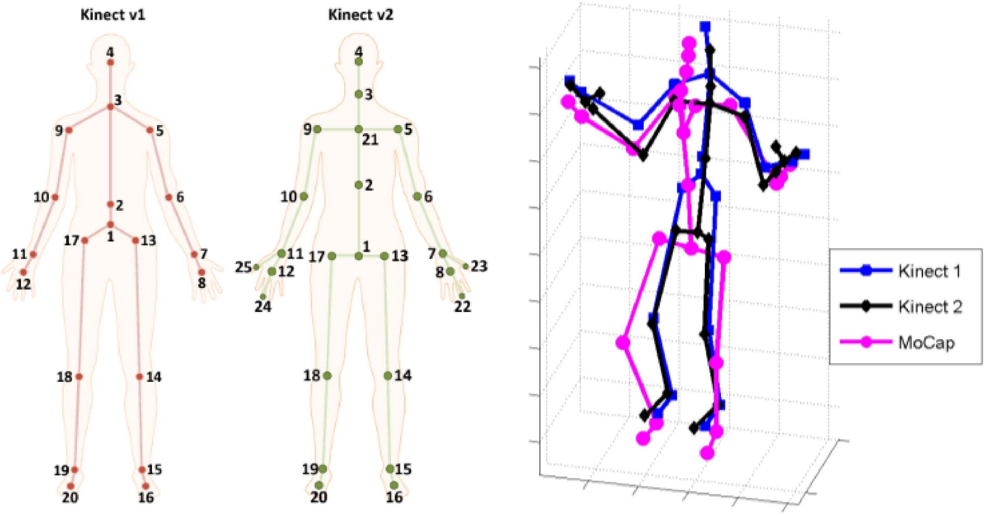


FIGURE 2.5: 3D Kinematic model from Kinect v1 and v2 [63], [64]

Kinematic chain models can be used for both full body modelling [65] ,[66], [67] and hand modelling [68]. Introduction of Kinect RGB-D camera [69], [70] was the source of significant advances in real-time reliable hand tracking and after that, RGB modelling and tracking methods has also seen a significant improvement with using neural networks techniques [71], [72], [73], [74], [75], [76], [77], [78], and some methods also used a combined body and hand tracking [79], [80], [81], [82]. In the area of full body modelling, one of the first approaches is assigning an ellipsoid or superquadratic to each limb. This model is fitted to each frame using the extracted silhouettes or matching occluded edges [83], [84], [85], [86]. Skeletal model tracking in real-time using Kinect depth cameras was one of the first breakthroughs in this field that was first used for interactive gaming [87], [88], [89]. In the current skeletal modelling and tracking some methods use 2D skeletal models and measurements and some use 3D measurement (from range data or multi-view videos) and corresponding 3D model [90] or use monocular video for direct 3D model inference [91], [92]. Temporal models are often used in periodic motions such as walking, for example, joint angles can be analysed as a function of time [93], [94], [95]. Principal component analysis (PCA) and prior knowledge can help to learn typical motion patterns in these methods and improve their generality [96], [97].

2.2.1.4.2 Adaptive shape modelling Full-body modelling of the human subject can be done by fitting a parameterized shape model to visual data. Morphable models for appearance and shape of an object can be created when a large set of registered 3D scans is available [98]. Similar to this idea, in [99] first a large number of range data scans of different people in a variety of poses are acquired and then these scans are registered using (semi-automated) marker placement. Different human shapes can be modelled as a function of human characteristic and its skeleton pose using this registered dataset. The resulting system called SCAPE (Shape Completion and Animation for PEople) can be used for recovering a full body 3D human mesh model from few captured marker data. This process is called shape completion which is finding the best fitting shape and pose parameters of the model to the measured data. The parametric shape models of the SCAPE system are made using subjects with close-fitting clothes and therefore is not compatible with loose-fitting subject clothing. This problem is considered in [100] where the body shape is fitted within the visual hull of the specific subject in various poses. In [101] an initial surface mesh model of the subject is used for fitting to the parametric shape model of the algorithm which cause better matching to the visual hull.

The previously mentioned methods used multi-view data for fitting the body model and pose estimation, whereas monocular image data for human pose and shape model fitting is used by Guan et al. [102]. This method uses an image of a person in a natural background and firstly a manual initialization determines a rough skeleton and the height of the subject and using this information the subject outline is found using the GrabCut segmentation algorithm. The extra information from the edge and shading is then used to refine the estimated shape and pose the model and the result can be used for creating animated 3D subject.

The first research efforts in the area of 3D human pose and shape fitting was done by the above-mentioned method SCAPE and also BLENDSCAPE [103]. Following these work, another method named **Skinned Multi-Person Linear** (SMPL) model [104] can model a wide range of different accurate human body shapes and natural human poses with a skinned vertex-based model. The SMPL model is built via training on a large dataset of human 3D scans and composed of a template model (for both genders) for a rest pose, pose-dependent blend shapes and identity dependent blend shapes. In [105] the parameters of the SMPL body model in a single image are found using a method called SMPLify. The mentioned SMPL model does not have underlying articulations for the hand and therefore cannot capture the hand poses in the image. In [79] a hand model is introduced that is called MANO (hand Model with Articulated and Non-rigid deformations). They further attached MANO to the SMPL model and create a fully articulated body and hand model called SMPL-H model. The SMPL body model and a face and a hand model are joint together in [80] for multi person tracking. The resulting 3D model is called the Frank and Adam model. Face and hand articulation is added to the SMPL model in [81] with a method called SMPL eXpressive (SMPL-X), because of expressive face and also gender specific models. The SMPL model is using the mixture of gaussians as a prior. This is changed in the SMPL-X model with a variational autoencoder as a prior called VPoser that is trained on the AMASS [106] motion capture dataset. All the previously mentioned SMPL estimation methods are using a single frame image as an input. To extend this to the video input another method called VIBE [107] used the temporal deep neural networks architecture and 2D and 3D human motion datasets in video and also the AMASS motion capture dataset and estimate a moving SMPL model inferred from the video input. Another method called ExPose (EXpressive POse and Shape rEgression) is introduced in [108] for direct regression of hand, face and body parameters in an input image. In the AMASS dataset [106], optical motion capture data from more than 15 different datasets are converted to realistic 3D human meshes using the MoSh (Motion and Shape Capture) method [109]. This representation replaces the previously widely used 3D skeleton data in MoCap datasets. Given a standard marker-set, MoSh estimates the position of markers on a 3D body model, estimate the body shape and articulated body pose. Recovering the moving shape of the body, MoSh is capturing non-rigid soft tissue motion with only a small number of markers producing a reasonable result.

Figure 2.6 shows the SMPL model with the underlying skeleton. Figure 2.7 and figure 2.8 shows mosh markerset for front and back of the body.

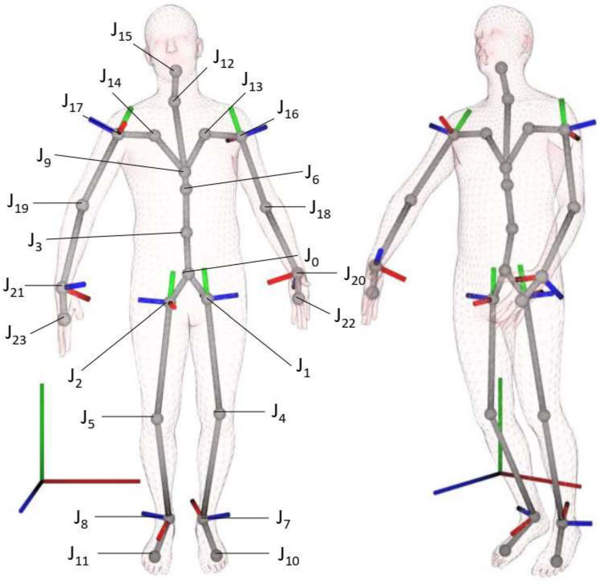


FIGURE 2.6: The SMPL model with the underlying skeleton with n=24 joints. The first 21 joints of SMPL-H and SMPL are also identical and in the same positions.

More recently another model called STAR (Sparse Trained Articulated human body Regressor) is introduced [110] with less parameters than SMPL and without long range correlation between the model vertices. Shape dependent pose-corrective blend shapes are used in STAR that can change depending on the subject's body pose and the BMI. it also used an additional 10000 scans of subjects with different genders that increase the variety of data. Body and face deformations are inferred using nonlinear shape spaces made from deep variational autoencoders in a method called GHUM and GHUML [111]. The skeleton kinematics are shown by the flow representation that is further normalized. The accuracy and speed of the methods are the focus of the recent methods and they often use the challenging 3DPW (3D Poses in the Wild) dataset [112] as a benchmark [113], [114], [82].

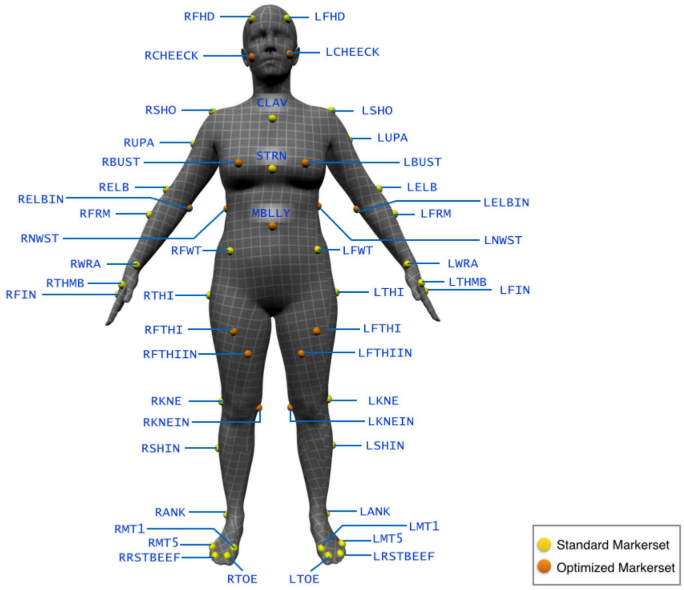


FIGURE 2.7: MoSh [109] Marker-set, 47 yellow standard Vicon marker-set and 20 orange markers added to shape improvement

2.2.1.4.3 Probabilistic models Due to difficulty of human tracking, to estimate how likely a person is in a specific state probabilistic inference methods are used. One of such methods is called particle filtering [115] that initially was used for tracking the person outline and hands, but was later applied to the problem of full body tracking [116], [96], [117] and as of today it is still used in modern trackers [118]. Another method that can be used in order to handle the tracking uncertainty is the multiple hypothesis tracking [119] and inflated covariance [120]. An example of the spatiotemporal probabilistic graphical model is loose-limbed people model in [121], that models both the geometrical relationship between limbs and their likely temporal dynamics. Using the training data, the conditional probabilities relating to different time instances and limbs are learned and the final pose is inferred using particle filtering.

2.2.1.5 Human Motion Datasets

Human tracking, shape and appearance modeling, activity recognition of humans are among the most popular areas in computer vision [122], [123], [124]. Many datasets are produced for this purpose, some important ones include the HumanEva dataset [125] with multi-view videos of human actions and its corresponding motion capture data, a tracker based on particle filtering is also provided. The Human3.6M dataset [62] with 3.6 million images from 7 professional actors with 15 daily activities (for example walking, sitting, eating, making a phone call or face to face discussion), 2D joint locations and 3D joints ground truth positions are provided as well as camera parameters and body proportions information for the actors. Humaneva [125] is a smaller dataset, but it is widely used as a benchmark in the last decade. The

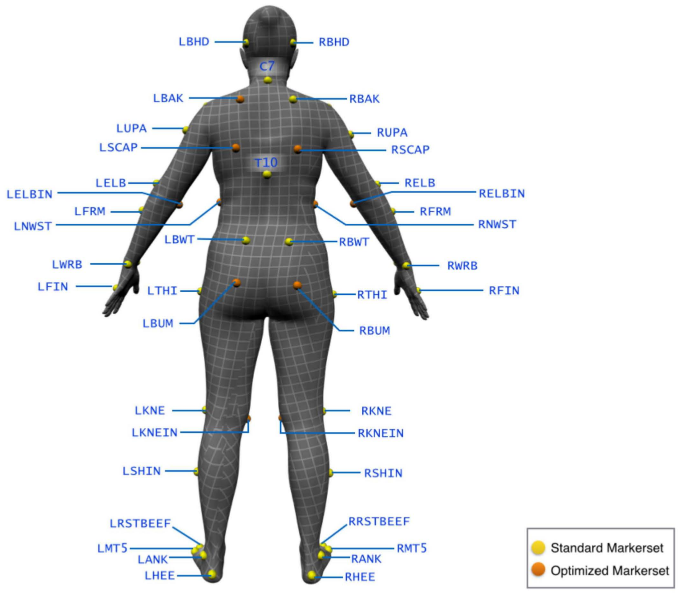


FIGURE 2.8: MoSh [109] Marker-set, 47 yellow standard Vicon marker-set and 20 orange markers added to shape improvement

MPII dataset [126] is a standard 2D pose estimation dataset containing thousands of short YouTube videos. The MPI FAUST dataset [127] with 300 human scans of high resolution and automatically computed ground truth and the new AMASS dataset [106] that has more than 40 hours of motion capture data with 300 subjects and 11000 motions are among useful 3D datasets. 3DPW (3D poses in the wild) dataset [128] is the first dataset in the wild with accurate 3D poses for evaluation, this dataset is also widely used as a benchmark for evaluation. The data in 3DPW dataset is captured using moving camera and IMUs, containing 60 video sequences with corresponding 2D and 3D pose annotations and camera poses for every frame, 3D body scans and 3D people models.

2.2.1.6 HPE Specific Research

3D human pose estimation in monocular images has been an active area of research in the last two decades. One approach to solve the problem of 3D human pose estimation in RGB images is using the training data of images and corresponding 3D poses. Some popular evaluation datasets like HumanEva [23] and Human3.6M [129] contain synchronized images from multiple cameras and the associated ground truth obtained from marker-based systems. As the resulting data is produced in a lab environment, they are not realistic in many ways. Since 2D and 3D data from one source is not practical in real world applications, an alternative approach is to use independent 2D and 3D sources of data. As annotating 2D data are done manually after the experiment, it does not require any change in the input data. Moreover, there is no limitation in the choice of input datasets using this method. The second source of data that is considered independent from the first source is accurate

3D pose data acquired from a marker-based motion capture system. From the first source, it is possible to estimate the 2D pose of the human in the image. The second source (motion capture database) is also processed by projecting 3D poses to several 2D planes of virtual cameras. The next step is retrieving the nearest 3D pose. Then, a mapping from 3D pose space to 2D image will refine the estimate of the 2D pose. The iterative repetition of this process can improve estimation of 2D and 3D poses.

Estimating 3D human pose from a 2D pose by exploiting motion capture data has been introduced recently in some previous work. Yasin et al. [130] describe a method where from 2D annotations of the images the 2D poses can be estimated and tracked in the video. The 3D poses are retrieved using the nearest neighbor search. The 3D configuration of the human body is computed from 2D anatomical key points as described in [131]. They estimated the parameters of the sparse representation of the 3D pose in an over-complete dictionary with a matching algorithm. This approach was further improved by Wang et al. [132]. They also represent the 3D pose as a linear combination of a sparse set of bases learned from the 3D human skeletons. An anthropomorphic constraint on the limb length is enforced. By minimizing the L1 norm error between detected 2D pose and the projection of the 3D pose, the correct 3D pose is estimated. To solve the optimization problem, they use the alternate direction method (ADM). In [133] a 2D part detector and a stochastic sampling that explores each part region are used. For sampling corresponding 3D poses from the pose space an evolutionary algorithm is used. This produces a set of 3D poses that all have the same image projection. By imposing some kinematic constraints that ensure the shape of the human body, the 3D pose is inferred. Simo et al. [134] extended this approach by iterating 2D and 3D pose estimation. Unlike the previously mentioned algorithms, this approach deals with the error in the 2D pose estimation step. Making use of motion priors is discussed in [135], [61], by assuming that information about the type of motion is available in advance. These priors can be learned from the motion capture data and are used for tracking the 3D pose.

A general pose estimation framework using part based methods consists of some common stages:

- **Data calibration**
- **Foreground segmentation**
- **Human detection**
- **Human tracking**
- **Body part parsing**

2.2.1.6.1 Data calibration Camera Calibration is the process of finding camera parameters that affect the imaging process. Both the extrinsic (translation and rotation) and intrinsic (focal length, skew factor, lens distortion, aspect ratio, principal point) parameters can be found in single or multiple camera settings. In [136], the performance of the actor is captured by synchronizing and calibrating multi-view sequences recorded with 12 cameras. Depth imaging devices like Kinect can also be used for calibration. In [137] the captured depth and colour data from three Kinect devices are synchronized and calibrated automatically using OPENNI [138]. To capture motion in outdoor settings, Shiratori et al. [139] used body mounted cameras to reconstruct the motion of the subject. To calibrate and estimate extrinsic parameters of the outward looking cameras attached to the limbs, they first choose the initial two images with high number of matches and estimate relative position and orientation and incrementally add the images. From the resulting correspondences they reconstruct the camera pose.

2.2.1.6.2 Foreground segmentation One of the first steps in many human tracking systems is modelling the background and extracting the moving foreground object (silhouette) that corresponds to humans in the scene. Many human pose estimation methods also need the segmented human as their input. A review of several different matting and background maintenance (modelling) methods is introduced in [140]. The silhouette of the person can be extracted by considering that the background is static. The various approaches in background subtraction (segmentation of the moving region) differ in their background model and the method of updating the model. Stauffer et al. [141] introduced a probabilistic background subtraction method of modelling each pixel as a mixture of Gaussians. In [142] a more comprehensive approach was proposed that not only models the background image statistics, but also appearance of the foreground objects for example their edge and motion statistics. To improve the segmentation error and increasing the robustness, Agarwall et al. [39] used the histogram of the edges of extracted silhouette to encode the local shape. In part-based methods, the search area of body parts can be restricted by finding the foreground. The goal is not having a perfect segmentation, but reducing the background clutter and not losing body parts. A more traditional approach is background subtraction used in [143] to extract silhouettes. In [144], the detection window is used for initializing the Grabcut segmentation algorithm[145]. The Grabcut algorithm is rather conservative, and part of the background usually remains, but unlike background subtraction, this approach doesn't need any prior information about the background and allows the change of background during the time in videos. There are also more recent methods for video background matting such as [146] and [147]. After extraction of the silhouette from one or more camera, they can be modelled using deformable templates or other contour models [148], [149]. These silhouettes can be tracked over time, and multiple people moving in the scene can be analysed in many ways, for example shape and appearance modelling or detecting a carried object [150], [151], [152]

2.2.1.6.3 Human detection Automatic tracking of people requires firstly detecting the human presence in each frame individually. The goal of human detection is determining if an image contains a human and finding the approximate location of the human body in the image. This topic is very much explored in the context of pedestrian detection, which can be considered a type of object recognition [153], [154], [155], [156], [157], [158], [159], [160], [161]. A more generic approach that can cope with a variety of poses is introduced in [54]. The marginal distribution of the position of the torso is first computed and its modes are used to predict the position of the human bounding box. For the specific application of human movies and TV shows, Eichner et al. [144] applied some methods that had excellent performance in rigid body detection for training an upper body detector. They first start with the Histogram of Oriented gradient [55] and linear SVM classification approach and then tried latent SVM [162] that improves the previous method by using a filter on HOG features. They also tried to improve the performance by combining a face detector [163] with their upper body detector. For purposes of pose estimation in videos, Sapp et al. [164] incorporated a similar initial upper body detector. They introduced a feature-based model that uses a variety of single frame features (e.g., colour and flow-based hand detector, HOG based limb detector for shoulder and elbow) and between the frame features. A multi-person detection and pose estimation by Eichner et al. [165], used full and upper body detector based on latent SVM and face detectors to increase the detection rate and decrease the false positive rate. Additional methods for initialising 3D trackers based on 2D images are introduced in [166], [167], [20], [168], [169], [170] and [171], [172]. Many of the single frame human detection and pose estimation methods are used for tracking purpose too [173], [174], [175], [176], [13], these algorithms are often combined with frame to frame techniques for more reliable result [177] [178], [179].

2.2.1.6.4 Human tracking Frame to frame tracking of human and its pose can be done with computing optical flow and comparing the appearance of the limbs across frames. In [180] appearance of upper and lower leg portions is modelled as a moving rectangle and optical flow is used for estimation of the location of these parts in each frame. In [119], [96] limbs are tracked using templates and optical flow and additional methods are employed to deal with uncertainty and multiple hypotheses. Full 3D model of the limb and body motion is used in [85] for tracking. Estimated motion fields are matched to some prototypes in [181] for identification of a specific phase in the motion (running motion) or for matching two portions of a low resolution video in the video replacement procedure. Non-rigid deforming of the objects such as the subject clothes also can be tracked using flow based methods [182], [183], [184], [185], [186], [187]. The 3D mesh model of a moving person that is evolving can also be estimated using inter-frame motion [188].

2.2.1.6.5 Body part parsing After pre-processing the image via detection and segmentation, the most important step is locating different body parts in the image. We will review different approaches regarding different features used for body part detection and the models employed for parsing the body parts.

The features that can be extracted from the input image can be handcrafted local features or learning based features. Handcrafted features, refer to features that are derived using various algorithms from the image data. They are usually used with traditional machine learning methods for various computer vision tasks (e.g., Object recognition). The most recent approaches like convolutional neural networks do not need these handcrafted features and are able to learn the features from the image data directly. Histogram of oriented gradient [189] and shape context [54] are the most widely used handcraft features. Sapp et al. [190] used a combination of appearance, geometry and shape information of parts or pairs of parts in their cascade of models which is used to prune the state space (all possible positions and orientations) of parts. The recently emerging deep learning methods, automatically learn the image representation by capturing the data structure. In recent work, deep learning methods are also used for feature extraction in human pose estimation. In [191] the location of body joints is found by a 7-layered generic convolutional Deep Neural Network (DNN). For increasing the precision, a cascade of DNN based pose predictors is also introduced. Chen et al. [192] trained a Deep Convolutional Neural Network to estimate part presence and also pairwise part relations from local image patches that centered at the body joints. In [193] human detection and pose estimation is learned jointly by a single deep model. The proposed multi-source deep model nonlinearly integrates various information sources such as appearance and deformation. Deep learning approaches are more tolerant to the variation of the dataset and bottom-up part based models make it easier to incorporate prior knowledge about the human body structure. A convolutional network part detector is combined with a part based spatial model in [194] into a single learning framework to improve the performance.

Structural models show the relationship between body parts. In a tree structure model, each body part is a node that is connected to its neighbour part (node). To extend capability of the tree structured graphical models and capturing all the interactions between body parts, Ramakrishna et al. [195] introduced an inference machine framework to learn a spatial model and use a supervised predictor that results in an improved pose estimation. A full relational body model [196] is used instead of a tree model for improving body part parsing. The relation between body segments (for example right and left limbs) is defined in the full relational model as well as the kinematic relation in the tree model. This improvement needs adopting approximate inference and learning procedures. Karlinsky et al. [197] introduced a part detector that use linking features instead of kinematic constraints. The model also learns the appearance of part linking and individual parts. The Pictorial Structure Model that is originally introduced by Fischler et al. [198] around forty years ago, describe a visual object by breaking it down to a number of primitive parts and specifying a range of spatial relations that should be satisfied. The deformable model of nodes (body parts) and the springlike connection between them enables a high variation in the appearance. Felzenszwalb et al. [143] applied the Pictorial Structure Model to human pose estimation for the first time. The pictorial structure framework is generic and different methods can describe appearance of each part and geometric relationship between the parts. It enables modelling faces and articulated bodies using the pictorial structure. The appearance models which are related to parsing each body part individually, restricts the likely regions containing a part. The success of a pictorial structure framework is highly dependent on a good appearance model. Eichner et al. [199] used a general learned appearance model that learns the relationship between location and appearance of different body parts from training data. Yang et al [57] introduced an extension to tree structure models that models co-occurrence between a mixture of parts in addition to spatial relations between part locations. (An Example of co-occurrence is a constraint of two body parts of the same limb to have the same orientation). They also showed that a mixture of non-oriented pictorial structures have a better performance than explicitly articulated parts because it can be tuned to find parts with a specific orientation. The models that are used in many part-based pose estimation methods are an extension of linear pictorial structure models. Body parts in all poses and appearances are detected using the same model. Multimodal decomposable models in [200] increase the number of modes in models that can capture different poses and appearances. A model based on the conditional random field is introduced in [201] to detect human body parts. It is a modification of the pictorial structure framework in which a binary random variable shows the presence or absence of a body part in every possible position, orientation and scale. In a similar approach by Pishchulin et al. [202] [203], the dependencies between body parts that are not connected is captured using the poselet feature representation. This improved the body part estimation results.

2.2.2 Research Gap

The aim of this section is the identification of research gaps in human motion estimation. Human skeleton and mesh estimation is an important problem that has attracted a lot of research in the previous decade. Vision-based monocular estimation of the human pose and shape from video/image and its improvement is the focus of the current work. There is a wide variety of solutions and methods introduced in the literature to solve the HPE (Human Pose Estimation) problem. More specifically, deep learning-based approaches are extensively used in this area and have shown a significant performance increase as compared to the previous methods. Despite this advancement, some unresolved issues and challenges are still open in this field. For example, problems regarding the pose ambiguity, depth and global position perception, occlusion by self or an object, collision of body parts, unnatural poses or issues with the training data set such as inadequate 3D annotation, low diversity of poses within the 3D data set and unavailability of more specialized 3D human motion data set that are going beyond daily human activities, still exists.

This section provides an overview of some of the most important existing challenges in human pose and shape estimation using monocular videos. More specifically, the occlusion problem and the related literature are reviewed. Some solutions for improving the result of human pose estimation that can address problems such as body part occlusion, unnatural poses will be suggested. For comparison with the previous work and understanding that the solutions are effective and relative to human pose estimation problems, some examples of previous work that made use of the same concepts will be mentioned. Finally, the section also will mention some of the less explored and new research directions that are more related to the current project objective.

2.2.2.1 Problem Statement

The main objective of the research is to improve 3D human inference from monocular RGB videos. The result represented by the output model of the HPE methods can be skeleton-based or model-based. The skeleton is the result of position-based estimation that infers root-relative joint positions. Joint positions only represent 3 degrees of freedom, however, in many potential applications, 6 degrees of freedom including the rotations are needed. Therefore, the information related to joint/link orientation or appearance is lacking from the skeleton-based pose which limits their practical application. To illustrate this issue, Figure 2.9 shows an example of pose ambiguity in skeleton-based methods with having only joint positions as an output. This shows, specific 3D skeleton poses and joint positions can represent many different joint angles and lack sufficient information.

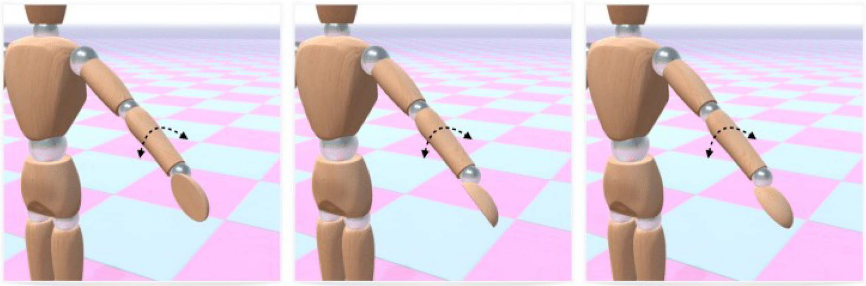


FIGURE 2.9: Left, middle and right images show different wrist rotations yet all have the same joint positions

Due to the mentioned reasons, the SMPL-based methods are chosen as the output representation. They can infer a 3D character with body shape and joint rotations that is a more accurate representation of the real 3D human body.

Focusing on SMPL-based methods, the aim is to identify and resolve some common problems that human 3D estimation faces, more specifically inaccuracies caused by occlusion, unnatural poses.

The improvement and comparison of results from model-based output with a 3D mesh can be shown qualitatively and quantitatively. For qualitative comparison, usually, the resulting 3D human model is illustrated in 2D images in the research papers to show improvement or the result can be exported to the 3D environment where more detailed information about the output can be seen. SMPL-based human models lack specific evaluation metrics that can make comparison between the current work easier. The 3D model output contains more information than only 3D key points and using the skeleton-based 3D benchmarks and their evaluation metric is not enough.

As the joint rotations are the main parameters of the pose in these models, the mean rotation error of the joints using quaternion representation is used to show overall improvement and compare the result across different methods. Comparison between joint positions is also still possible but as mentioned before, using position data results in pose ambiguity. This also relies on the estimation of a simplified (weak perspective) version of the camera model to compute 3D positions which itself is a source of error and not always available. It should be also noted that the ground truth skeleton structure of the position-based benchmark datasets is different from the underlying skeleton of the 3D model and also across different datasets, making the position error measurement imperfect. In this work, the 3D joint position errors of SMPL models will be found by exporting the models to 3D environments such as Blender.

2.2.2.2 HPE Specific Research Gaps

The deep learning methods have contributed significantly to the development of the monocular pose estimation methods. The unresolved challenges and the gap between research and practical applications still persist. Some example common problems are body part occlusion, temporal consistency, unnatural pose estimation, depth ambiguity and lack of 3D data sets. There is a history of related research in each area that is trying to address these general problems. Some of these problems such as occlusion can also occur when using other types of inputs such as depth sensors or motion capture or can occur due to a variety of reasons such as the presence of other objects, crowded scenes, or being outside the video frame. Here the focus is on the self-occlusion of body parts in a full-body RGB video input. It should be noted that the state-of-the-art HPE methods also progressed over time and the latest research shows fewer errors and improvements in these areas.

Because of fast progression of the HPE methods through time, some of the latest methods on the lab videos (containing data about exercise and body part occlusion) and in-the-wild videos (containing data of challenging poses and movements) are tested on the state-of-the-art to better understand areas that can be improved. From our observation, some of the main sources of inaccuracies in the model-based estimations are body part occlusion, temporal inconsistency, extreme and unnatural poses, and wrong estimation during the collision of body parts together. There is also the difference between the result of the skeleton-based and more difficult model-based predictions that require 3D mesh recovery.

In terms of current research trends, one of the future directions in this research area is improving benchmarks, protocols, and toolkits for 3D mesh recovery. Despite being a promising direction, there is a lack of large-scale 3D mesh data sets online and protocols for effective evaluation of such methods are still not available. These problems are addressed by recording and producing a custom 3D mesh data set in the lab consisting of synchronized video and 3D human mesh sequence. A new evaluation metric for the model-based output that is focused on the correct pose prediction will be developed and will be covered in Chapter 3.

More realistic human representations are also among the potential future directions. Few work are available that combine the models of all body parts and there is also a lack of paired and annotated data sets that merge all this information in a unified human body within the data set.

The real 3D world is a dynamic space where objects interact with each other. Building a system that is interaction-aware is among the interesting directions.

Some general approaches for improving the current HPE methods are context modeling, better and more effective training (efficient network or adequate training data), and post-processing techniques. In this work, the planned improvements will be done using post-processing techniques that can improve the resulting 3D human model output of the state-of-the-art HPE method by resolving the common issues of these systems.

As the name suggests, post-processing deals with previously processed but imperfect results. It can also get some information from the input data or do additional processing of input data to enhance the result. Although there is the ranking of HPE methods online based on the joint positioning, in reality, considering all input scenarios, there is no perfect human pose estimator and sometimes lower-ranking ones perform better than higher-ranking methods in challenging situations or other specific situations. The post-processing methods can be applied to any HPE method to improve the result of the pose estimator.

2.2.2.3 Suggested Solutions

Here a range of potential solutions for improving the output of the model-based pose estimation methods with SMPL output are introduced.

The problem of lack of 3D model-based data sets is addressed in this research by recording and producing a data set of ground truth mesh and synchronized videos. Unlike most current data sets, the actions in the recorded data set are not daily activities but more challenging exercise motions with a range of different actions aimed at better evaluation of the pose estimation performance. There is repetition in each performed action and the same actions are repeated by different subjects which is also unique among the existing datasets. This enabled us to implement action modeling in motion estimation part and also be able to use the dataset for motion evaluation purpose.

The recorded data set focused on self-occlusion scenarios, a widely known problem with monocular video pose estimation methods. The repeated actions in the data set enabled us to produce action-specific data and train a machine-learning model to address the problem of self-occlusion. Using the rotation-based evaluation metric, is shown an improvement in the overall pose while training a model with corresponding correct and incorrect pose data. Supervised training on the action-specific data can also significantly improve the incorrect poses due to body part occlusion and reduce the error of the evaluation metric even further.

One solution to problems such as occlusion and unnatural poses can be using the deep learning based 3D key point prediction of limb's endpoints and applying inverse kinematics (IK) to guide the model arms and legs to the correct destination. This can be done by aligning and scaling the deep learning predicted skeleton structure with the 3D mesh model and finding the 3D position target for the 3D human model limbs. This will also improve and close the gap between model-based and skeleton-based predictions. Looking at the current research papers, the idea of applying IK is tried and combined with different methods. This method can be considered as the initial stage or be improved to have distinction from current methods.

2.2.2.4 Existing Solutions

In this section, previous research about the problem of occlusion and self-occlusion in human pose estimation is listed.

2.2.2.4.1 Self-Occlusion/Occlusion in HPE The video input of our project is a single-person, full-body monocular video and no external object or person is between the subject and the camera. Hence, the main source of occlusion is self-occlusion of the body parts due to having only one camera v. Current HPE methods are ranked based on the achievable pose error and not by handling extreme cases so their ability to evaluate self-occlusion or external occlusion may vary. Most of the more recent work on occlusion is working on severe cases, for example being occluded by another person, external objects, being outside of the frame, crowded scenes or having only a single image. Self-occlusion still can be an issue with some state-of-the-art methods at various levels depending on their architecture.

Human pose estimation algorithms are divided into top-down and bottom-up approaches. The bottom-up approach has lower accuracy, but they are better at handling the occlusion because of considering the joint relationships. Some examples of such bottom-up methods are ORPM [204] (occlusion robust pose-map) which uses the joint location redundancy, that can be applied to only extremity joints to infer occlusion, and Xnect [205] that encodes the joint's immediate local context in the kinematic tree to handle occlusion, and depth aware part association algorithm [206] that is robust to occlusion. Apart from bottom-up approaches, the occluded pose estimation problem is solved by various methods: a category of methods first finds the 3D skeleton that has some missing joints and then completes the missing joints with statistical and geometric models [207], [205], [208], [209]. The attention mechanism is used to enforce the model to focus on non-occluded areas that result in more robustness in the final output [210], [211], [212]. If the input is video, it is possible to use temporal information [213], [214], [215], [216], [217], [218], [219], [220], [221]. If the complexity of occlusion in the real world is higher than the available data, data augmentation methods [222], [221], [223], [224] can solve this problem. In severe occlusion scenarios where there is no or little cues, some recent methods regress multiple plausible poses [225], [226], [227], [228].

For better handling of occlusion in [229] two separate upper and lower body parts are used. Similarly, in [230] two separate models for occluded and non-occluded body parts are trained to resolve occlusion by another person. In [231], [222], [213] a multi-stride temporal deep neural network and a discriminator to check the validity of the pose are employed. Some key points are masked during training and final optimization based on projection of 3D key points to 2D key points is performed. A pair of occluded 2D key points and correct 3D key points are used for training. Similarly, in [232] this pair of 2D and 3D poses are used for training graph convolutional neural networks and transformers. Synthetic data is used for data augmentation. More analysis based on the number of occluded 2D joints and number of occluded frames is presented and the result is compared to the existing HPE frameworks. The data augmentation approach used in [231] is in the form of discrete/continuous framewise and point-wise occlusion. A Bayesian approach for 2D Human pose estimation that employs both local and global information is used in [233]. A CNN-based pose estimation like Open Pose is used for detecting the visible joints, then a Bayesian pose estimation is applied to fill the missing joints. The task of pose estimation is divided into two parts: visible 2D key points detection and occluded 2D key points reasoning [230]. Motivated by the advantage of explicit modeling compared to the previous methods and implicit modeling, they have proposed a deeply supervised encoder distillation to solve the occlusion problem. To create occlusion labels on the datasets to enable this explicit reasoning, a skeleton-guided shape-fitting method is introduced. The problem of 2D occluded key points estimation is also solved by [213]. Confidence heat-map of key points and optical flow-based constraints can find the incorrectly estimated key points. Then the incomplete 2D pose is used to infer the complete 3D pose. Synthetic data with plenty of occluded 2D poses are produced using the projection of the cylindrical man model into the image plane.

The problem of 3D human pose estimation is treated as an unsupervised learning problem in [234] to avoid the common issue of supervised methods when dealing with unfamiliar out-of-distribution data. To quantify the prediction uncertainty, model-free joint parameters and model-based pose parameters inference are used. The adaptation process aims to minimize the uncertainty for unlabeled data and maximize it for an extreme data case. In [235], Instead of using synthetic occlusion datasets, a 3D occluded dataset is built. Non-occluded human data is used to learn joint-level spatial-temporal motion prior to inferring occluded humans in a self-supervised strategy. Similar to the previously mentioned research, a 2D input skeleton with missing joints was used in [236] as an input to infer the 3D skeleton. A temporal dilated CNN along with a mechanism for removing the 2D occluded joints using confidence values is introduced. Comparison with a range of current static and dynamic HPE methods is made under various levels of occlusion, for example removing around 90 percent of the joints from the input data.

Some previous research also worked more specifically on the problem of self-occlusion. For example, in [208] to reduce the ambiguity in inferring a 3D pose from a single image, both kinematic and orientation-related constraints are used. This is done by projecting the 3D model into the image and some synthetic views and improving the ambiguity. The occlusion problem in a single image is solved by using the Euclidean Distance Matrix in [237]. A Markov random field is used to represent the occlusion relationship between human body parts in terms of occlusion state variables [238]. The depth ordering of the body parts is considered as occlusion states that need to be estimated. The dataset is labelled according to how the depth ordering of body parts and self-occlusion is changing during the video. The inference is done in two separate stages: body pose inference and occlusion state inference. A new cue is used in [239] to address the problem of self-occlusion. The self-occlusion handling process uses the torso orientation as a cue. A new occlusion-aware graphical model is introduced in [240] that explicitly models both self-occlusion and other occlusion to improve the robustness to occlusion. The model learns the part-level occlusion relationship from data and infers the occlusion states of parts explicitly. There are a few other work that tried to model self-occlusion. In [241] pixel level hidden binary variables are used for self-occlusion reasoning. Some others try to model self-occlusion holistically. In [242] self-occlusion of pedestrians is modelled in a joint shape and appearance tracking framework. In [208] self-occlusion reasoning is treated as post-process with Twin-GP regression for 2D pose rectification. In [243], an LSTM-based method that uses multiple frames is proposed that considers both spatial and temporal correlation by connecting multiple LSTM networks, suggesting better solutions for self-occluded scenarios. A sequence-to-sequence LSTM framework is shown to be effective in handling self-occlusion [244].

In summary, most of the previous research in self-occlusion is related to a single image and 2D human pose estimation or 2D silhouette. Some of them are demanding in terms of data, for example, they are taking advantage of multiple view training data and need a large training dataset. The depth ordering of body parts is also required to be known beforehand. Some are video-based method but being part-based they are unable to estimate invisible body parts or limbs. The most widely used human body model in the previous work on self-occlusion is the body parts model. Existing work on occlusion problems in SMPL-based methods [211] worked on occlusion by other objects and do not perform well in self-occlusion scenarios in which body parts are occluded by other body parts.

This research focused on the problem of self-occlusion in video-based 3D human model estimation which is an under-investigated area. We also believe this is the first work on the problem of self-occlusion in SMPL-based methods. Our approach does not require a complicated or large training dataset and only uses single-view video data. Furthermore, it is able to estimate invisible motions and body parts, unlike the previous video-based research.

2.2.2.5 Suggested Solutions

In this section some current work in HPE that incorporates the suggested computer vision concepts are discussed to ensure the distinction of the proposed methodology from the currently existing research.

2.2.2.5.1 Inverse Kinematics (IK) for HPE Inverse Kinematics can improve the 3D model estimation when the predicted 3D positions are better than the estimated 3D model parameter. This might not be the case in all challenging scenarios.

The IK method uses the joint positions to place the joints of the parametric models in the desired place. The current evaluation metrics are solely based on joint position errors. Deep learning methods (e.g. [245]) can infer 3D joint positions in challenging scenarios. Despite that, the position data itself is incomplete and using only analytical IK, leads to correct joint positions but incorrect orientations. This is resolved by using a combination of IK and other methods or enforcing more constraints on the model. In [246], a post-processing refinement is applied to the result of the IK. A combination of analytical and learning-based (Neural) Inverse Kinematics is used in [247] where the rotations are estimated jointly from the image and 3D key points. It has been improved recently [248] by learning both forward and inverse processes and using the error from the learned manifold of plausible human poses. In [229] the pose is divided into the upper and lower body for increasing robustness. The 3D pose (joint orientations) is regressed by deep neural networks. Then the IK is applied to the result along with applying some constraints to the body model using the camera information and known limb parameters.

2.2.2.5.2 Optical Flow/Kalman Filter for HPE Optical flow is used as a temporal feature for improving the result of 2D and 3D human pose estimation results. For example, in [249], [250] using dense optical flow as an additional feature, showed improvement compared to using only RGB input. In [251] the under-constrained problem of human pose and shape estimation showed improvement in the "optimization-based" method when the constraints based on silhouette and optical flow were added to the joint position-based constraints. The use of flow-based features and CNN for real-time/online processing is investigated in [252]. Optical flow is used in [253] to improve the robustness and compensate for the issues caused by the temporal continuity error of the 2D pose estimator. A new representation called Pose Flow is introduced in [254] that can jointly describe pose and motion. Temporal differences are used instead of optical flow in [255] to use motion cues. In [213] optical flow in conjunction with joint confidence maps (heatmaps) are used to realize which joint is occluded and remove them from the 2D estimations because of being inaccurate.

2.2.2.5.3 Attention Mechanism for HPE A temporal attention-based mechanism is used in [256] for the purpose of 3D human pose estimation. The attentional mechanism adaptively identifies significant frames from each deep neural network layer leading to better estimation.

2.3 Human Motion Evaluation

Human motion analysis is a rapidly growing field that is focused on studying the movement of the human body to gain a better understanding of it. Some useful applications of human motion analysis is in the areas such as healthcare or sport science where the tasks are not currently fully automated. For example motion analysis can be used in the development of musculoskeletal models to understand the effect of different movements on the body, and this can help in preventing injuries. Kinesiology which is studying the mechanics of human movement can also benefit from human motion analysis. In the area of physical therapy, human motion analysis can lead to a more effective treatment plan for the patients with better outcome and less pain. In ergonomics it can be used to design more efficient and comfortable workplaces by study of worker movements. The study of athlete motions can help optimizing the performance and choose the best training for each individual.

In this work, the human motion is analyzed with the purpose of evaluating (scoring) the performed sport action by an athlete more specifically martial artist. Human motion evaluation is concerned with how well a certain action is performed and, in some cases, providing some feedback on motion improvement. This branch of human motion analysis is different from action recognition/classification (labelling the actions in a set of predefined categories), action detection (specifying the beginning and end of a specific action in the motion sequence) or action prediction (predicting the future motions based on the previous history in incomplete motion observation). Many emerging applications for the human action evaluation are introduced in the recent years including healthcare, skill training and sport scoring.

2.3.1 Previous work

Due to the diversity of research in this area, categorizing all the research is a difficult task. The main research that introduces some insights into the main problems in a complete human motion analysis pipeline (main taxonomy of the research problems) are provided in surveys. According to [124] the research can be divided into initialization, tracking (sometimes includes background segmentation), pose estimation, and action/activity recognition. In [123] the paper is divided into tracking (background subtraction, deformable template, flow, probabilistic models), 3D pose recovery from 2D observation, and data association and body parts. The topic of motion synthesis, which is more discussed in the area of computer graphics was also mentioned [257], [258], [259]. Another important topic is about the taxonomy of the research in this area that can be divided into whether monocular (2D) or multi-view (3D) data are used for processing, and if the human body model is in 3D or in 2D.

2.3.1.1 Classic machine learning methods

The statistics of human motion (e.g. mean and standard deviation of the timeseries) are important factors in understanding the motion in various areas such as biomechanics, computer animation or ergonomics.

Handcrafted features are very common in the classic solution of two stage feature extraction and feature learning. Most famous feature detectors such as spatial–temporal interest points (STIP) [260], histogram of gradient (HOG) [261], histogram of optical flow (HOF) [262], scale-invariant feature transform (SIFT) [263], and motion boundary histogram (MBH) [264]. Depending on the application more custom features can also be developed. In the next step which is learning the action model can be trained using the methods such as a bag of words (BOW) [265] or the hidden Markov model (HMM) [266] and an evaluation function result shows the quality/score of the motion. Some issues can cause less quality in the result of this method, for example, in the cases that variety of action types are performed, using one handcrafted feature might not be possible. Also, if the action is complex or time duration is long, this method doesn't produce a satisfactory result.

2.3.1.1.1 Human Movement Features Human motion data can be diverse and in different formats and may be generally large and complex. Some of the most common forms of this data are a sequence of joint positions/rotations, RGB-D data, silhouettes, RGB data, etc. In order to process and perform inference from this data, it is necessary to achieve some more abstract representation of this data. Furthermore, depending on the type of study, some aspects of the data might be unimportant in decision making and this extra information can be removed for easier processing. The aim of designing a movement feature is highlighting some aspects of movement which are important to us and can help in achieving the task. In this section general categories of movement features are introduced and in the next section the features related to the martial art motions are discussed. Movement features can be grouped into three areas:

1. Subject Features

2. Transition Features

3. Motion Features

2.3.1.1.1.1 Subject Features Human bodies are different depending on the physical characteristics of the subject. This morphology has an effect on the perceived motion as it might be an important prior data that can help in the motion analysis step. This type of feature is not directly related to the motion, but is about the subject itself. Some examples can be specific bone size, the distance between the various joints in human body skeletal representation, the height of the subject, width of shoulder/hip or rotation limit of each joint. Measurement of these features is easier using motion capture systems when there is access to the subject, otherwise it can be complicated and require extra data to be done correctly. When using a motion capture system anthropomorphic characteristics of the body can be measured in T-pose/rest-pose. To extract body proportions more than one pose is usually used. Distances in these systems are shown with different statistics such as mean, median or standard deviation. It should also be noted that high quality captures should be used for extracting physical properties to reduce the effect of issues such as occlusion, capture space and reflections.

In image-based methods, because of an unstructured form of the data compared to the MoCap data, there are fewer descriptors about the subject. Silhouette width and height are one of the features that are used [267], but they are not view-independent and might not be a reliable measure in some situations. Body parts length was used in the model-based application [268], distance between head and pelvis, pelvis and feet, head and foot plus some dynamic distances like distance between the two feet during walking are used as the feature vector. A synchronised camera system is used for marker-less human motion capture and estimation of a 3D model composed of twelve 3D cylinders where the length of the cylinders is extracted as a subject-specific feature [269]. Bone length and subject height are also computed from the 3D human data captured by Kinect [270], [271].

Pose Features: Static features are extracted from the pose of the person at any given time, and they are defined independently from other poses happening at different times. Therefore, they are not affected by factors such as speed of motion or the type of action performed by the subject. Since they are defined as a function of human pose, their input space will be all possible poses that can be generated by the subject which is a continuous nonlinear space [272].

Appearance (Silhouette) based Features: This type of feature uses the human silhouette or some other extracted information from it, as a pose feature. Using a silhouette on walking videos with a fixed velocity is used in [273]. First, the area of the silhouette in the first and last frame is manually specified as a bounding box, and because the moving speed is considered to be constant, then intermediate bounding boxes for other frames can be computed. In the final step, background subtraction is done in each of the frame bounding boxes and the silhouette is extracted as a motion feature. This appearance-based approach is combined with model-based methods by defining each area of the silhouette as a specific body part (for example: right arm, head, torso, etc.) And each of these parts of the silhouette is considered as a pose feature [274]. This method was tested on the USF dataset [275], where some body part features showed a worse discrimination ability than others. Especially the body parts related to the right side of the body which was mostly occluded in the videos. Therefore the discrimination depended on whether the body parts appeared in the image or not and it was not affected by the shape of each extracted part. The silhouette itself can also be a source of other information and features, for example the contour of a 2D silhouette is extracted by Wang et al. and then is unwrapped as 1D signal [276]. The unwrapping procedure is done by computing the distance of each pixel on the silhouette contour (starting from the top in a clockwise direction) and the center point of the silhouette. The initial length of the produced 1D signal will be equal to the number of pixels on the contour. The magnitude of the signal is normalized and the length of it is re-sampled to a fixed length. Poor quality of silhouette images could affect this feature [277]. Further improvement to this method is done [278], [279], [280] to increase the effectiveness. The width of the contour (distance between the right and left pixels in each row) is also used as a feature vector [281], [282] in various applications.

Skeleton Joints Rotation and Distance Features: The second group of features that are a function of the human pose in the image, are model based (skeleton based) features. The human skeleton model can be defined in 2D or 3D, so these features can be defined in 3D when spatial data are computed or is available. In rotation-based features, the degree of freedom of each joint in the human model is equal to 3 [283], while in a 2D skeleton model each joint (connection between body segments) has 1 DOF. For simplicity and accuracy in measurement of the joint angles in the two-dimensional space, usually the sagittal view is used, and motion also is constrained to this view. Joint angles used in [284], [285], and shows as a periodic signal in [286], [287] using joint angles. The hip and knee angles [288] in the sagittal view of the human body, are used in a model of the human leg as a 2D pendulum. The whole human 2D body model with nine key points is used by Yoo et al. [289]. Six joint angles from this model are extracted as a feature. Similarly, in [290], [267] body parts are modelled as a rectangle/truncated cones, and angles between these parts are measured. Measuring the joint angles only in sagittal view, means that the solution is not view-independent, and its application is limited to such an input data. As a view invariant method, 2D angles are extracted from the 3D skeleton model by projecting the 3D skeleton to a 2D plane. The lower limb is projected into the sagittal plane in [291], [292], and the joint angle is found from the projected trajectory, for the purpose of human motion retrieval. The same method with additional projection onto transverse and frontal planes is used in [293].

In a 3D human model, the joint rotation angle can be also described in the 3D space. This 3D rotation value can be expressed with different values depending on how the rotation of the joints is modelled in a specific type of coordinate system for the human body model. Generally, there are two main types of coordinate systems for assigning rotation/position values to 3D skeleton joints. In the global (scene) coordinate system values are computed with respect to the world coordinate system in the scene, while in the local coordinate system values are with respect to some points on the human body itself. The local system is camera view-independent and is further divided into two types: a hierarchical and absolute coordinate system. In the hierarchical system the rotation of each joint is expressed with respect to the parent joint while in the absolute system the root joint (usually pelvis) is the reference point. 3D Joint angles in the quaternion form can be acquired from the wearable gyroscope and accelerometer sensors to be used for action recognition [294]. Other motion capture systems such as Kinect or optical motion captures like Vicon can directly give the rotation values.

If only joint positions are available, it is possible to convert joint positions to joint rotations and vice versa using inverse kinematics (IK) and forward kinematics (FK) respectively. Because converting joint positions to joint angles using IK is computationally expensive, a simplified computation is usually used. One of the common solutions is computing the angle between the two bones (body segments) attached to the joint in the body model (for example humerus and forearm bones for elbow angle). Another way for approximating the angle is definition of some planes using the human body and compute declination of the bone from that plane. For example the shoulder angle is computed by first defining a plane on the shoulder-thorax link and another plane on the shoulder-elbow link. The angle between these two planes is measured and projected on the first and second plane respectively. This results in two angle values describing the 3 DOF joint rotation. A human skeleton of 18 joints is used in [295] with dividing joints into 1DOF (elbow, wrist, knee and ankle) and 3DOF (hip, shoulder and spine) groups. The joint angle of 1DOF joints is defined as the angle between the related bones, and the rotation angle of the 3DOF joints is approximated by measuring the declination of the bones from sagittal and frontal planes. The roll rotation is ignored in this approximation. Joint rotation approximation in [296] is defined as an angle between two vectors each specified by two body joints. The vectors used in feature extraction were not limited to the bones of the skeleton. Other vectors using different pairs of joints were also defined. In [297] angles between a spine vector and humerus, radius, femur and tibia vectors are used as features. The rotation representation of the rotation matrix is used in [272] and all joint rotation values are relative to the torso which is considered as an origin. The orientation of the bone vectors is used as a feature in [298], if the skeleton rotates without any change in the pose. Using this method will result in a different set of features for the same pose and this is a disadvantage. 3D joint rotations are represented in the quaternion form in [299], while in [300] joint angles computed in Euler and quaternion forms are used for finding acceleration and angular momenta.

Joint distance features measure the distance of the joint to a certain point. There are two common types of this feature:

- Joint to joint distance can involve any arbitrary pair of joints depending on the definition of the feature.
- Joint to plane features where the plane can be defined relative to the human body. For example the body frontal plane, which is made by left hip, right hip and root joints. It can also be a fixed absolute plane in the scene for example the floor plane.

Joint to joint distance is used in [301], [302] and [303], [294], [304] as a pose feature. The distance-time signal is used in [301], [302]. A similar approach is used in [303] for action recognition with normalizing joint to joint distances. Anthropometric properties of different human bodies can affect the distance feature, while in action recognition it is desired that the feature is only a function of the human pose not the shape or proportions of the human body. This is the reason for normalizing the distance feature. The length of the path between two joints in the kinematic tree (3D human skeleton) is used as an indicator of human size and the direct 3D distance (distance features) are divided into this value to be normalized. A motion capture classification method proposed in [305], [296] uses the joint to plane distance as features. These types of planes are used in the definition of features, they can be defined by: (1) three joints, (2) one joint and normal vector of the plane (defined by two joints) or (3) points in the scene. Depending on negative or positive sign of the computed distance between a joint and specific plane, a set of binary features such as right hand above head, right foot behind left leg, etc. are extracted. Joint to plane distances are also used in [304] for action recognition where the distance of some body joints like hand, pelvis, etc. and the floor plane is measured as a pose feature.

2.3.1.1.1.2 Transition Features If we consider the human motion as a series of poses each happening in a the specific time instance, it is also interesting to study how the transition from one pose to the next poses is done. Like the pose features, transition features describe an information about the motion at a specific time instance, but for computation of their value, data from the time instances around (before or after) them is used. At least two consequent poses are needed to compute a transition feature, but usually no more prior information about the whole motion signal is needed. Instantaneous velocity, the rate of change in position for the time period, is an example of transitional features. The sampling frequency of the motion capturing technology that provides the input signal specifies the time period value. For example in the video or Kinect, the input frame rate is 30 Hz (fps) and optical MoCap devices have a higher sampling frequency like 120 Hz. The transition features can be computed from appearance based or model-based information in the input data, so depending on the given data various forms of features can be defined.

Transition features based on appearance use the image frames in the video directly. The optical flow technique [306] which is used in video processing is an example of such features. Optical flow in conjunction with silhouette deformation and pixel-based silhouette difference is used in [307] for video action recognition. Silhouette deformation is defined as changes in the Chamfer distances [308] between points on the silhouette in two consequent frames. Silhouette difference between two frames is computed as the pixel wise difference between the contours. Combined Local Global (CLG) motion flow, which is composed of optical flow and shape flow features, is introduced in [309]. Optical flow is represented by velocity vectors that are derived from the normalized flow of each quadrant of the silhouette image, characterizing the recognition of motion in action with less noise. The shape flow features are extracted from the global flow of the shape. The global flow of the silhouette images is computed by applying robust description of geometric orthogonal moments [310], flow deviation and anthropometry flow, which characterizes the recognition of shapes in action.

Model based transition features use data from the human body skeletal model as an input. Cartesian or angular velocity of the joints that are extracted from the position or angle trajectory of the joints are examples of such features. The angular joint velocities are extracted from 2D skeleton data in [289]. In [271] the joints of a 3D skeleton are projected on 2D views, and the angular velocity is then computed. 3D angular velocity of the human skeleton joints and the corresponding rotation angles in the form of quaternions [299] and angular momenta [300] are used as features. One of the problems in computing the velocity from the joint position signals is the appearance of high frequency errors (temporary spikes) due to noise in the joint motion data captured by various means such as Kinect, optical MoCap or video processing. To solve this problem motion smoothing techniques are used. Low pass filters are suitable for removal of high frequencies and convolutional low pass filters are usually used for this purpose [311], [312], [305]. In addition to the absolute value of the velocity [270], [313], [271], the direction of the velocity vector can also be used as a feature [294], [299]. The concept of relative velocity, which captures changes in joint positions relative to each other, can also be explained by the velocity vectors. The relative velocity features are used in [272] and this idea is extended in [305], [296] for motion annotation purposes. They define an absolute value for the velocity of a joint j in the direction of a specific vector $\overrightarrow{\omega}$. The vector $\overrightarrow{\omega}$ is made using the position of two joints on the human skeletal model. The acceleration is another transition feature which can be derived from velocity values and is used for motion segmentation [307] and motion generation [300] applications. In addition to kinematic features such as velocity or acceleration, kinetic values such as force and torque can be extracted. For example, in [293] force plates and motion capture are used for extraction of joint reaction forces during walking. The measured forces on lower limb joints from sagittal, frontal and transverse views are evaluated in their work. To find the correct time instance for transition from one motion sequence to the next sequence, a transitional feature is defined in [258]. Consecutive poses in a window with predefined length are used to make a large point cloud. The extracted feature from the point cloud will help to solve the mentioned motion segmentation problem.

2.3.1.1.1.3 Motion Features The motion features are assigned to an entire motion sequence or a set of frames. Other lower-level features introduced in previous sections can help in this task or it can be done manually by the user. Like previous types of features, the motion features can be in the appearance based or model-based category. An example of appearance-based motion features are temporal templates introduced in [314] for action recognition in 2D RGB videos. A temporal template is an aggregation of all frames corresponding to an action, into one image in which each pixel is a function of the foreground motion at the specific time instance. The foreground motion is defined as the image difference of silhouette image of two consecutive frames. The MEI (Motion Energy Image) feature is defined as the aggregation (union) of such differential images over all frame sequences. The contribution/weight of each frame is equal in the computation of the MEI feature, while in the MHI (Motion History Image) feature each frame can have a weight which depends on the time instance of that frame. The temporal templates are used in [315] for walk cycle actions. The feature Gait Energy Image (GEI) is introduced which is like the definition of the MEI feature previously discussed. Assume having the silhouette images of the human gait over a series of n frames, GEI is computed by the summation of all the silhouette images divided by the number of frames n. In the resulting image the brightness of each pixel shows the duration of human walking occurrence at that pixel position. Similarly, the Gait History Image (GHI) features [316] is inspired by the MHI features. MHI and GHI can preserve the temporal variation in the resulting image feature by gradual intensity distribution on the moving trace, so the brightness of pixels in that image is also a function of frame time and increases gradually. More improvement on temporal templates are introduced in [317], [318], [319], [320], [321], [322]. To reduce the effect of incomplete silhouettes, in [323] Frame Difference Energy Image (FDEI) is introduced as a robust dynamic gait representation. Temporal templates are also used in 3D, where such a data is available through multiple camera setup or depth sensors. The concepts such as Motion Energy Volume and Motion History Volume are introduced in [324], in which the 3D silhouettes were created by the RGB videos captured from different views using a multi-camera setup. Gait Energy Volume using depth map of the walking sequence in frontal view is used in [325]. Apart from temporal templates, statistical image description is another approach to describe a sequence of silhouettes that represents an action. An Example of image statistical description methods is Hu moments [310] that is used in [326] to discriminate shapes in a scale and translation invariant way. The matrix of moment's mean and covariance is computed through the entire movement that is analysed. The 3D extension of Hu moments is introduced in [327]. Another statistical image descriptor is the mean shapes. A series of mean shapes is extracted from a series of silhouettes of walking cycle using Procrustes shape analysis [328] [290], [329].

Model based motion features are using the joint motion data of a sequence to compute a feature. An example of such features is the average velocity or traveled distance of a joint in the motion sequence related to an action. Features such as cadence of specific steps [330], stride [271], [330], [270] or length of walk cycle [271] are in this category. Features such as velocity, distance and action duration are used in [273] for action recognition. Statistical descriptors of human motion sequences such as mean, standard deviation, minimum and maximum values, median, median absolute deviation and modus are another example of motion features. The mean, standard deviation and max and min values of joint rotation angles are used as a feature in [271], [292]. Signal processing techniques can also be used to find some patterns in the motion signal, for example [271] examined the periodic pattern in the signals such as arm joint rotation, knee joint rotation, hip joint rotation, distance of head to floor and distance of COM (centre of mass) to the floor. They have used the autocorrelation (correlation of the motion signal by itself) as an identifier of periodicity. The position of 3D joints in different time instances during an action can be considered as joint trajectories $XYZT$ describing a human action. In [331] a set of action bases is generated where any action can be defined as a linear combination of these action bases. Similarly, features of the action can be computed with linear combinations of action bases and features are represented as a vector of the corresponding coefficients for this linear formula. Another approach in [332] mapped the action data from the space-time domain to the space domain which means showing an action as an image. Each human body model joint coordinates in XYZ had the corresponding RGB colour and converted to a pixel on the image depending on its position.

2.3.1.1.2 Movement Features Processing The features related to human motions introduced in the previous sections, are usually vectors with real values. The resulting feature space required for analysis can be very high dimensional, and this will cause fast growing of the feature space and consequently sparsity of data which cause problems when statistical significance is required for data analysis. The values of the features also can contain error values and noise values that can cause issues in further interpretation of the input through the extracted features. Therefore, there is a need for processing the extracted features using some methods before performing the analysis.

2.3.1.1.2.1 Quantisation The extracted features of human motion data are usually in the form of real numbers, these values can deviate from the real-world range of the features which is constrained by different factors including the physics of the human body. For example, rotation angle of knee cannot be 360 degrees or there is a limitation to the joint velocities too. In feature quantization we use the fact that slight differences in feature values are associated with similar human motions, by defining a set of predefined bin values for assigning the exact values of features to them. Quantized features were used instead of numeric features in [333] and showed invariance properties in recognition problems. They are also used for indexing in [296]. In a more general form, feature quantisation can be defined as a way of discretizing the feature values to integer numbers. For example, in [305] 39 different features were defined that further quantised to binary classes $0, 1$. The resulting binary values create binary feature vectors that can be more easily clustered in comparison to feature vectors with non-quantized values. The thresholding process is a common issue with quantization methods. That is because the fluctuation of input around the threshold value will cause fluctuation in the output. This is not desirable because a small change in the data due to some artifacts can produce a different value. This issue is solved in [305] by using hysteresis thresholding which have a two threshold values for the increasing and decreasing trend in the input. This will prevent the output quantised values from fluctuation by creating a band for thresholding.

2.3.1.1.2.2 Dimensionality Reduction The features extracted from the human motion signal have high dimensionality also known as the curse of dimensionality. It is preferable to reduce the dimensionality of these feature vectors as much as possible. Dimensionality reduction allows both computational and memory advantages (i.e., reducing the training time and the amount of required training data) and reduces the risk of overfitting, by searching for a low dimensional feature subset. One of the most used dimensionality reduction methods is called Principal Component Analysis (PCA) [334] that can decrease the dimensions of the feature vector by keeping the most important values. If we consider each element of the feature vector as a variable that jointly describe the feature data in the n-dimensional space, principal components are new variables that are made of linear combination of these initial variables. The new variables (principal components) are chosen in a way that more information about the initial variables can be explained and compressed into the first principal component and most of the remaining information can be explained with the second component and so on. Principal components should also be uncorrelated to each other.

PCA is used in [335], [298] for feature dimension reduction. It is mentioned that although dimensionality reduction is more efficient, the resulting feature vectors will be more sensitive to deformation of input poses [336]. In [276] PCA is used for converting the high dimensional features to a low dimensional eigenspace. There are other approaches similar to PCA, for dimensionality reduction like singular value decomposition (SVD) that are used for handling the large data. More ideas and methods for dimensionality reduction can be found in [337].

2.3.1.1.3 Human Motion Analysis Human motion analysis research is motivated by applications over a wide spectrum of topics and focusing on diverse research areas such as body structure analysis (model based/non model based), tracking (single camera/multiple camera), action recognition, action detection, action evaluation, etc. An example of application areas for human motion analysis in action recognition and detection is gaming [338], computer vision [339], animation [259], surveillance [340], human-machine interaction [341] and robotics [342]. The research efforts until recent years were mainly focused on using video sequences or RFID sensors, and later with the advent of low-cost depth sensors, research on 3D joint positions as one of the most important data that can indicate the body motion is increased and recently, research efforts also been devoted to human action evaluation. Some examples of research in this area are applications such as interactive gaming [343], rehabilitation [344], self-learning and practising sports [345], dance [346] and martial arts [347]. Assume having an input human motion, the problem can be defined as finding (1) the corresponding predefined category for the motion or (2) similar motions to the input motion. At the core of such analysis is a distance function that besides the input human motion takes a dataset of human motions (and for the first case corresponding target values), and outputs a real number related to the category or similarity of the input motion. In the following sections these two common approaches in motion analysis are introduced.

2.3.1.1.3.1 Motion Classification The learning methods, learn patterns from the training data consist of a dataset of motions paired with their corresponding target values. The quality of training data plays the major role in the success of these methods, also it is important to avoid overfitting to the training data which is only a subset of all possible observations in the real world. Supervised learning methods are very common in these types of problems and are reviewed in this section.

Neural Networks: Neural networks are computational methods that try to find the underlying relationship between a set of provided data like the way human brain do it through learning things by observation. For complex operations, we would need a multilayer neural network where perceptrons are arranged in interconnected layers. Neural networks were also used in fall detection problem [348]. Convolutional neural networks for action recognition in Mocap data is used in [349], [304], where the MoCap data are converted to image inputs of the CNN.

Support Vector Machine (SVM): Support Vector Machine is another supervised learning algorithm that can solve classification and regression problems. An SVM kernel is a function that converts the low dimensional input to higher dimensional data and in this way a non-separable problem becomes a separable problem with the data transformation method. All possible poses of a person in MoCap data, creates a non-linear input space that can be classified using SVM. SVM is used in [272] for action classification purpose using quaternion joint angles and their velocities as features, it is also used in [294], [298], [350] and [271], [351].

Naive Bayes and HMM Stochastic Methods: The input data in some machine learning problems can be in a form of sequence. Human motion data can also be considered as a sequence in which the sequential supervised learning methods can be applied. The sequenced training data is in the form of (x, y) pairs of inputs and targets, and in most cases these sequences have sequential correlation. This means that it is very likely that nearby x and y are related to each other. In sequential supervised learning the aim is to find all y values in the sequence $y_1, y_2, \cdots, y_m$. This is slightly different from time-series problems in which all the previous observed y values until the time t are available and can be used for predicting the next value of y at the time $t + 1$. Two common algorithms for sequence classification are HMM (Hidden Markov Models) and Naive Bayes methods.

Naive Bayes classifier is used in [271] using a set of subjects, pose and action features, similarly bone length, bone speed and length of step used as a feature in [270]. Naive Bayes is also used in action recognition problem [352], [353], [354], [299]. HMM based classifier is used in [353] and [281] for different problems. Similar application of HMM classification for human motion analysis can be found in [355], [356], [280], [357],[299].

Decision Tree and Forest: If we have a test set of data, the classification problem can be solved by asking a series of questions about the characteristics and attributes of the input data. After receiving each answer, a follow-up question can be asked until reaching a conclusion about the target label of the input. These series of questions and their answers can form a hierarchical structure of a decision tree. Decision forest is the collection of decision trees and randomized decision forest is based on selecting random subset of features during learning in order to decrease the effect of feature correlation. Decision tree, decision forest and randomized decision forests are used in [294], [270], [358], [359].

2.3.1.1.3.2 Motion Similarity The human motion analysis methods described in the previous section were based on machine learning. There are another group of analysis methods called similarity-based methods that does not use learning or any predefined set of categories for output and are used in cases where there are few positive examples are available. Since there is no training stage in these methods, they rely on a distance function d that can compare two different motion sequences. The quality of features and data also plays an important role in the success of these methods. The distance function is dealing with two time-series data of different lengths, so the motion sequence alignment methods [360] need to be used to produce a sequence of corresponding (frame, pose) pairs for both motion sequences. Various motion alignment methods are used in the literature, including Uniform Time Warping (UTW) [361], Interpolated Uniform Time Warping (IUTW) [362], [363], Uniform Scaling [364] and Dynamic Time Warping (DTW) [365]. The distance functions used in the similarity methods are usually implemented metric measures over pose features such as joints positions, rotations or velocity. These features can be defined as a vector with n dimensions. Different distance measures such as Euclidean, Manhattan, Cosine and Quadratic form distance can be defined over the vector space.

2.3.1.2 Deep learning based methods

In the recent years using convolutional neural networks (CNNs) in the field of action recognition replaced using handcrafted spatial and temporal features. During this short number of years, a lot of progress has been made in different areas such as using CNNs for learning features and action recognition in single images, using 3D convolutions in spatiotemporal volumes and use of recurrent networks for modelling temporal transitions. CNNs show a very good performance in solving computer vision and image processing problems. They are also used to extract deep features from human motion, for example in action recognition problems. Examples of such networks are the 3D convolutional network (C3D) [366], [367], long-short term memory (LSTM) [368], [369], and the two-stream convolutional network [370], [371]. Long videos can be processed with frame-based aggregation methods or video based temporal relation methods. The frame based methods use a pooling operation for fusing extracted features of each frame to form video based features. The video based methods use a recurrent neural network to model the temporal relationship between frames.

Monocular (single image) action and gesture recognition using deep architectures is like object recognition solutions. In both applications feature learning and classification is jointly done by the deep neural network in an end-to-end manner. The dimensionality of the network's input is also expanded to using spatiotemporal data blocks, as mentioned before when having such cases the dimension of the convolution filters will be in 3D [372], [373], [374]. The spatiotemporal framework is extended in [375] to include a large number of frames at the same time, which is called long-term convolutions. The model is shown to have an improved performance in video classification. In gesture recognition, the frames are concatenated in [376]. Then a 3D convolution is applied, and the result is a multiplication of the class probabilities in the double resolution framework. The use of hand-crafted features as a pre-processing step such as optical flow or gradient also is common in the previous work. In [374] some handcrafted filters are used for low-level feature processing that later act as an input to a CNN that can learn spatiotemporal features and also do the classification. Optical flow estimation is used in [375], [377], [378], [370] for learning the action models. Another similar approach in [379] used DeepFlow for a patch matching problem. In a different approach spatial (scene characterization) and temporal (motion dynamics i.e., optical flow) stream of data is separated and given to a pair of convolutional networks in [370]. Later the prediction of the two networks is combined to produce the result. In another work [380], only optical flow volumes are used for human activity recognition. Pose based CNNs (P-CNNs) are introduced in [381] which learns the motion and appearance data along trajectories. In a similar approach, two 3D convolutional networks with different scale are used for action recognition in [382] where the first network input is the depth image, and the second input is the specific region of interest provided by the Kinect skeleton tracker.

As mentioned before, it is shown that when the input is a spatiotemporal 3D volume, processing of temporal data using the spatial methods improves the result. But this is not a significant improvement, so many work use temporal recurrent models to also process the temporal dependencies in the data [372], [368], [383]. Among these methods, the use of the double-deep network [368] (named LRCN-Long-term Recurrent Convolutional Network) is a major contribution. LRCN networks are further used in many work. [384] use separate spatial and temporal convolutions in the input layer and also a bidirectional LSTM as a long-term temporal model. In [385] LRCN networks are used in conjunction with the focus selectivity idea [373]. During the action recognition process a soft attention mechanism [386] is used instead of a fixed focus on the center of the frame. In this way the model can learn the important region of the frame for the assigned task and weigh the extracted features from that area higher than other regions.

2.3.1.3 Pre-processing of human motion data

Dynamic changes in human body motion have a direct relation with finding the quality of human motion, so this research field is mostly focused on skeleton-based analysis of human motion. Acquisition and pre-processing of this skeleton data is one of the challenges in this field. Capturing skeleton data often requires a lab environment setting with different devices like optical motion capture, stereo cameras or depth sensors and this limits the availability of the data. Some widely used sensors like depth cameras facilitate providing the skeleton data, but suffer from issues like occlusion, sensing from a distance, and being impractical in outdoor situations. More diverse data can be achieved via RGB data using the recent deep learning techniques [17], [13], [8], [176]. The resulting skeleton data still can be noisy due to different issues like cluttered background or occlusion. Noise filtering solutions can involve traditional image filtering techniques such as Laplacian smoothing, Gaussian filtering, discrete cosine transformation (DCT), and discrete Fourier transformation (DFT) to preserve only low-frequency components of the joint trajectory components. Normalization of the different sizes of the subjects due to variation in height and distance from the camera, and alignment of the skeleton position in spatial (view-variance) and temporal (action start-end) space also is an important step in pre-processing of the motion data.

2.3.1.4 HME Specific Research

Human Motion Evaluation (HME) is performing analysis on the actions with the aim of finding how well the motions are performed by a subject. Action quality assessment in human motion has been studied in diverse research areas such as rehabilitation, sport movement analysis ad skill training. The evaluation criteria are usually problem dependent which limits the ability to compare the result of different work. In rehabilitation research, normal and abnormal movement models comparison is done to find anomalies in a specific movement. In the sports applications, the predicted score is compared with the ground truth scores provided by the experts in the field. For skill training methods, the expertise level is ranked in a few categories, for example expert, intermediate and novice.

2.3.1.4.1 Physical rehabilitation Assessment of functional mobility motions (gait on stairs, walking on a flat surface and the transition between sitting and standing) is done by Paiement et al. [387], [388] with comparing the statistical model of healthy subjects by new subject observation, frame by frame. Extracted features are joint positions, joint velocities, pairwise joint distances and pairwise joint angles. Redundancy of skeleton features is reduced using nonlinear manifold learning (diffusion map) which reduce the dimensionality of the mentioned low-level features. Hidden Markov Models (HMMs) are used for modelling the sequential motion data. In the frame-by-frame analysis, continuous HMMs performed better than discrete-state HMMs in detecting abnormalities. The motion sequences are recorded with an RGB-D Xmotion camera and a Kinect2 camera for Staircase, Walking and Sit-stand experiments. The area under the curve (AUC) of the Receiver Operating Characteristic (ROC) curve of sequence classification is used as a performance metric for motion abnormality detection. The best AUCs for walking, sitting and standing were 1.00, 0.99 and 1.00 respectively.

Elkholy et al. [389] worked on the same motions and trained a normalcy model that classifies a test sequence as being normal or abnormal based on its likelihood. To capture the abnormality, a number of medical-related features such as asymmetry, velocity magnitude, and centre of mass (COM) trajectory deformation are extracted. Two probabilistic models, Gaussian Mixture Model (GMM) and Kernel Density Estimation (KDE) were built from extracted features of normal sequences during training. The test sequences likelihood is computed by evaluating the trained models and based on the learned threshold a decision will be made about normality/abnormality of the motion. The best AUCs achieved by the KDE model, with 1.00 values for walking, sitting and standing and 0.98 for gaits on the stairs. The previous work's SPHERE data set as well as a new dataset (EJMQA) with 32 patients with abnormal gait and 11 healthy volunteers used for testing the proposed method. In addition to abnormality detection, the quality of action is also assessed for finding the degree of abnormality. A multiple linear regression predicts an assessment score on the scale of 1-5 (1 correspond to highest abnormality and 5 is no abnormality) based on the features as independent variables. Multiple linear regression has the best fitting result compared to quadratic regression, linear Support Vector Regression (SVR), Gaussian SVR, and squared exponential Gaussian Process

Regression (GPR) on the data. Vakanski et al. [390], [391] made a different dataset with Vicon optical motion capture system and Microsoft Kinect sensor, consisting 10 different physical therapy movements (Deep squat, Hurdle step, Inline lunge, Side lung, Sit to stand, Standing active straight leg raise, Standing shoulder abduction/extension, Standing shoulder internal-external rotation, Standing shoulder scaption). 10 healthy subjects repeated the exercise 10 times in both correct and incorrect manner to stimulate the patients that have musculoskeletal constraints. A deep autoencoder neural network reduces the dimensionality of the captured data. It is compared to linear techniques like PCA as it can produce a richer data representation. Then a parametric probabilistic model (Gaussian mixture model) was used for modelling the specific exercise. Model-based performance metrics evaluate the repetitions data with respect to this model and employs the log-likelihood for performance evaluation. Model-based methods can handle the stochastic variability of the motion data better than model-less techniques such as Euclidean distance, Mahalanobis distance or DTW Distance that are calculated directly from joint trajectories without modelling the movement. Instead of specifying only correct and incorrect classes, a scoring function maps the result of the performance metric of a movement quality score between 0 and 1. These scores are used in supervised learning by a neural network to regress the quality of the movement score from the test input motion. From the three deep learning architectures that were investigated (CNN, RNN and DNN), CNN had the best performance and acquired the minimum deviation between input and predicted scores in overall for ten different exercises. The deviation is between 0.013-0.041 for CNN, 0.016-0.093 and 0.030-0.192 for RNN and HNN.

2.3.1.4.2 Sport activity scoring To score the sports activities a regression problem is usually solved and the performance of the regression is found by computing the similarity between ground truth score and the predicted score. To measure the similarity the Spearman rank correlation coefficient or Pearson's correlation coefficient is employed. Much of the recent work in this field was tested on MIT Olympic Scoring dataset [392] (Diving and Skating) and UNLV AQA-7 dataset [393] (Diving and Vault). Both of these datasets are from the video of Olympics competitions on YouTube and there are variations in view of samples. MIT dataset is made of 309 diving and figure-skating videos with 60 FPS and 24 FPS frame rate respectively. The AQA (Action Quality Assessment) score is between 20-100 and 0-100 for figure skating. Seven action categories (single/synchronous diving, gymnastic vaulting, skiing, snowboarding and trampoline) with the total number of 1189 samples exist in the UNLV AQA-7 dataset Multiplication of execution and difficulty score is considered as ground truth. The performance of skeleton/kinematic data-based methods [392], [394] with the handcrafted approach were lower than deep learning methods [395], [393], [396], [397], [398], [399] on these datasets. The best correlation performance measures on the UNLV AQA-7 dataset are 0.84 and 0.7 for diving and vaulting while for the smaller MIT dataset the best correlations are 0.86 and 0.59 for diving and figure skating.

2.3.1.4.3 Skill training In the skill training research category, the performance of learners of the particular task like surgical skills or the daily activity of life is assessed. In the JIGSAWS dataset [400], [401] three levels of expertise (expert, intermediate and beginner) were classified in the different work. In the JIGSAWS dataset, eight persons are doing three surgical tasks (Suturing, Knot Tying, Needle Passing) with five repetitions using a da Vinci Surgical System. There are 103 Samples (Video and Kinematic data) with two left and right views and the same background. Evaluation criteria on this dataset is varied between classification accuracy, score prediction and rank accuracy (percentage of correctly ordered videos in ranking). Both deep learning [402], [403], [404], [405] and non-deep learning, skeleton/kinematic data-based methods [406], [407], [408], [409] were able to achieve high classification accuracy with this dataset.

2.3.2 Research Gap

The motion evaluation problem is highly dependent on the specific chosen application. We have focused on sport motions evaluation and more specifically martial arts because of the slower motions with mostly upright and normal body poses that can be less challenging for capturing with video cameras in video-based motion capture.

One of the limiting factors for motion evaluation methods is availability of annotated dataset with the expert judge scores on quality of motion. This ground truth is specially needed when using the supervised machine learning methods. For training the deep learning methods, large number of data is required. The mentioned datasets in the previous research for the sport activity scoring (see section 2.3.1.4.2) have between around 150 to 1100 video sequences. Since 3D data of the sport motion with the scoring ground truth information can be collected in limited quantity, increasing the accuracy of classic machine learning methods is valuable in this area.

In the previous work on martial art motion evaluation which is our chosen application, despite trying various combination of features and a range of machine learning methods, the achieved accuracy between the predicted and actual scores could be still improved. Using only classic machine learning, we can show that prediction accuracy can be significantly improved with the right selection of the features and the machine learning method. Furthermore, if the model is only trained on an specific movement, the best combination of feature type and machine learning method is not fixed and depends on the type of movement. The achieved correlation between the predicted score and ground truth is the highest among the state-of-the-art methods in sport activity scoringnd also better compared to previous methods that used the same martial arts dataset.

The output of the human pose estimation is the 3D SMPL model. There is little work about motion evaluation of the SMPL human models or their related datasets.

We have used a large dataset of SMPL motions (AMASS) with diverse motions and annotated it using pseudo-scoring computed by a program as ground truth. Using this large dataset, it is possible to demonstrate the potential of deep learning methods in motion evaluation of movement of SMPL human models.

2.4 Summary

In this chapter some background and previous research regarding to human motion estimation and analysis are discussed. This will give the reader a good overview of common methods used and a lot of insight in pursuing future research in this area. The identified research gaps are also highlighted and related work to the research gap and the suggested solutions are discussed. The research gap section is the basis of the research that is done in the next chapters and is related to the methodology and results presented in this thesis.

Chapter 3

Human Motion Estimation

3.1 Introduction

In this Chapter the methodology and the result of the research regarding to Human Pose and Motion Estimation is explained. The methodology is mainly focused on the methods designed for improving the accuracy of current state-of-the-art human pose estimation methods. The baseline method and quantitative evaluation metrics are introduced. The data that is recorded for testing our methods is an action based dataset focusing on occlusion. The data acquisition method and the training data and training procedure are explained as well as the experimental setup. Different validation and robustness tests are done regarding the introduced methods. The experiments and their results are presented and compared with state-of-the-art pose estimation methods including work focusing on occlusion of SMPL human model estimations. It can be shown that the introduced methodology can improve the results compared to the baseline and state-of-the-art methods.

3.2 Research Design

The input data in this project are derived from single view video footage. These data go through a pipeline where the human motion is estimated first, then the resulting motion is evaluated. The research objective is to improve the accuracy of motion estimation and therefore, modifications are done to improve the results compared to the baseline and other state-of-the-art methods. To resolve problems such as self-occlusion and unnatural poses in SMPL model estimation, extra processing stages are added to existing human pose estimators.

Better accuracy in video based markerless human motion capture, that estimates the SMPL parametric models, is the research objective in human motion estimation. As mentioned before, the choice of the SMPL model is because of its more realistic representation of human motion in 3D that includes both rotational and positional degrees of freedom.

The majority of the state-of-the-art research in human pose estimation report their performance based on testing motions in the available datasets and they usually lack information about the performance of the method on challenging motions.

In order to find the source of inaccuracies in the current state-of-the-art human pose estimators, we have tested a diverse range of online videos that contain challenging motions. Based on my observations, two areas of inaccuracies are chosen to further investigate: self-occlusion and unnatural poses.

Two possible approaches to solve this problem are either designing a new model to estimate the SMPL parameters from 2D input data or expanding the existing 3D SMPL estimation models to increase the accuracy. The difference between these two options lies in 2D and 3D data processing respectively.

2D human data is vastly available and its annotation can be easily done manually. In contrast, the large scale 3D annotated data, especially with SMPL model annotations, is not available. The key to advancement of SMPL-based human pose estimation models is in fact improvement in accuracy of the 2D pose estimation methods in finding the joints in an image/video frame caused by abundance of available 2D data. While this improvement leads to better results for 3D prediction, it also becomes the bottleneck of the 2D-to-3D algorithms since they are mostly reliant on 2D joint information in various stages. Therefore, any problems in the 2D human pose estimation will be directly transferred to the 3D prediction. This can be the result of incorrect 2D prediction, for example in a challenging pose, or correct 2D prediction, for example depth ambiguity in self occluded poses.

Because of the aforementioned limitations, processing the 3D data will be a better choice compared to 2D data processing. Every pose estimation model has its own inaccuracies and performs better or worse on different poses depending on the model design and the training data that is used. Therefore, post-processing of the output 3D data also makes it possible to adjust to different model-dependent inaccuracies in the 3D pose estimation output.

The proposed machine learning based post processing techniques, use two different type of models for handling the self-occlusion. The first one is a random forest that performs frame to frame pose mapping. The second one is a predictive autoencoder that maps sequences of poses and is therefore performing motion to motion mapping. The autoencoder model is called predictive because it maps a sequence of poses from current and past times to a sequence of poses of the future. Predictive setting is used because of the assumption that the motions before the self-occluded frames are available and it is possible to use this information to estimate the future occluded frames. In this project, self-occluded poses correspond to incorrect estimation of the pose not lack of information about the joint data. The predictive method can be especially effective when we don't have any information regarding the poses of current occluded frames. The autoencoders that are used have one and two layers of LSTM neural networks. The LSTM network unlike the traditional neural networks incorporates feedback connections allowing it to process the entire sequence of data instead of individual data points. LSTM networks are proven to be effective in predicting and understanding patterns when the data is sequential, for example in speech, text or other timeseries data like human motion.

For handling the unnatural poses in SMPL model estimation, we have realised that the 3D key point estimation is more accurate and robust in dealing with unnatural poses due to its non-restricted nature. It is possible to take advantage of this extra information and use them as a position targets for the IK algorithm to lead the hands and feet joints to a more correct position. Before being able to do that, the skeletons of SMPL and 3D key points should be unified with the same size and shape and aligned together. The whole process will be done automatically using the Python libraries of blender software.

3.2.1 Overview

After human pose estimation, post-processing will be done to improve the resulting motions. The post-processing method can be:

- data driven and use machine learning
- non-data driven and use Inverse Kinematics (IK)

Figure 3.1 shows the flowchart of the processing pipeline. In the first step the baseline HPE method is applied to the video to infer 3D key points, the SMPL human model and action. If the input video contain self-occlusion or unnatural poses, the resulting SMPL model from the first step can be improved in the second step which is post-processing. We have introduced machine learning based method for resolving self-occlusion and IK based methods for dealing with unnatural poses. The improved SMPL human motion can be used in the third step for motion evaluation. The third step will be discussed in the next chapter.

3.2.2 Data Acquisition

In the previous section it is mentioned that the 3D data (3D pose) processing is chosen as a approach for better human motion estimation. This means that the current 3D motion resulting from a baseline 3D pose estimator will be processed further to achieve better human motion estimation in terms of self-occlusion and unnatural (complex) poses. We will call this "post-processing" of the motion but it is not a typical post-procedure that uses optimization to gain a perfect result. The aim is using some external information to automatically correct the incorrect estimations made by the human pose estimator. If machine learning is to be used for such a corrective procedure, it would need data to be trained on. For supervised machine learning, it would need a pair of incorrect (wrong estimations due to self-occlusion or unnatural poses) and correct motions as input and ground truth respectively. Since such specific video data that is focused on self-occlusion is not available and the input motions that are bad estimations due to self-occlusion are dependent of the HPE method that is used, a new dataset should be created.

The raw data for the new dataset was collected in the motion capture lab that has eight Vicon optical motion capture cameras that are synchronised with a single video Vicon camera. A total of 30 simple exercise motions were designed to test the accuracy of the video based pose estimation methods. These actions include different scenarios of self-occlusion with body parts invisible to the single video camera or body parts in collision with one another. The subjects were asked to repeat the same action 5 times. Different viewpoints of the same movement are captured in some cases.

The data was collected in two rounds. In the first round the frame rate of both camera and motion capture was 120 fps. There were problems with dropped frames that resulted in un-synchronised input and ground truth as well as problems with the large size of video input data for each motion that made the dataset unreasonably large and difficult to process. Also the video and motion capture both had the same frame rate that made the synchronization issue more difficult to solve. In the second attempt, the video frame rate was reduced to normal 30 fps and motion capture recording stayed at 120 fps. The video frame resolution was 1280 by 720 pixels and the length of each action was between 10 and 40 seconds with the majority of actions taking between 20 and 40 seconds.

The raw MoCap data in the .mcp format and the video data in .mov format is collected by the Vicon Shogun software. Before data collection, the motion capture cameras and the subject are calibrated. For marker placement, the standard model with 53 markers without fingers is used. This made subject calibration step easier as the subject was recognised by the system easily. Although capturing with the normal clothes was tested successfully, it was decided eventually to use the MoCap suit for all subjects due to its reliability and ease of use. It is worth mentioning that the cameras should be given time to warm up before starting capturing to avoid the need to re-calibrate again in the subject calibration stage.

After capturing the data, the MoCap was converted to .c3d format using the Shogun Post software. The .mov videos were also converted to .mp4 format with the same resolution decreasing its size without reducing the quality. These raw data files are not the data that can be used for post-processing. The ground truth MoCap should be converted to an SMPL model (using Mosh++ Python code) and the video should be processed with HPE to be converted to the SMPL model. If all actions are recorded in one go, the actions should be separated using knowledge of exact start and end frames of each part so we can make correspondence between video parts and synchronised motion capture frames. This is because the current HPE method cannot accept long videos as input.

The resulting SMPL motions from the video and motion capture will have a difference in the number of frames that should be reasonably small. The chosen approach cuts the small number of extra frames from the longer motion sequence and assumes that they are almost synchronized. The coordinate systems and the root bone orientation of the video and MoCap SMPL models are also different. Therefore, the two models are rotated and adopt the same root bone orientation before any comparison or error computation between them can be made.

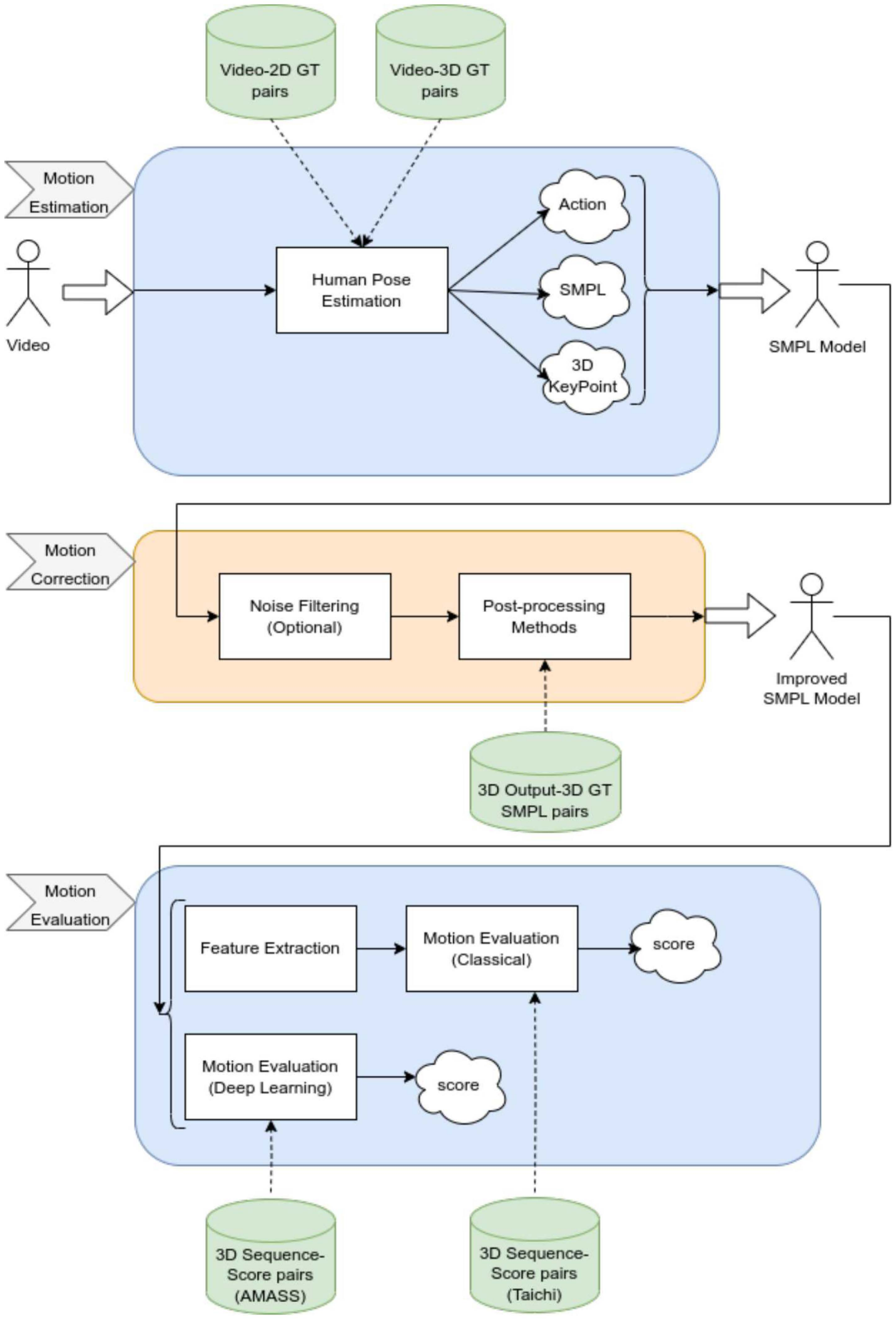


FIGURE 3.1: Human Motion Estimation and Evaluation Pipeline. The video is processed by the baseline HPE, result in SMPL model, 3D key points and Action type. Post-processing using IK or machine learning can improve the problems such as self-occlusion and unnatural poses. The improved motion from the SMPL model can be evaluated using classical machine learning or deep learning method

3.3 Methodology

Human Pose Estimation refers to extracting the 3D key-points or 3D parametric model such as SMPL from the input monocular video. Depending on the methodology and data that is used in designing the human pose estimator, each algorithm can have a advantages or drawbacks and there is no perfect human pose estimator. Some common drawbacks in the current human pose estimation methods are lack of sensitivity to occlusion and also unnatural motion.

In this work, we are aiming to improve the result of a human pose estimator (it can be any of the state-of-the-art methods) using post-processing of the result. We will try out methods based on machine learning (Random Forest and LSTM) and without machine learning (Inverse Kinematics and Kalman Filter) for this purpose.

3.3.1 Inverse Kinematics

In this section, the IK (Inverse Kinematics) method is briefly explained and then its application to the problem is explained. Before starting to work with the human data, we should choose a model to represent human motion. The human body is shown using a tree-like hierarchical structure with joints and links. In such model, each limb of the body is like a mechanical chain of different limbs. The joints movements in human body is only rotational and such joints are called revolute joints.

The hierarchy between limbs starts from the root joint, so the root joint is the local reference. The center of the hip is usually chosen as the root joint of the human skeleton or armature. The end-effector is the final position of the most outward link in each individual limb. A pose of an articulated body like a human skeleton is a set of joint articulations that result in specific positioning of the articulated body. Articulation is rotation or translation of a joint, for example in a planar arm with three links we can have 12, 6 and 3 degrees articulation that make up a pose.

FK (Forward Kinematics) takes the pose of an articulated body as an input and gives the position of the end effector as the output. IK in the reverse process, having the position of the end effector, it computes each articulation which is the pose of an articulated object.

The human pose skeleton as an articulated object has four end effectors at the end of each limb. In order to position the hands and feets in a more correct position, we would need the end effector position and orientation as the IK targets. Here we are using the result of 3D key points estimation as the limbs targets. During our testing on various challenging movement videos, we realized that there is strong evidence that the 3D body key points estimation is more powerful in finding the challenging poses that we are calling unnatural poses compared to the SMPL human body model estimation. For a few cases of self occlusion we also found that 3D key points might be more accurate in some self-occluded frames but this is not true for all videos. Generally for the whole sequence, the key points estimation is also affected by self-occlusion similar to the SMPL human body model estimation.

It can be shown that in extreme and unnatural poses, IK on a parametric SMPL model using estimated 3D key points can achieve lower position error for the limb end-effectors. Figure 3.2 demonstrates post-processing of SMPL human motion using IK and HPE estimated 3D key points.

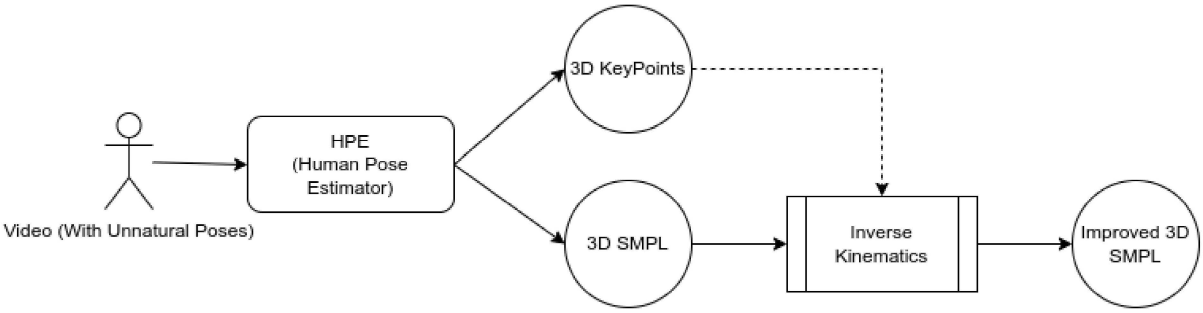


FIGURE 3.2: Human Pose Estimation Post-processing using Inverse Kinematics

3.3.1.1 Skeleton Alignment and IK

The aim in this part is to make use of deep learning results (3D key points) as target points for IK. More specifically, hand and feet 3D positions of the SMPL model are moved to more accurate target positions provided by 3D key points.

The HPE estimated 3D key point skeleton structure in Figure 3.3 is based on the Human3.6M benchmark dataset and is not the same as the SMPL skeleton. Therefore, a series of transformations is applied to match both skeletons. The procedure below illustrates this process. All the process is done automatically using python code and blender Python library.

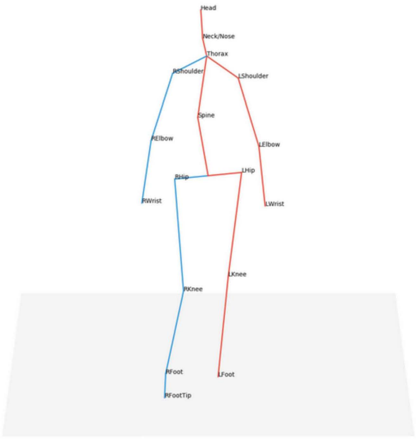


FIGURE 3.3: 3D key point skeleton structure (Human3.6M ground truth [62])

1. Importing 3D key points into Blender (see Figure 3.4)

2. Re-calculate the 3D key points skeleton to match the 3D SMPL model joint positions in terms of bone length and root position.

3. Rotate the 3D SMPL model root bone to the right orientation to match the 3D key point skeleton orientation

4. Use foot and hand key points as an IK target and move hands and feet to the new target position

In step 1, for importing the key points into Blender, a small cube for each joint in 3D is created - see Figure 3.4. In situations like unnatural poses or occlusion when the 3D estimated key points are more accurate than the estimated SMPL model parameters, IK improvement can be done.

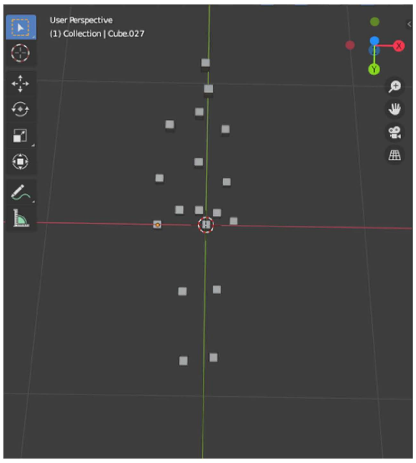


FIGURE 3.4: 3D key points position imported in the Blender 3D environment

In step 2, the 3D key point skeleton is automatically transformed to match the SMPL model's proportions and structure. This involves adjusting the key points to ensure that their bone lengths correspond to those in the SMPL model and that they are positioned relative to a common root joint. Here's how the process works:

- Root-Relative Positioning:

  First, each 3D key point is converted into a position relative to the root joint (typically the pelvis or hip). This centers the key point skeleton, allowing for easier alignment and scaling without altering the entire skeleton's position in 3D space.

- Bone Length Calculation and Adjustment:

  With the key points in a root-relative format, the bone lengths (the distance between connected joints) are calculated. These calculated bone lengths from the 3D key points are then compared to the corresponding SMPL model bone lengths. Since the SMPL skeleton may have different proportions, each bone vector in the key points skeleton is scaled to match the SMPL bone length for that joint pair. This reshapes the 3D key point skeleton to ensure its proportions align with the SMPL model while preserving the general body structure.

- Consideration for Structural Differences:

  This automated adjustment assumes that most joint pairs in the two skeletons correspond directly. However, for joints with notable structural differences (like hip joints), special handling may be needed to prevent distortions. This process preserves the 3D key point skeleton's orientation and positions while reshaping it to align with the SMPL model's proportions, setting up for smooth integration in later steps.

Figure 3.5 shows the result of reshaping the 3D key-points skeleton.

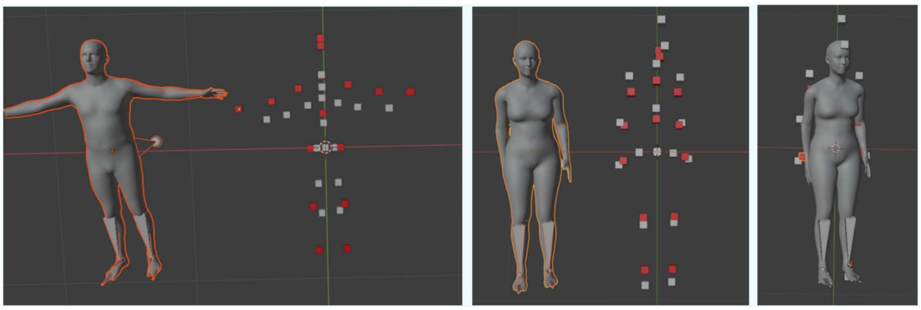


FIGURE 3.5: Reshaping and scaling the 3D key points skeleton to match the SMPL prediction (Gray: original key points. Red: transformed key points.)

In step 3 the SMPL human model is imported and rotated. In step 4, after finding potential target positions for IK in step 2, the distal joint of leg and arm bones is moved to the new targets.

In the program, we can control the number of bones in the IK chain. This number is set to two so that spinal bones are not affected. Figure 3.6 shows the difference between IK chains with unlimited and limited (2) length respectively.

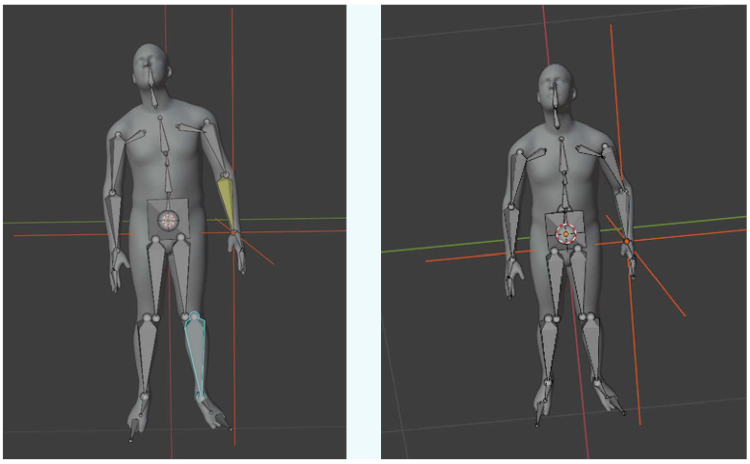


FIGURE 3.6: Left: IK chain of unlimited length causes an asymmetric spine. Right: IK chain of length two prevents unwanted spine movement

It should be noted that, considering model constraints, sometimes it is not possible to move both hand and feet joints simultaneously to the desired position.

Figure 3.8 and Figure 3.9 show handstand and kneeling as a unnatural pose respectively. Figure 3.7 shows hand behind back as a self-occluded motion.

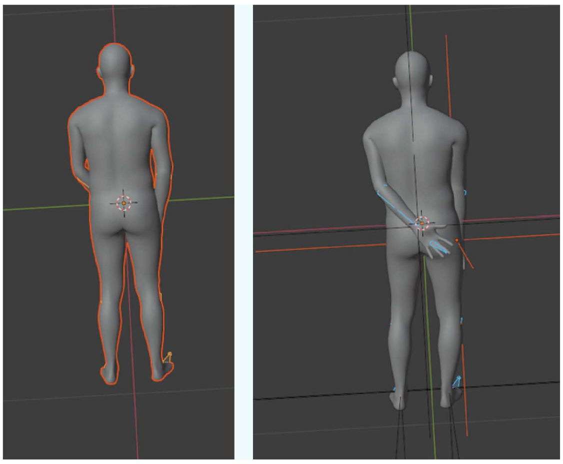


FIGURE 3.7: Hand behind back self-occluded pose improved by IK

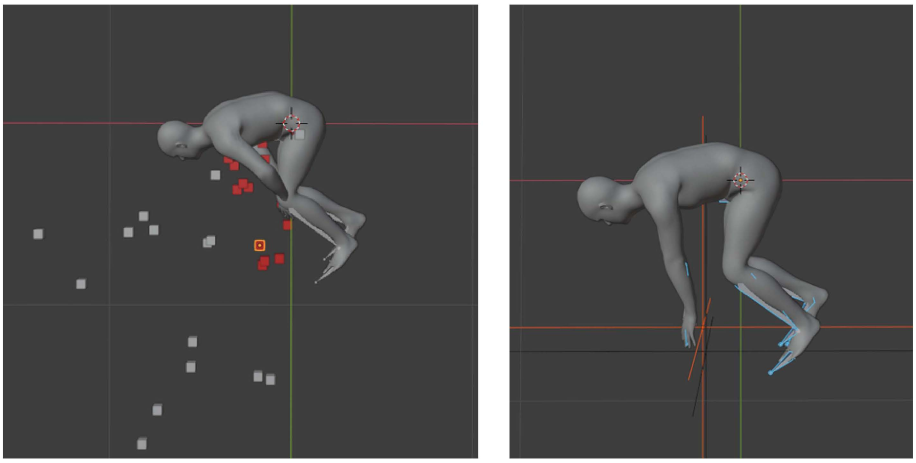


FIGURE 3.8: Handstand unnatural Pose improved by IK, Left: Scaling and reshaping the 3D key point skeleton, Right: Using hand and feet targets

3.3.2 Machine Learning

The main aim of section 3.3 is investigating the current gaps in the state-of-the-art human pose estimation frameworks and suggesting appropriate solutions. One of the main problems we are focusing on is the occlusion problem more specifically self-occlusion. For this purpose, one of the solutions is adding another stage of machine learning for correcting the incorrect poses. The mentioned model should learn how to assign the correct pose or sequence of poses if given an incorrect pose or motion sequence - see Figure 3.10.

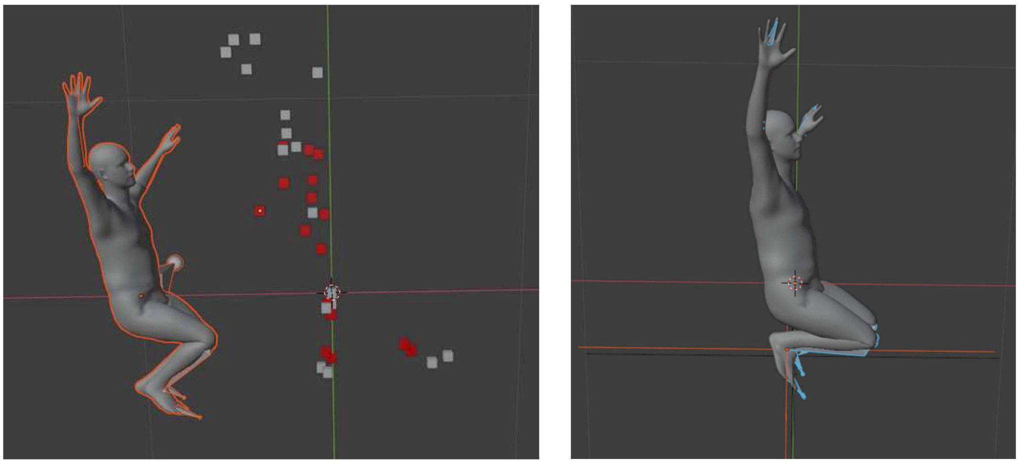


FIGURE 3.9: Kneeling unnatural Poses improved by IK, Left: Scaling and reshaping the 3D key point skeleton, Right: Using feet targets

3.3.2.1 Frame Based Method

In this section, a frame to frame correspondence between incorrect and correct SMPL pose is established. We need to use a multi-input multi-output model such as random forest. The model is trained using the incorrect and correct pairs and the resulting human motion is evaluated. Assuming the most part of the motion is invisible from the camera point of view, we should have extra knowledge about the action that is performed to be able to reconstruct the invisible motion correctly. This is done by first recognizing the action and also using the model trained for correcting that action to reconstruct the human motion. We have also tried training the model using pose parameters on all the actions in the dateset at once and it showed lowering the pose parameters error that it is trained on.

The input/output data would be a one dimensional array of pose parameters that are joint rotations and the SMPL model joints. This rotation data is in the form of Euler angles but can be converted to other rotation representations such as Quaternions.

3.3.2.2 Sequence Based Method

Human motion is a multi-dimensional signal. One dimension of such signal is time and the other dimensions are the joint's orientations. Such multidimensional time series data can be viewed as a 2D matrix data. The goal is to map the incorrect 2D matrix to the correct 2D matrix.

3.3.2.2.1 Data Preparation Supposing we have a motion sequence of length n which is down-sampled from 120 frames per second to 30 frames per second, a 124-frame window from this sequence is chosen which is equal to around 4 seconds of video. This is enough time for completing one action. We are choosing overlapping windows from the sequence one each every second (30 frames).

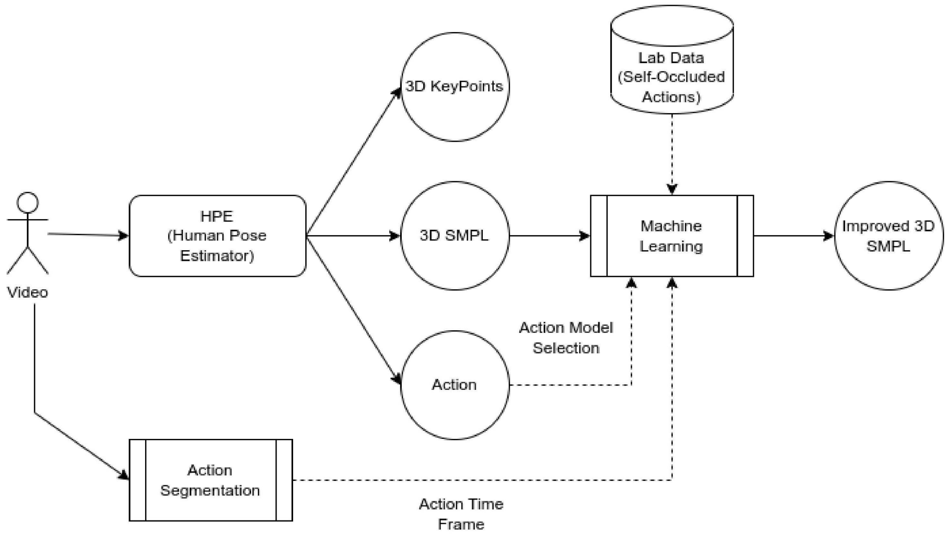


FIGURE 3.10: Human pose estimation post-processing using machine learning. The baseline HPE method is predicting 3D key points, 3D SMPL and the Action. The action-specific machine learning models are trained on the self-occluded videos. Specific model is chosen based on the recognized action. If the input video contains different actions, action segmentation is used to find the start and end frames of each action

Our motion input signal is a $T \times 72$ matrix, in which T is the total number of frames in the video and 72 is representing three rotational values for each 24 joints. It is divided into overlapping windows of size 124×72 one each every 30 frames. After dividing our data into overlapping matrices of fixed length (124), we got 10773 data matrices. To use all the data frames of the video, down-sampling was done with 4 different starting points.

The values of each matrix data point are root relative rotations of joints. The root joint of the 3D SMPL model from the MoCap data is moving in the 3D space while it is fixed in the SMPL model derived from the video. MoCap data also has more joints including hand and face key points which are filtered out to match the input data from the video.

We are using a recurrent neural network for the prediction of the motion. A window from the past is mapped to a window from the future. Since the data from the occluded parts is not available or is incorrect, the idea is that by using the previous frames, we should be able to predict the future frames. We are using a smaller window compared to the previous convolutional network. We can increase the window size to assess the effect.

A time series is a sequence of data points that occur in successive order over some period of time. When we have only one time-dependent variable, it is called a univariate time series. In the case of human motion, each joint position or rotation variable is a time series. The movement of joints is interdependent. This means that human motion can be expressed as a multivariate time series. In this case, the goal of using LSTM is that we use the value of variables in a window of time from the past, to predict the values for a window of time in the future. This is helpful and better than the previous pose correction machine learning method in two ways. Firstly, instead of learning to map only one pose (incorrect pose output of pose estimation) to the correct pose (ground truth MoCap) at any given frame, we are mapping an entire incorrect pose motion sequence to the correct pose motion (the duration of motion in frames in the length of the window). Secondly instead of mapping only corresponding frames, we are mapping the past to the future. This is especially helpful for occlusion, because based on the information of the motion from the past we can estimate where the self-occluded parts of the body "will" go in the future.

Two different networks with one and two LSTM layers are used for this purpose - see Figure 3.11 and Figure 3.12. The LSTM network can be structured into an Encoder-Decoder LSTM architecture, which enables the model to handle variable-length input sequences and generate variable-length output sequences. In this setup, an encoder LSTM model processes the input sequence step-by-step. Upon reading the entire input sequence, the hidden state or output of the model encapsulates an internal representation of the sequence as a fixed-length vector. This vector is then passed to the decoder model, which uses it to generate each step of the output sequence. In an Autoencoder LSTM, an encoder-decoder LSTM is designed to read, encode, decode, and reconstruct the input sequences from a given dataset. The model's performance is assessed by its ability to accurately recreate the input sequences. Once the model achieves satisfactory performance in sequence reconstruction, the decoder can be removed, leaving only the encoder model. This encoder model can then be used to convert input sequences into fixed-length vectors.

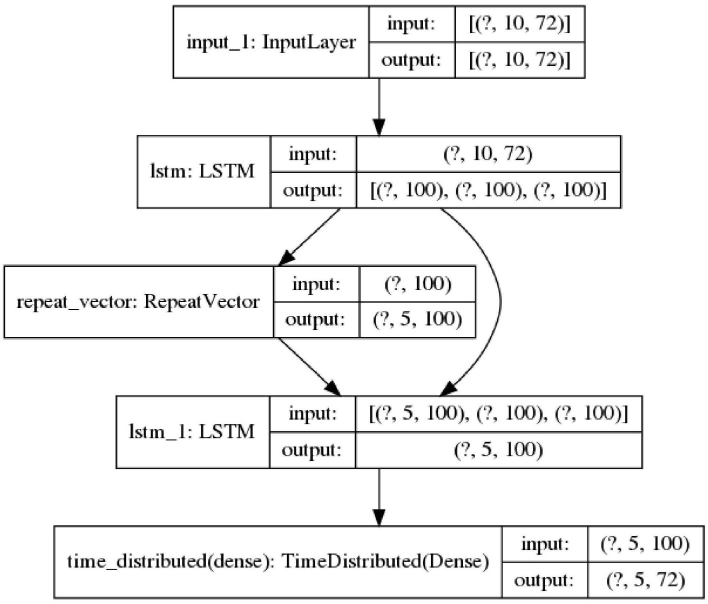


FIGURE 3.11: AutoEncoder with one layer of LSTM, Input sequence of length 10 and output sequence of length 5. The brackets shows the dimensions of input and output data of each layer. The first dimension (n) is the batch size.

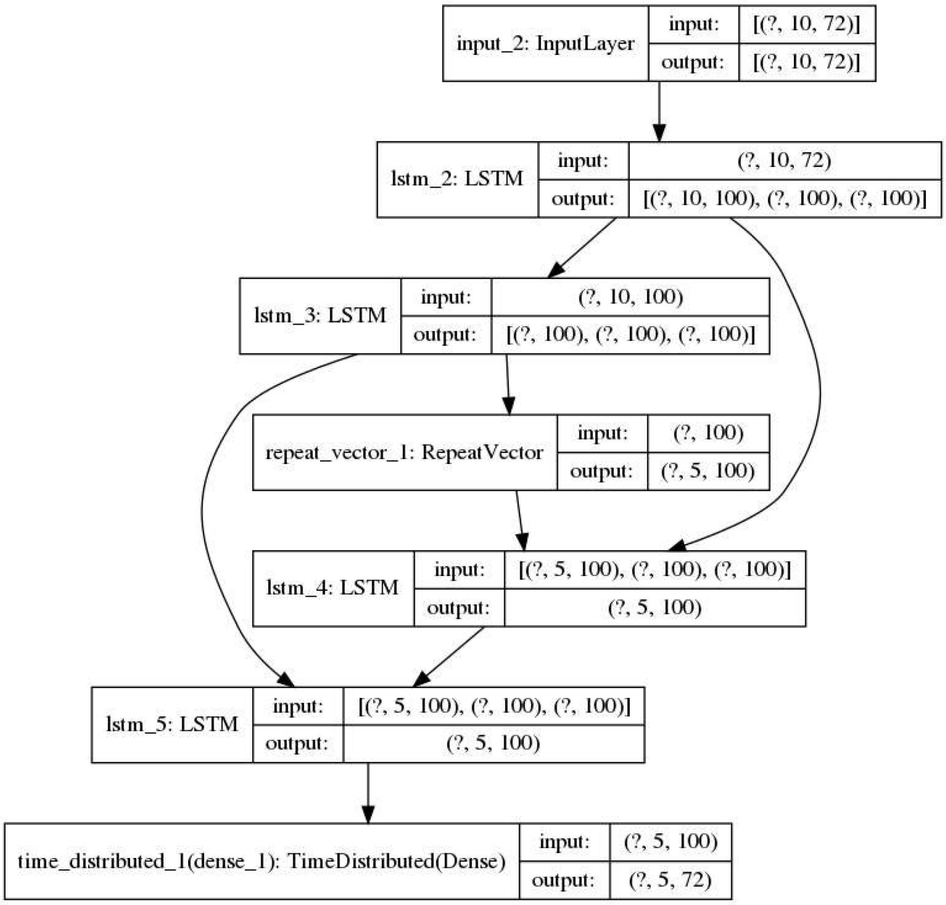


FIGURE 3.12: AutoEncoder with two layer of LSTM, Input sequence of length 10 and output sequence of length 5. The brackets shows the dimensions of input and output data of each layer. The first dimension (n) is the batch size.

The training data is first scaled between -1 and +1 and test and validation data are scaled accordingly. Then the data of 120 frames per second is divided into 4 sequences of similar motion with 30 frames per second. The goal is to use all the data we have and not discard any part with down-sampling. Then we split the time series into a set of overlapping windows with increments of one. Each pair of windows is a member of the training, test or evaluation data. The length of the window from the past is 10 frames, which is mapped to a window of the future with a length of 5 frames.

As it can be seen in the Figures 3.11 and 3.12, the output of the network is the predicted window of future motion with a fixed length of 5. Since output windows are overlapping, for each frame of the initial sequence we will have more than one predicted value from different overlapping predicted windows. The average of predicted values for each frame is computed to find the final output motion signal.

To properly compensate for the self-occluded motions, the uniqueness of such mapping can be ensured when having some information about the action that is taking place. A model that is trained to map incorrect motion affected by occlusion, to correct motion can better predict future motion when it is trained on actions of that category. Therefore, an additional step is needed to do action segmentation and action recognition of an arbitrary sequence of different actions and use the related model for predicting the output.

3.4 Experimental Design

In this chapter the experimental design and the results regarding human pose and motion estimation are demonstrated. Improvements are proposed for the state-of-the-art methods to increase the accuracy. These improvements aim to reducing the error of human pose estimation caused by self-occlusion and also unnatural poses.

It is worth mentioning in human pose estimation research, there are quantitative and qualitative comparisons between the state-of-the-art and the implemented method. Qualitative comparison utilises visual demonstration of the improved motions. Since the quatitative errors are average measurements during the whole motion sequence, qualitative comparison is also needed to show how a method can improve the overall configuration of the predicted pose in specific occurrences such as occlusion and unnatural poses.

The output of the human pose estimation is an SMPL model with position and orientation of the joints. The measured error in human pose estimation is the average orientation and position error of the joints between the ground truth and video pose estimation. The output of the human motion evauation is the predicted score or the evaluation metric. The error of score prediction is measured by computing root mean square of the error (RMSE) between ground truth and predicted score. These errors are the basis of the comparison between the different methods.

Figure 3.1 shows the pipeline of all experiments across human motion estimation and evaluation. The human pose estimation can be any of the current methods that result in human joint positions and orientations. It can be shown with extra stage (it is called "post-processing" in this thesis), common problems in current pose estimation methods with unnatural poses and self-occlusion can be improved.

Finding occluded movements is made possible using knowledge about the performed action (from human pose estimation part). To experiment this, a dataset containing various self-occluded scenarios and unnatural poses are recorded using a synchronized monocular video and optical motion capture.

The experiments in this research are designed according to the objectives the research is aiming to achieve. The designed methods aim to decrease the error between the ground truth and prediction in human pose/motion estimation. The motion estimation improvement experiments are done to address self-occluded poses using machine learning and unnatural poses using IK. Machine learning experiments are done in two ways. The first is training the models on all videos of the dataset with different actions. In the second experiment this is changed by training on videos of a specific action. This is done to improve the mapping between incorrect and correct poses or motions. The input video can be of any motion therefore the extra information from action recognition is used to first find out the motion that is performed and then use the model trained on the specific action to recover the incorrect motion. The IK experiment is also taking advantage of the extra information about 3D key points to guide the SMPL joints that are incorrectly estimated due to unnatural poses to a better position.

3.4.1 Datasets Details

The human pose estimators usually are trained on a set of benchmark datasets. Generally, the datasets may vary depending on the recording modalities but they mostly consist of synchronized multi-camera videos and motion capture. The motion capture data that is used for ground truth usually is converted to the human model in the form of skeleton or SMPL model which consists of human pose and shape parameters.

There are two parts in the overall motion estimation pipeline that uses datasets for training. The first part is the main pose estimation algorithm which uses benchmark datasets for training models to predict the 3D key points and SMPL model. These are 3D datasets such as Human3.6M [62], 3DPW [112] and AMASS [106] and 2D datasets with in-the-wild RGB video such as PoseTrack [410] and InstaVariety [411]. The second part is focused on solving the problem of self-occlusion. A different dataset is captured in the lab with self-occluded motions to train the second part. For unnatural poses problem the videos from the EMDB dataset [412] are used.

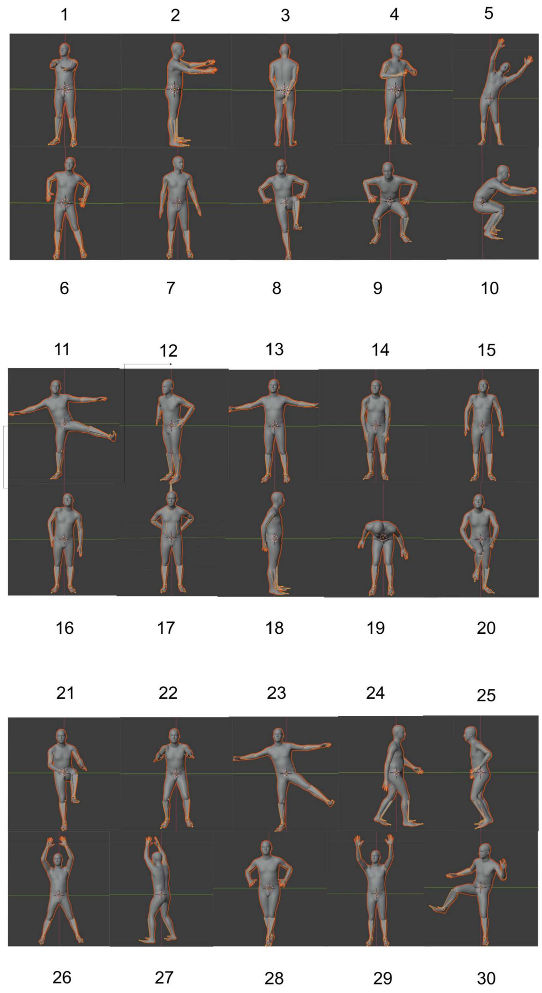
The data we recorded in the lab consist of synchronized motion capture and monocular videos similar to the existing dataset. The ground truth is converted to SMPL format. The list of 30 different exercise and range-of-motion actions that were done by different subjects were as follows [413]

- Action 1: Lift arms to T position – move arms forward with cross-over – Front
- Action 2: Lift arms to T position – move arms forward with cross-over – Lateral
- Action 3: Arms behind back (not touching) face backwards (alternating leading L-R)
- Action 4: Upper Body rotation (twist) arms inwards, fists touching
- Action 5: Upper Body tilt left and right arms up
- Action 6: Hip rotations, CW and CCW
- Action 7: Head rotations L-R/nod/tilt (in all three planes), arms down
- Action 8: Knees up (L/R), arms akimbo
- Action 9: Squats front – arms akimbo: Heels down and Heels elevated

- Action 10: Squats lateral – arms front stretched: Heels down and Heels elevated
- Action 11: Lateral leg lift (L/R) foot pointing forward, arms lateral
- Action 12: Turn CW and CCW in steps of 45 degrees, arms to the side and arms akimbo
- Action 13: T position – rotate arms hands facing back, down, forwards, up
- Action 14: Shoulders forward and backward (popping)
- Action 15: Shoulders up and down
- Action 16: Shoulders alternating
- Action 17: Arms behind back (not touching) Front
- Action 18: Arms behind back Lateral
- Action 19: Forwards bend to horizontal (straight back), arms down (next to body)
- Action 20: Knees up (L/R), grab knee, arms akimbo
- Action 21: Knees up (L/R), turn out, arms lateral
- Action 22: Squats front – arms forward: Heels down and Heels elevated
- Action 23: Lateral leg lift (L/R) foot pointing forward then point upwards, arms lateral
- Action 24: Walking (L to R and back)
- Action 25: Jumping, arms akimbo (L to R and back)
- Action 26: Star jumps (facing front)
- Action 27: Star jumps turn 45 degrees, CW and CCW
- Action 28: Star jumps with crossed legs (facing front)
- Action 29: Swinging straight arms, forwards and backwards
- Action 30: All body shake, i.e. arms, legs, torso, head.

Each individual action performed by a subject is cropped from the recorded video and motion capture sequences. Therefore, each video in the dataset contains a subject doing only one motion. From these data fields around 20 percent of the data is selected for test and validation and the remaining will be the training data. Assignment of data as part of the train, test or validation set is done randomly. The error related to the test data video is demonstrated in Table 3.2. The test data videos that are chosen randomly are:

FIGURE 3.13: Actions in the dataset recorded in the lab



- Subject 1:
  - 1) Rotating wrists
  - 2) Knees up (L/R), grab knee, arms lateral
  - 3) Squats front – arms front stretched
  - 4) Jumping, arms akimbo (L to R and back)
  - 5) Star jumps (facing front)
  - 6) All body shake, i.e. arms, legs, torso, head
- Subject 3:
  - 7) Arms behind back (not touching) face backwards (alternating leading L-R)
  - 8) Knees up (L/R), arms akimbo
  - 9) Lateral leg lift (L/R) foot pointing forward, arms lateral
  - 10) Arms behind back Lateral

- Subject 5:
    - 11) Squats front – arms akimbo
    - 12) Lateral leg lift (L/R) foot pointing forward, arms lateral
    - 13) Rotating wrists
    - 14) Arms behind back Lateral
    - 15) Knees up (L/R), grab knee, arms lateral
    - 16) Star jumps (facing front)
    - 17) crossed legs (facing front) – arms akimbo

The dataset was collected in two phases. In the first phase, motion data from six subjects was captured. In the second phase, data from twelve subjects was recorded. Each subject performed a total of 30 actions, with five repetitions per action, and each action averaged 30 seconds in duration. In the first phase, one combined video and motion capture sequence was recorded for each subject. In the second phase, each action was recorded separately. For the first dataset, we needed to separate each action from the continuous video in a synchronized manner, which could be challenging—particularly if synchronization issues arose, such as dropped frames in the motion capture data. Figure 3.13 illustrates each action included in the dataset.

The random selection of the data presents one of the possible scenarios of selecting data. Later, cross validation is done instead of random selection of the data and the model performance is reported (see Section 3.6.3). When training on a specific action, the actions in the dataset will be limited to only one action and therefore the data in this action specific dataset will be different by the subject that performed it. The positional error reported in the results is with assumption of standard subject's height of around 170 cm in Blender. In reality, height of each subject is different and for more accurate results the position errors should be scaled based on the subject's real height.

3.4.2 Training Procedure

The two models that were used for machine learning based post-processing are frame-to-frame random forest and motion-to-motion LSTM AutoEncoders. Random forest model have different parameters that can be set. To find the best parameters for the model, an iterative random search is done on the parameter grid that is initialized beforehand. Each model in the iteration is evaluated by the mean squared error of the validation data and the best model is chosen for the optimal parameters.

The AutoEncoder models have one layer or two layers of LSTM network in in encoder and decoder parts. Adam optimization method is used for training. The Adam optimization is a stochastic gradient descent method that is based on adaptive estimation of first-order and second-order moments. According to[414], the method is "computationally efficient, has little memory requirement, invariant to diagonal rescaling of gradients, and is well suited for problems that are large in terms of data/parameters". The Huber loss function is used for training. The Huber loss function is used in robust regression, that is less sensitive to outliers in data than the squared error loss.

We have also performed cross-validation training with 10 folds on both random forest and LSTM AutoEncoder models. In this case both models used the MSE (Mean Squared Error) as a loss function. The selected model of each fold is evaluated by computing the mean squared error loss and r2-score.

3.4.3 Experimental Setup

The Vicon software Shogun Live and Shogun Post software are used for capturing the data for self-occluded motion estimation and converting it to the correct format. The motion capture data is processed by the Mosh++ software to be converted to SMPL human model. The video data should be also converted to SMPL format. This is done by state-of-the-art pose estimation methods. The main pipeline implementations used Python and related machine learning libraries such as Sci-kit Learn and Tensorflow/Keras. Automatic error measurement is made possible by exporting the result and ground truth SMPL models to the 3D environment and measuring the poses frame by frame using Blender Python and related mathematical libraries.

3.4.4 Baseline Methods

In the human motion motion estimation part, state-of-the-art methods are checked against the problems such as self-occlusion and unnatural poses which is common with different algorithms. Based on the existing online ranking of the current pose estimation methods, the best performing ones which have higher probability of handling difficult poses are chosen for baseline and also comparison. We have used the MotionBERT [415] method as a baseline which provides SMPL output, 3D key points output and action as output. For comparison with the state-of-the-art, another good performing model HybrIK [247] and the only occlusion based method for SMPL prediction PARE [211] are chosen.

3.4.5 Evaluation Metrics

In the human motion estimation part, the goal is obtaining the most accurate motion which means the pose in each frame of the motion should be accurate. This accuracy is measured by comparing the errors between the joints of the ground truth and the result. The SMPL joints have rotation and position characteristics that can be measured. For the joint position error we compute the L2 norm of the difference between two pose vectors (vector of all rotation values). For the joint rotation error, first the rotation values are converted from axis-angle to quartenion then the difference between two quartenions are computed which is an angle in radians.

The accuracy measure (joints positional and rotational error) is compared to the baseline method in Figure 3.14 and numerical comparison with state-of-the-art and baseline are provided in Tables 3.2 and 3.3. The errors are the average value across all joints and frames of the motion sequence. When more than one action video is evaluated, the mean and standard deviation across all actions are provided. To cover all scenarios of data partitioning, 10-fold cross validation is done and the loss and accuracy of each proposed machine learning method prediction is reported.

The mean joint position error, denoted as *j3d*, is calculated as the L2 norm (Euclidean distance) between the predicted and ground truth positions of each joint, averaged across all frames and joints. Let:

- $\mathbf{p}_{i,t}^{\text{pred}}$: Predicted 3D position of joint i at frame t,
- $\mathbf{p}_{i,t}^{\text{gt}}$: Ground truth 3D position of joint i at frame t,
- N: Total number of joints,
- T: Total number of frames.

The formula for *j3d* is:

$$\text{j3d} = \frac{1}{T}\sum_{t=1}^{T}\left(\frac{1}{N}\sum_{i=1}^{N}\|\mathbf{p}_{i,t}^{\text{pred}} - \mathbf{p}_{i,t}^{\text{gt}}\|_2\right)$$

where $\|\mathbf{p}_{i,t}^{\text{pred}} - \mathbf{p}_{i,t}^{\text{gt}}\|_2$ represents the Euclidean distance between the predicted and ground truth positions for each joint i at each frame t.

The mean joint rotation error, denoted as *rot*, is calculated by measuring the shortest angular distance between the predicted and ground truth quaternions for each joint and averaging across all frames and joints. Let:

- $\mathbf{q}_{i,t}^{\text{pred}}$: Predicted quaternion for joint i at frame t,
- $\mathbf{q}_{i,t}^{\text{gt}}$: Ground truth quaternion for joint i at frame t,
- N: Total number of joints,
- T: Total number of frames.

To compute the angle $\theta_{i,t}$ between the predicted and ground truth quaternions, we use:

$$\theta_{i,t} = 2\arccos\left(\left|\langle \mathbf{q}_{i,t}^{\text{pred}}, \mathbf{q}_{i,t}^{\text{gt}}\rangle\right|\right)$$

where $\langle \mathbf{q}_{i,t}^{\text{pred}}, \mathbf{q}_{i,t}^{\text{gt}}\rangle$ is the dot product of the two quaternions, and the absolute value ensures that the angle remains within 0 and π.

The formula for *rot* is then:

$$\text{rot} = \frac{1}{T}\sum_{t=1}^{T}\left(\frac{1}{N}\sum_{i=1}^{N}\theta_{i,t}\right)$$

This gives the mean rotational error in radians, averaged over all joints and frames.

The HPE predicted model parameters are joint rotations, therefore the time-series input data that is processed by machine learning are rotation values. Hence, the model is optimized to better predict rotation values (improve the pose parameters of the model) that results in a more similar pose to the ground truth.

The rotational error of joints is independent from the global position of the character and shows improvement in overall pose of the SMPL method. When the position of joints is dramatically incorrect for example in unnatural poses, the joint position error can show better positioning of the selected joints. To understand which parts of the body have more errors, we have also measured right leg and left leg, right arm and left arm errors separately. These parts are the limbs are the body that are moving and cause self-occlusion.

3.5 Results

3.5.1 Inverse Kinematics

In this section, post-processing of predicted SMPL the predicted model is done by taking advantage of the availability of predicted 3D key points. With examining the output of the human pose estimation algorithms on various video inputs, we can realize that in challenging scenarios especially unnatural poses that are not the everyday activity, 3D key-points can be used to improve the SMPL model prediction.

For an occlusion scenario (unlike unnatural poses) the improvement from applying the IK method is marginal, as the 3D key point prediction will be also affected by occlusion and might not always be able to provide better joint positions to guide the limbs to the correct position. The dataset that was recorded in the lab was mainly natural poses with self-occlusion. Videos of unnatural poses with ground truth are needed to demonstrate the reduced error after using IK in these scenarios. For this purpose, handstand and cartwheel videos that are part of the recent EMDB dataset [412] are used to show the IK performance. It can be shown that as expected, IK using the predicted 3D key-points can decrease the joint position errors in the SMPL model.

It is also worth mentioning that 3D key points can only specify the position targets for the body end effectors (wrist and foot joints) and as IK is a position based method, information about target orientation is not available in IK. This limitation of application of IK with only position targets can also be seen in the result of orientation errors.

The results presented in Table 3.1 highlight the effectiveness of inverse kinematics (IK) post-processing when applied to predicted SMPL models, especially in unnatural poses like handstands and cartwheels. The table shows both joint position and rotation errors, with comparisons between the baseline (HPE without IK) and three different IK configurations: targeting both hands and feet, hands only, and feet only.

3.5.1.1 Analysis of Position Error

In the position error results, we observe a reduction in overall joint position errors when applying IK to both hands and feet, or to hands alone. Specifically:

- The **overall joint position error** decreases from 110.60 (Baseline) to 104.58 with IK on both hands and feet, and to 102.67 with IK applied to hands alone. This reduction demonstrates that incorporating 3D key-point predictions for hand targets significantly enhances the spatial accuracy of joint predictions.

- In contrast, applying IK only to the feet results in minimal changes to the joint position error (112.50), indicating that feet targets alone are less effective in guiding the model's joint positions accurately for these complex poses.

The **arm-specific position errors** further confirm these findings, with a notable reduction in errors for both the left and right arms when IK targets include the hands. For instance, the left arm error drops from 26.14 (Baseline) to 21.46 when IK is applied to the hands. This improvement underscores the utility of IK in aligning hand and arm positions more accurately during challenging poses, where limb occlusions and unusual body orientations make it difficult for the model to maintain correct joint positions.

3.5.1.2 Analysis of Rotation Error

The rotation error data presents an interesting insight into the limitations of IK based solely on position targets:

- Overall rotation errors increase when IK is applied to hands, feet, or both. For example, the baseline rotation error of 317.65 rises to 373.95 when using IK for both hands and feet, and to 371.56 when targeting only the hands.

- This increase suggests that while position-based IK improves spatial accuracy for hand and feet placements, it does not provide the necessary orientation data to maintain accurate joint rotations, leading to higher overall rotation errors.

3.5.1.3 Interpretation and Support for IK Contribution

These results from Table 3.1 support the contribution of using IK in post-processing for unnatural poses by demonstrating that:

1. **IK improves positional accuracy** for joints, particularly for hand and arm positions, where hand targets significantly lower position errors.

2. **IK alone is insufficient for orientation accuracy** due to its position-based nature. Without orientation targets, joint rotations may deviate, especially in complex poses where limbs are constrained by the SMPL model's kinematic structure.

In summary, this analysis of Table 3.1 validates the benefit of using IK to refine joint positions in predicted SMPL models for unnatural poses but also emphasizes the limitations of IK when orientation accuracy is essential. Future improvements could include methods that integrate both position and orientation data, potentially enhancing the overall accuracy in complex motion scenarios.

TABLE 3.1: Inverse Kinematics, Unnatural poses (Handstand and Cartwheel)

	Joint	**Baseline**	**Post-Processing**		
	Error	**HPE**	***IK Hand and Feet***	***IK Hand***	***IK Feet***
Position	All	110.60 +- 18.92	104.58 +- 11.07	102.67 +- 13.15	112.50 +- 0.02
	LArm	26.14 +- 4.38	21.46 +- 0.89	21.46 +- 0.89	26.14 +- 4.38
	RArm	24.56 +- 6.04	21.31 +- 3.75	21.31 +- 3.75	24.56 +- 6.04
	LLeg	0.08 +- 0.04	0.07 +- 0.04	0.07 +- 0.04	0.08 +- 0.04
	RLeg	14.81 +- 0.16	16.46 +- 0.91	14.81 +- 0.16	16.46 +- 0.91
Rotation	All	317.65 +- 5.37	373.95 +- 13.66	371.56 +- 14.48	320.04 +- 4.55
	LArm	54.42 +- 2.92	83.07 +- 2.89	83.07 +- 2.89	54.42 +- 2.92
	RArm	57.80 +- 2.61	83.07 +- 6.48	83.07 +- 6.48	57.80 +- 2.61
	LLeg	56.05 +- 5.82	56.38 +- 7.74	56.05 +- 5.82	56.38 +- 7.74
	RLeg	56.98 +- 4.79	59.05 +- 2.05	56.98 +- 4.79	59.05 +- 2.05

Since moving the joints together is affected by the constraints in the human model, three different experiments were done. The feet joints and the hand joints are moved separately and at the same time. The results on the cartwheel and handstand videos shows that using the feet targets alone was not sufficiently effective in improving the overall positional error. In contrast, simultaneous hand and feet targets or hand targets alone will lead to better overall joint position estimations.

3.5.2 Machine Learning

In this section, machine learning based post-processing is done on the imperfect result of human pose estimation to reduce the effect of self-occlusion. As it can be seen in Figure 3.10 the SMPL model, 3D key points and the action is predicted using the baseline HPE method. When the input video contains a self-occluded action, an imperfect SMPL output from the HPE can be improved by the machine learning based post-processing method. Performed action found in the video can be used to select the model trained on an specific action. To separate each action from the video, action segmentation is done. An action based dataset with different subjects doing the same action is needed to train the post-processing methods. An example of such dataset is collected in the lab with exercise motions as different actions.

In order to show the effect of the machine learning method on prediction of self occlusion videos, we have to choose a challenging action. In this respect, we have chosen the hand behind back motion. With the help of action recognition we can choose the model trained on the recognized action and predict the occluded motion.

3.5.2.1 Frame based method with classic machine learning

The first method is using a random forest model to learn the corresponding correct pose to each incorrect predicted pose from human pose estimation. Each pose is a set of joint orientations. The random forest method is chosen due to being a multi-input and multi-output model suitable for this purpose. This is a frame based method, because each index of data belongs to only one frame.

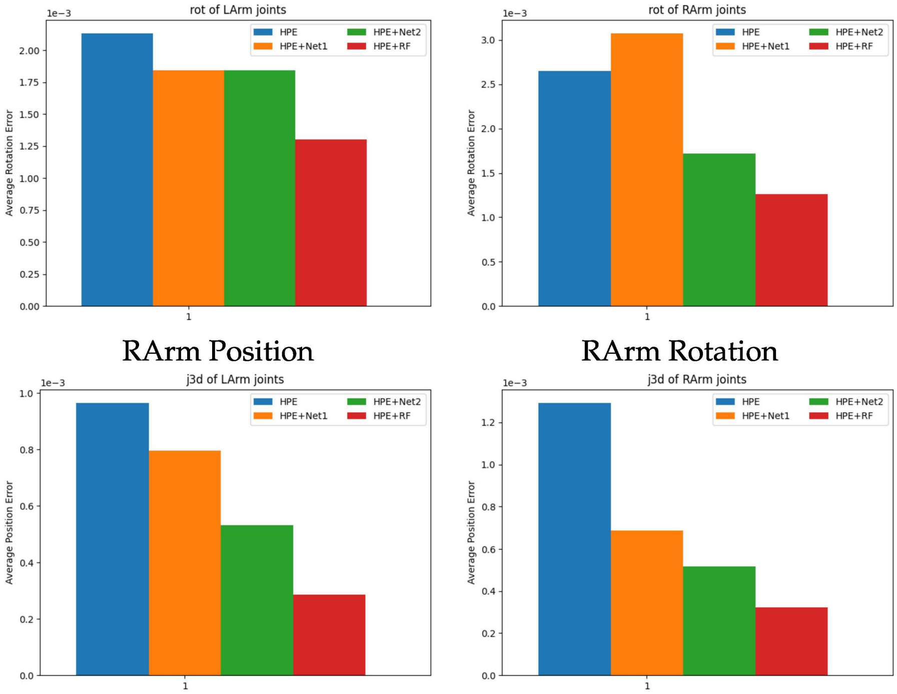
The result of the predictive sequence based method is compared with the baseline HPE and frame based method in the Tables 3.2 and 3.3. The frame based method have access to the current incorrect motion to predict the current correct motion while the predictive methods use motion information from the past frames to predict the current correct motion. The frame based method with RF outperform predictive sequence based methods AE1 and AE2 (Auto Encoders with 1 layer and 2 layers of LSTM) when trained on all actions and specific action. When training on all actions, the frame based RF method is the only method among the post-processsing methods that improves both rotation and position error of the joints compared to the baseline HPE.

The results shown for all action training, are not representative of the final result and only intermediate experiment. In the next experiment, training on specific action is performed to make the machine learning able to not only reduce the error but reconstruct the motions in the output correctly. This means in order to reconstruct the self-occluded motion, with an input sequence of arbitrary motions, first the action performed by the subject is identified. Then, the model that is trained on actions of that category (action-specific model) is chosen for post-processing. In order to know which parts of the video needs to be post-processed by this action specific model, action segmentation is done to find the start and end frames of the performed action. The result of action specific training which is our selected method is shown in the next section (see Table 3.3) in order to compare all the machine learning based post-processing methods. It should be noted that there is an error with the training data ground truth legs captured with MoCap. When converting the motion capture to the SMPL model, this error causes unusual bending of knees in the training data while the subject standing straight (see Figure 3.16). The high legs error in all action specific post-processing methods is the mentioned problem with the data.

3.5.2.2 Predictive sequence based method with deep learning

The second method is using a sequence of motion as the input data. It is also making correspondence between a window of motion from the present to a window of

FIGURE 3.14: Arm joints (LArm=Left Arm, RArm=Right Arm) error for action specific learning, Action: Hand Behind Back



RArm Position RArm Rotation

LArm Position LARm Rotation

motion from future frames. This is useful when the information about the present self-occluded or other wrong poses are not available and we are trying to guess or predict the future motion only based on the previous motions.

Table 3.2 shows the difference between the baseline, introduced post-processing methods and state-of-the-art PARE [211] and HybrIK [247], when training the models on all actions of the test data. It can be shown that the pose parameter of the SMPL which is joint rotations is improved in all the suggested methods. The simultaneous frame to frame method which is based on random forest performs better compared to the predictive methods that only have access to the previous motions. The autoencoder with two layers of LSTM performed better than the autoencoder with one layer of LSTM. The performance of all introduced methods are better than other state-of-the-art methods. In terms of joint position accuracy, unlike the RF (simultaneous frame to frame) method, predictive methods AE1 and AE2 do not provide joint position improvement compared to the baseline HPE method.

Table 3.3 shows the result of action specific training on one of the actions compared to the state-of-the-art methods. Here, a problem arose when using only one subject data for training, one subject data for test and another subject for validation. This caused a large error for the leg-based motions due to being biased towards the training subject's posture which has bended knees while standing (see Figure 3.3). Ignoring the source of bias, the self-occluded motions are reconstructed correctly and this is shown in qualitative results by the improvement of occluded arms motion that was incorrectly estimated by the baseline. A snapshot of the qualitative

TABLE 3.2: Frame Based and Sequence Based Machine Learning (Lab Data)

	Joint	Baseline	Post-Processing			SOTA	
	Error	HPE	*RF*	*AE1*	*AE2*	*PARE*	*HBIK*
Rotation	All	467.10 +- 91.02	186.33 +- 72.19	262.82 +- 88.35	258.13 +- 81.18	480.87 +- 92.84	535.53 +- 94.98
	LArm	64.04 +- 11.62	37.98 +- 17.35	51.18 +- 15.78	47.95 +- 14.67	63.60 +- 13.66	83.94 +- 21.05
	RArm	67.99 +- 15.19	40.03 +- 19.06	53.18 +- 16.94	50.26 +- 15.71	76.52 +- 15.26	83.19 +- 19.11
	LLeg	52.436 +- 23.78	27.73 +- 15.62	48.57 +- 19.98	47.35 +- 17.50	59.64 +- 24.11	62.65 +- 25.36
	RLeg	53.08 +- 28.61	27.98 +- 14.78	46.29 +- 21.12	47.40 +- 19.21	57.85 +- 25.30	69.13 +- 27.76
Position	All	74.90 +- 39.56	57.70 +- 25.60	88.60 +- 32.60	90.72 +- 35.11	88.91 +- 41.78	70.65 +- 34.81
	LArm	15.57 +- 5.76	2.91 +- 7.07	20.20 +- 7.68	21.06 +- 8.73	19.31 +- 8.89	15.23 +- 4.91
	RArm	14.02 +- 4.46	13.73 +- 7.49	18.45 +- 6.67	18.45 +- 6.65	17.43 +- 9.00	13.45 +- 4.76
	LLeg	12.67 +- 8.71	8.77 +- 5.15	15.25 +- 5.64	15.18 +- 6.39	14.70 +- 9.31	11.79 +- 8.35
	RLeg	15.58 +- 11.58	9.27 +- 5.14	15.83 +- 8.48	16.89 +- 8.56	16.89 +- 10.62	13.15 +- 10.26

results can be seen in Figure 3.15. In Table 3.3 and looking on the self-occluded left arm and right arm motions, both the random forest and the autoencoder with two layer of LSTM show improvement in both position and rotation of the joints and all of the introduced methods perform better than the baseline and the selected state-of-the-art methods when trained on specific action.

TABLE 3.3: Frame Based and Sequence Based Machine Learning (Action 3 (hand behind back) of Lab Data)

	Joint	Baseline	Post-Processing			SOTA	
	Error	HPE	*RF*	*AE1*	*AE2*	*PARE*	*HBIK*
Rotation	All	16.83	18.81	14.82	12.70	20.27	23.47
	LArm	2.13	1.30	1.84	1.84	3.17	4.95
	RArm	2.65	1.26	3.08	1.71	4.05	4.58
	LLeg	0.97	3.24	3.01	2.75	1.37	1.14
	RLeg	0.95	3.72	3.30	3.07	1.20	1.27
Position	All	2.90	2.59	3.42	3.02	2.89	3.09
	LArm	0.96	0.29	0.80	0.53	0.87	0.78
	RArm	1.29	0.32	0.69	0.52	1.08	0.98
	LLeg	0.13	0.55	0.55	0.60	0.18	0.27
	RLeg	0.25	0.77	0.80	1.01	0.13	0.43

In the next experiment, we have tested a range of other types of motions with action specific training. While action specific training is effective in predicting the occluded motions it cannot fix all other problems that are caused by incorrect prediction. If the training is person specific it also has the problem of getting biased towards the posture of the training subject which in our case happened to be an unusual ground truth posture compared to the test and validation data that belong to the other subjects.

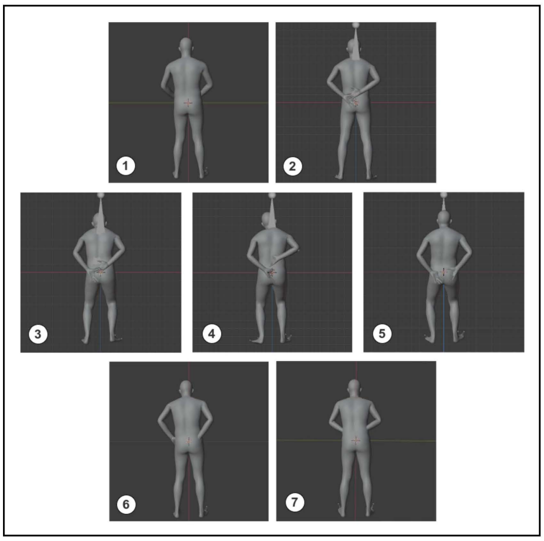
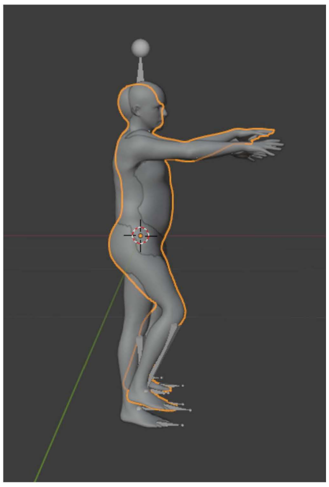


FIGURE 3.15: Self-Occlusion human pose recovery using post-processing methods, compared to the state of the art and the baseline HPE. 1: HPE, 2: Ground-Truth MoCap, 3: RF Post-Processing, 4: Predictive AutoEncoder Post-Processing with 1-Layered LSTM, 5: Predictive AutoEncoder Post-Processing with 2-layered LSTM, 6: PARE Method, 7: HybrIK Method.

FIGURE 3.16: ground truth motion capture knee problem in training data causes high leg error in action specific post-processing



3.6 Validation and Robustness

3.6.1 HPE Training on Lab Data

In order to show the efficiency of the post-processing in human motion estimation compared to training the baseline with the lab data, we have trained the original pose estimation network with the default benchmark dataset (3DPW) and the occlusion dataset (LAB data) and compared the error of the result. The result shows that training the baseline HPE method with the lab data offers slight improvement in some cases and no improvement in the other. The amount of improvement is less significant comparing the post-processing methods. This can also be confirmed with the qualitative results that shows the HPE model trained on lab data cannot reconstruct the self-occluded motions.

TABLE 3.4: Lab Data Training without Post-Processing

	Joint	**Baseline**	
	Error	**HPE+Benchmark Data Training**	***HPE+Lab Data Training***
Rotation	All	510.54 +- 86.11	534.88 +- 86.07
	LArm	85.65 +- 15.56	81.66 +- 21.48
	RArm	91.92 +- 18.73	97.07 +- 19.49
	LLeg	58.25 +- 24.57	56.40 +- 24.92
	RLeg	60.25 +- 29.31	58.68 +- 28.34
Position	All	119.46 +- 43.73	111.65 +- 38.40
	LArm	27.39 +- 8.03	29.29 +- 8.21
	RArm	29.12 +- 8.37	28.79 +- 8.96
	LLeg	17.77 +- 10.12	16.30 +- 9.88
	RLeg	19.54 +- 12.14	18.00 +- 10.91

3.6.2 Unseen Dataset Input

The post-processing networks are trained on our proprietary data which contains specific actions including self-occlusion. To check the performance on other datasets, the EMDB dataset [412] is chosen as the new source of input data. There are diverse activities in the dataset, performed outside the lab environment. The models that are trained on all actions of the lab data which are different from the EMDB dataset actions, are used in this experiment. The result of testing the sequence-based machine learning method on unseen "in-the-wild" dataset are shown in Tables 3.5 and 3.6. Data in the wild are naturalistic datasets and can be used to test how the method performs on real-world scenarios.

The result shows that position and orientation of the limbs do not improve when using another type of dataset with totally different actions as input. The average overall pose parameters error using post-processing trained on lab data showed improvement but this was not true for the leg and arm limbs estimation error that are usually the source of self-occluded motions.

3.6.3 Cross-Validation

The train, validation and test videos of the dataset are chosen randomly from all the data. To cover all possible choice of train and test data, we have done 10-fold cross validation on both action specific training and all actions training experiments. Tables 3.7 and 3.8 show the mean and standard deviation of accuracy and loss of each fold. The first three columns are related to training on all actions while the last three columns are related to training on a specific self-occluded action (hand behind back). The frame-to-frame RF method shows the best accuracy and loss value. The second

TABLE 3.5: Sequence Based Machine Learning (Unseen in-the-wild Dataset, 1-layer Network)

	Joint Error	**Baseline**	**Post-Processing**
		HPE + EMDB	***AE1 + EMDB***
Rotation	All	405.00 +- 34.36	343.08 +- 24.71
	LArm	55.07 +- 9.94	57.51 +- 0.01
	RArm	54.16 +- 10.041	84.23 +- 9.85
	LLeg	49.61 +- 9.02	54.08 +- 5.44
	RLeg	47.75 +- 12.21	54.70 +- 7.61
Position	All	77.32 +- 26.73	107.99 +- 16.06
	LArm	16.27 +- 6.57	23.34 +- 5.30
	RArm	15.32 +- 6.55	22.05 +- 4.80
	LLeg	12.89 +- 5.09	18.24 +- 2.64
	RLeg	12.79 +- 5.63	19.96 +- 3.50

TABLE 3.6: Sequence Based Machine Learning (Unseen in-the-wild Dataset, 2-layer network)

	Joint Error	**Baseline**	**Post-Processing**
		HPE+EMDB	***AE2+EMDB***
Rotation	All	405.00 +- 34.36	360.17 +- 22.54
	LArm	55.07 +- 9.94	58.79 +- 12.80
	RArm	54.16 +- 10.04	86.08 +- 9.61
	LLeg	49.61 +- 9.02	57.56 +- 5.37
	RLeg	47.75 +- 12.21	56.73 +- 6.11
Position	All	77.32 +- 26.73	113.45 +- 15.31
	LArm	16.27 +- 6.578	23.91 +- 5.67
	RArm	15.32 +- 6.55	22.32 +- 5.86
	LLeg	12.89 +- 5.09	20.02 +- 2.32
	RLeg	12.79 +- 5.63	21.47 +- 2.91

best model is the predictive autoencoder with two LSTM layers. The predictive autoencoder with one LSTM layer has the lowest performance. The standard deviation between the fold accuracy and loss is small, suggesting that model's performance is relatively consistent across different folds.

TABLE 3.7: Loss Value for 10-fold cross validation

Loss \ Model	RF All Act.	AE1 All Act.	AE2 All Act.	RF Act. 3	AE1 Act. 3	AE2 Act. 3
cross val Loss Mean	0.252	1.179	0.573	0.145	0.342	0.189
cross val Loss SD	7.58E-3	1.03E-2	2.95E-3	3.01E-2	9.72E-3	6.62E-3

TABLE 3.8: Accuracy Value for 10-fold cross validation

Accuracy \ Model	RF All Act.	AE1 All Act.	AE2 All Act.	RF Act. 3	AE1 Act. 3	AE2 Act. 3
cross val Accuracy Mean	99.447	95.960	97.299	99.459	94.220	95.877
cross val Accuracy SD	0.013	0.137	0.095	0.072	0.544	0.335

3.7 Limitations

In the previous sections it is shown how the motions affected by self-occluded and unnatural poses can be improved using post-processing on 3D motion. The limitations regarding the motion estimation is that in the data driven solutions based on machine learning, the solution will decrease the error on the type actions previously seen in the training data. In action specific training, the solution is effective for self-occluded movement recovery but for other type of inaccuracies in the output more investigation should be done to find the reason for lower effectiveness of the data driven method. In action-specific training, the number of subjects in the dataset should be increased otherwise the training will be biased to only one person posture or individual style of movements.

3.8 Summary

In this chapter, we presented two main post-processing methods to improve the accuracy of human motion estimation models: **Inverse Kinematics (IK)** and **Machine Learning-based Post-Processing**. Both methods were applied to the predicted outputs of state-of-the-art human pose estimation (HPE) models, particularly focusing on the **SMPL model**, a widely used human body model. These methods were specifically targeted at enhancing joint positions and rotations in challenging scenarios, such as self-occlusion and unnatural poses, where traditional HPE models struggle.

3.8.1 Contributions to Human Motion Estimation Post-Processing

3.8.1.1 Machine Learning-based Post-Processing

The first post-processing approach explored in this chapter was machine learning-based refinement using two distinct models: a *Frame-to-Frame Random Forest* and *Motion-to-Motion LSTM AutoEncoders*. Both models aimed to correct errors in joint positions and rotations based on temporal dependencies in motion sequences. This approach was particularly beneficial for self-occlusion scenarios, where parts of the body are hidden and pose estimation algorithms typically struggle.

- **Training and Evaluation**: Both models were trained using a cross-validation approach with 10 folds, where the Mean Squared Error (MSE) was employed as the loss function. The performance was evaluated using the **joint positional and rotational errors**. The machine learning models were able to learn the temporal patterns and improve the accuracy of joint predictions across a variety of poses. The Random Forest model, optimized through iterative random search, performed well in terms of predicting frame-to-frame pose changes. On the other hand, the LSTM AutoEncoder models, with one or two LSTM layers, demonstrated the ability to capture motion dynamics over time, resulting in lower joint position and rotation errors.
- **Validation**: These models were validated on diverse datasets, including videos with self-occlusion and other complex motions. The machine learning-based methods consistently outperformed baseline models (such as MotionBERT, HybrIK, and PARE) in terms of joint position errors, particularly when applied to natural motion datasets.
- **Best Performing Method**: The LSTM AutoEncoder model, in particular, showed the most significant result, reducing both positional and rotational errors compared to the baseline methods. This demonstrated the power of temporal modeling in improving the accuracy of human pose predictions. The overall error of the RF method was lower because of having access to the current frame information.

3.8.1.2 Inverse Kinematics (IK) Post-Processing

The second approach explored was the application of **Inverse Kinematics (IK)** to refine predicted 3D joint positions, leveraging **predicted 3D key points** from the HPE models. IK was applied specifically to **unnatural poses** like handstands and cartwheels, which are not typically encountered in natural motion datasets and are known for causing difficulties in pose estimation.

- **IK Methodology**: IK was applied to adjust the positions of key body joints, particularly the hands and feet, using predicted 3D key points as positional targets. The IK method was used to correct joint position errors by applying **position-based optimization**, targeting the wrist and foot joints to improve pose accuracy.

- **Validation and Results**: The IK method was validated using the **EMDB dataset**, which includes extreme poses like handstands and cartwheels. The IK method demonstrated a clear reduction in **joint position errors** when applied to both the hands and feet. For example, the overall joint position error decreased from **110.60** (baseline) to **102.67** when IK was applied to hands alone, and further improved to **104.58** when applied to both hands and feet. However, while IK improved position accuracy, it resulted in an increase in **rotation errors**, highlighting its limitation in handling joint orientations without additional orientation targets.

- **IK Performance**: IK showed the most significant improvement for arm and leg joints, particularly in reducing errors for the left and right arms. However, the **rotation errors** increased because IK is position-based and does not account for joint orientations, suggesting the need for methods that incorporate both position and orientation data to fully optimize joint accuracy in complex poses.

3.8.2 Comparison and Conclusion

Machine Learning-based Post-Processing (using Random Forest and LSTM AutoEncoders) and **Inverse Kinematics (IK)** were both shown to be effective in improving the performance of human motion estimation. However, each method had distinct advantages and limitations:

- **Machine Learning-based Post-Processing** was more effective in handling **self-occlusion** and improving both **joint positions** and **rotations** in complex motions. The **LSTM AutoEncoder** model, in particular, outperformed other baseline methods (MotionBERT, HybrIK, and PARE) in terms of accuracy and generalization, especially when the model was trained with temporal motion data.

- **Inverse Kinematics** was particularly useful for refining **joint positions** in **unnatural poses** (e.g., handstands, cartwheels), where traditional HPE models struggled. While IK significantly improved **positional accuracy**, it was limited by its inability to refine joint **rotations**, leading to higher rotation errors when applied in isolation.

3.8.3 Summary of Key Findings

- **Machine Learning-based post-processing** methods, especially the LSTM AutoEncoder, provided significant improvements in both positional and rotational accuracy across a variety of motion sequences, making it the better choice for general-purpose human motion estimation.

- **Inverse Kinematics** demonstrated substantial benefits in **unnatural poses**, especially when 3D key points for hands and feet were used as positional targets. However, IK's inability to handle joint rotations properly made it less effective for improving overall pose quality in some cases.
- Future improvements could integrate both **position and orientation** data in the post-processing stage to further reduce errors in both joint positions and rotations, especially in complex motion scenarios.

Ultimately, the results confirm that combining both post-processing methods, tailored to specific challenges like self-occlusion and unnatural poses, offers the most promising approach for enhancing the accuracy of human motion estimation in diverse real-world applications.

Chapter 4

Human Motion Evaluation

4.1 Introduction

In this Chapter the methodology and the result of the research regarding to human motion evaluation is explained. The methodology has two parts: classic machine learning and deep learning, each of these using different datasets. In classic machine learning, the baseline martial arts dataset and feature design are used to improve the human motion evaluation result by adding new classification models and feature selection. In the deep learning part, a larger dataset is annotated and used to compare the deep learning method with the classic machine learning models. The evaluation metrics and experimental setup as well as the details of the datasets that are used are discussed. The results are presented as the comparison of the ground truth motion evaluation score and the score value predicted by the models. The classic machine learning method shown significantly better prediction with combination of the selected feature type and machine learning model. Improving classic machine learning methods is useful when limited number of data with annotated ground truth scores are available. Action based modeling for evaluation of an specific motion type also shows improved performance compared to all action training. For large datasets containing diverse type of motions, and without feature extraction, deep learning improves performance compared to classic machine learning.

4.2 Research Design

In this chapter the methodology and the results of the research regarding the human motion evaluation is explained. The input is the estimated 3D human pose/motion that can be derived from the input video. The research objective is improving the accuracy of the motion evaluation.

Modifications are done to the pipeline to improve the results compared to the baseline and other state-of-the-art methods. In the human motion evaluation part, which is a supervised learning, the feature design and datasets from the recent research is used as a baseline. Modification is done to the types of machine learning methods, the processing pipeline and feature selection to improve the results.

Assuming we have the 3D motion data available for motion evaluation purposes, this part of the project is concerned with more accurate evaluation of the motion. Since this is a supervised learning problem that needs a motion dataset that is already annotated by experts, an existing dataset[416] and standard pipeline of motion analysis is used. The previous work on this dataset, worked on combining the different features and applying PCA on the result, also trying different machine learning methods with different combination of features. From observation of the results, it was obvious that combining more features did not act in favour of more accurate prediction of the motion evaluation metrics. Therefore, the introduced features are implemented and tried separately in combination with two new machine learning methods. These shown that separate features have the potential of better motion evaluation when combined with a compatible machine learning method. The potential of deep learning for motion evaluation and its better performance compared to classic machine learning is also demonstrated by automatic labelling of a large human motion dataset of SMPL models. This also showed that the output of the first stage which is 3D SMPL model can be processed similar to 3D key point skeleton data.

4.2.1 Overview

After human pose estimation and post-processing to improve the human motion in challenging scenarios such as unnatural poses and self-occlusion, the second part of the pipeline is focusing on human motion evaluation. Since this part is mainly application driven rather than abstract work, the purpose of this analysis should be specified. Here, scoring martial art motions is chosen as the application.

An available dataset of martial art motions with the annotated scores [416] is used for comparing methods with classic machine learning. Different simple and complex motion features are implemented:

- Kinematic simple features derived from the joint positions
- Quaternion and Euler simple features based on the joint rotations
- Ergonomic complex features related to martial art movements
- Muller complex features designed based on relation of body part with respect to each other

Compared to the previous work on the same dataset, two new machine learning models random forest and ridge regression are used.

Since the previous dataset is not SMPL based, an existing dataset of SMPL motion [106] is annotated to compare the functionality of a deep learning (CNN) based method with random forest and regression. In this part joint rotations in axis-angle format without any feature extractions are used as an input to the both classic machine learning and deep learning methods.

4.2.2 Data Acquisition

As mentioned before, the already existing datasets with 3D joint positions and annotated scores of martial art motions [416] and another un-annotated dataset with SMPL pose (joint rotations) [106] are used.

Although eventually online datasets are used for this part, we have also created SMPL ground truth for existing online multi-camera martial arts datasets as a preliminary research. Before this, another research is done in the lab using multiple cameras to calibrate and synchronize the cameras and eventually use 3D reconstruction to compute the ground truth. Using multiple cameras as the source of ground truth have the advantage of removing the need to synchronise between video and other modalities or sensors that create ground truth for the video input. Multiple cameras are not used further in this research as synchronized motion capture and video used for creating dataset.

4.3 Methodology

This section explains evaluation of 3D human motion. Its aim is assigning a score, or more generally a motion quality metric number, to an entire motion sequence.

4.3.1 Classic Machine Learning

To evaluate a sequence of human motion in the input data, a series of operations should be taken which can be summarized as a pipeline. The motion sequence input can be in the form of image data (RGB videos) or skeleton data (3D human pose sequence). In the first case we should convert the RGB data to a 3D pose (skeleton) sequence consisting of 3D coordinates of the joints before continuing the process. Figure 4.2 shows the pipeline when having skeleton data and Figure 4.1 shows the similar modified routine when having video data.

Figure 4.3 shows more details about the feature processing step in the pipeline. The extracted features can be in different types but generally all of them are in the form of time signals. Assume for each motion sequence in the dataset, there are F feature signals. Only two statistics from each signal e.g. mean and standard deviation are extracted and used as a representation of that feature signal. The statistics from all features related to a motion sequence form a group of feature numbers. Then principal component analysis (PCA) is used to reduce this set. Finally, the classifier predicts the evaluation score of the person that is doing a particular motion.

4.3.1.1 Motion feature extraction

Motion capture or MoCap systems provide the possibility of recording the human motion with a range of speed and accuracy depending on the technology that is used. Generating MoCap data can be done with different systems like mechanical, optical, video/image camera or magnetic. The resulting data can be in various

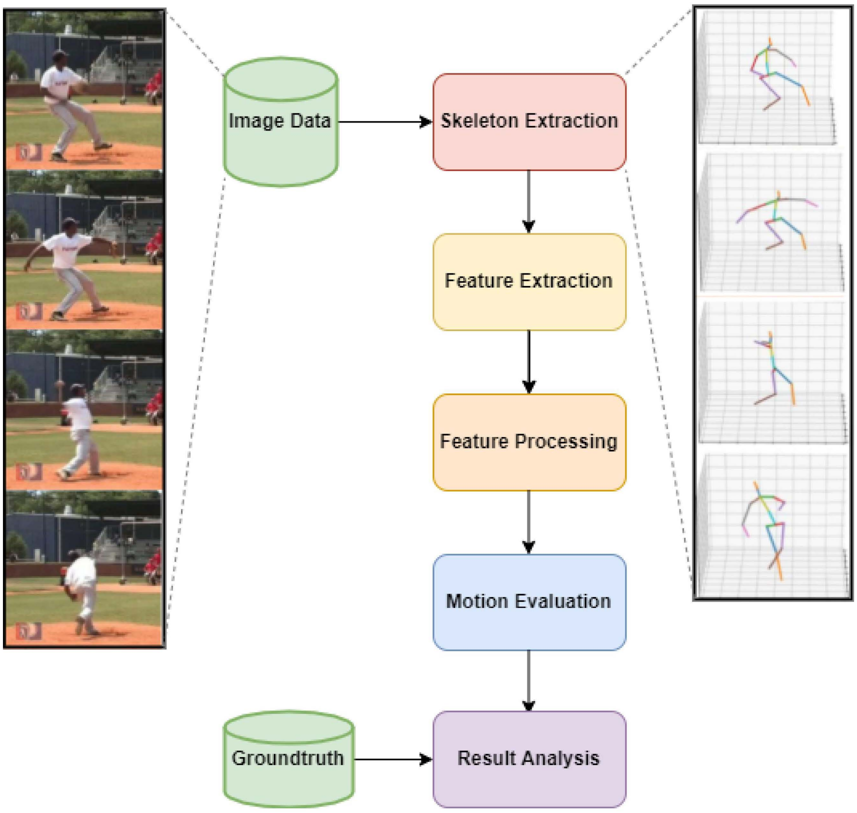


FIGURE 4.1: Motion Evaluation from RGB Input (video)

formats but we assume that the motion data is modelled using a kinematic chain which consists of joints and bones. Motion analysis techniques are using different kind of features that can be quantitative/numerical or qualitative/relational ways to describe the motion. Numerical features are more sensitive to details and observed changes in the pose while relational features provide an opportunity to compare the motions in a more semantic way.

4.3.1.1.1 Position and Orientation Three dimensional coordinates of the kinematic skeleton, joints or other landmarks provide the main information of human motion. In order to have a full description of the human skeleton, the orientation of the bones should be known. Orientation can be represented in different forms. In Euler representation, rotation is described by successive rotations around three axes. One of the disadvantages of this representation is that several numbers can describe the same rotation. Quaternion representation composed of a scalar value and 3D imaginary values has the advantage of being unique for each specific rotation.

4.3.1.1.2 Kinematic/Kinetic Features Geometrical aspects of the motion are described by kinematics related features such as velocity, acceleration and jerk which are derived by position and orientation. Kinetics is the relationship between the motion of the object and its mass and forces applied to it. Centre Of Mass (COM) of the body is used to compute the body kinetic energy. We can find body COM from a

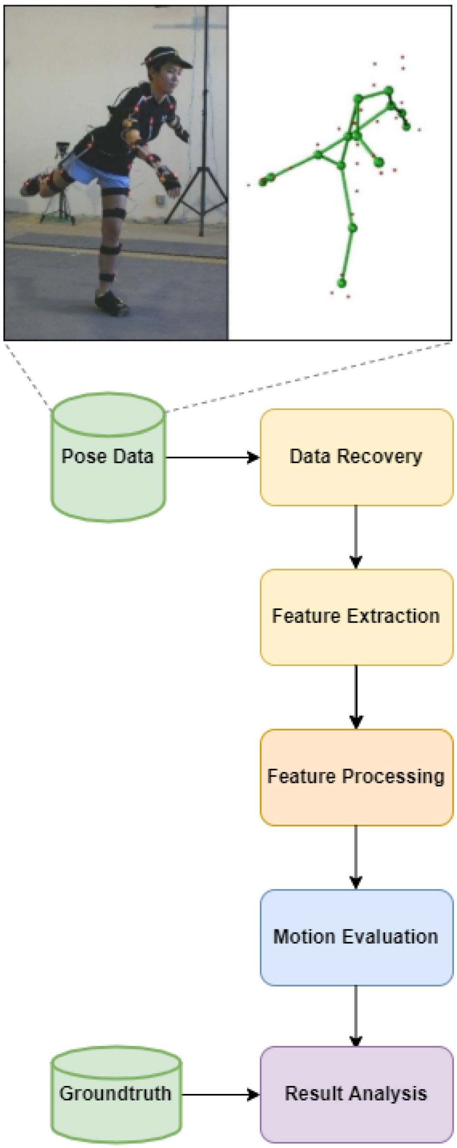


FIGURE 4.2: Motion Evaluation from Pose Input (optical MoCap or Kinect)

weighted sum of the COM position of each bone. We can compute the kinetic energy of the body using its centre of mass.

4.3.1.1.3 Relational Features (Müller) Relational features relate joins and bones motions based on human knowledge. The distance or angle between two joints is an example of a relational feature. We use 39 binary relational features introduced by Müller et al. [296] for human motion retrieval and also applied to classification, segmentation, annotation and gesture evaluation.

- F_{angle}: The angle between two body bones that are defined by joints j_1, j_2, j_3 and j_4
- F_{fast}: The normal speed of the joint (j_1)

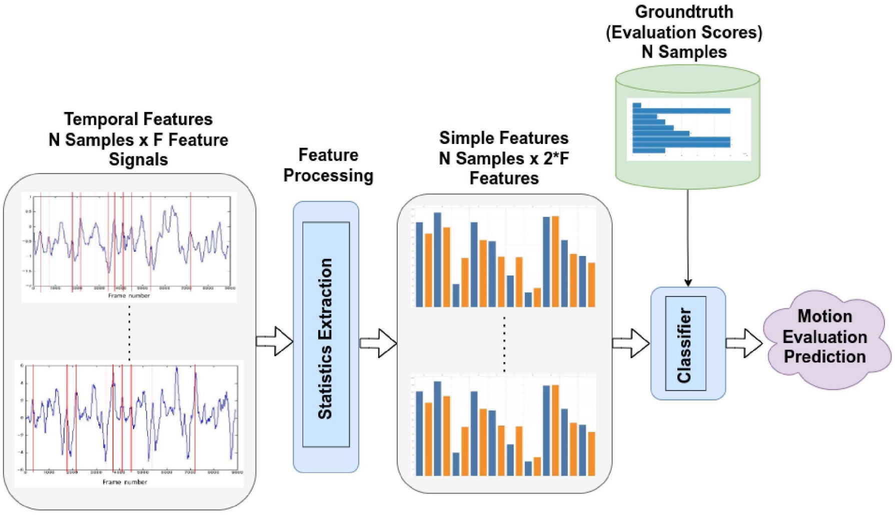


FIGURE 4.3: Motion Feature Processing and Motion Evaluation

- F_{plane}: The distance of the joint j_4 and the plane defined by the joints j_1, j_2 and j_3

- F_{nplane}: The distance of the joint j_4 and the plane with a normal defined by the joint j_1 and joint j_2 segments and passing through the joint j_3

- F_{move}: The speed of the joint j_4 in the direction of the bone $j_1 \rightarrow j_2$ with respect to the joint j_3

- F_{nmove}: The speed of the joint j_4 in the direction of the normal of the plane defined by the joints j_1, j_2 and j_3 with respect to the joint j_1

Threshold values are used for converting the features to a binary value using Schmitt trigger [417]. An illustration of some Müller relational features are shown in the Figure 4.4.

4.3.1.1.4 Ergonomics Features Biomechanical properties of the motion can be reflected in ergonomics features. For example, the quality of movement in the areas like comfort, load and robustness. For each scientific field of motion analysis different ergonomics features are introduced in the literature, for example based on the Taijiquan ergonomic principle, a set of features is made by Tits et al. [418], [416].

- Balance: The balance of the motion means having stability caused by equal weight distribution. Balance can be measured by finding the distance between the projection of the COM on the ground and the centre of the support base. Support points are the points that are in contact with the ground. The support base is then the area between all support points. Increasing the distance measure of balance decreases the balance of the object. In a dynamic situation other factors such as the velocity of the COM or the stabilizing force that keeps the COM above the support base should be taken into account [419], [420].

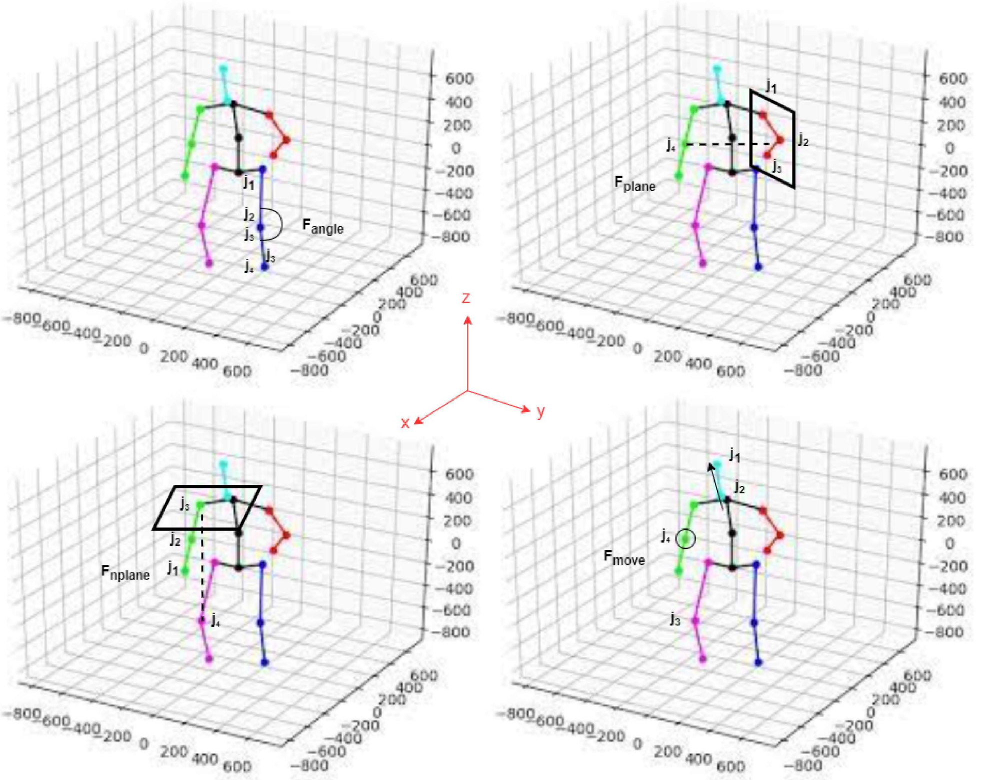


FIGURE 4.4: An illustration of Müller relational features

- Range of Motion (ROM): The ROM of each joint is the maximum rotation along each of its degrees of freedom and will vary among individuals.

- Postural Load: The postural load of the body can give us information about how comfortable a posture is. In extremis, it may convey information when an injury may occur. The sum of all joint stresses defines the postural load of the overall body.

- Torques: Are moments of forces that cause rotation of the joints. They have their application in evaluating muscle exertion or articular load in ergonomics. To compute the torques, we need the weight of the body segments, ground reaction forces and accurate measurement of accelerations. Details can be found in [421].

- Coordination/synchronization: Measuring synchronized motion of the joints or limbs have applications in analysis of gait, dance and sport movements. There are methods ranging from a formula for a specific action to more complex ways like neural networks, DTW or PCA for finding synchronization in more general settings.

- Tai Chi Ergonomics Features: In this chapter, different features used for score analysis in Tai Chi motions, including the ergonomics features are listed. In this section we explain further how these features can be computed from the

3D human kinematic model illustrated in Figure 4.5. The importance of these features is that ergonomics are very related to the skill, therefore, can be an indicator of motion quality. In general terms, ergonomics studies the effectiveness of motor control while minimizing the injury risk and energy consumption. This new set of ergonomic features for martial arts that was first introduced by [416] for Tai Chi motion sequences was inspired by the work of the ergonomist and Taijiquan teacher Eric Caulier [422]. The ergonomic features are categorized into four major groups: Stability, Joint Alignment, Favourable Angle and Fluidity. These four groups are explained in depth in the next paragraph.

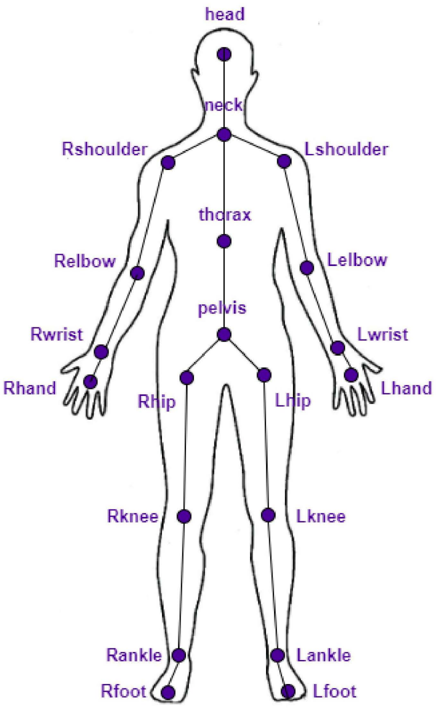
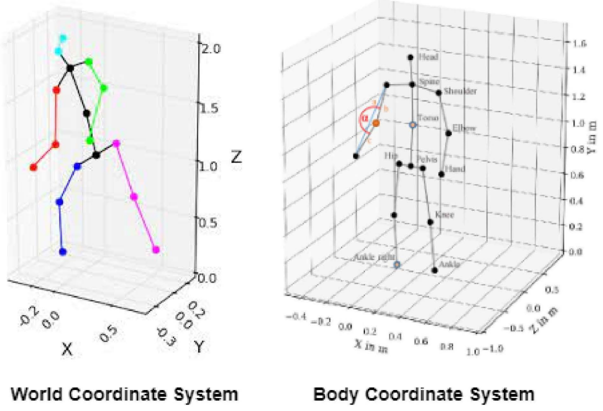


FIGURE 4.5: Body joint names of the 3D human kinematic model

4.3.1.1.5 Tai Chi Ergonomic Features As mentioned before, a set of ergonomic features were first introduced by [416] inspired from the work related to Tai Chi Movements [422]. An illustration of some ergonomic features are shown in the Figure 4.7.

4.3.1.1.5.1 Stability The body of the Tai Chi performer should be stable during the motion. Four different features are implemented to evaluate the stability:

- **Static Stability**: The Euclidean distance between the x and y components of the COM and Pelvis joint positions in the horizontal (x, y) plane
- **Dynamic Stability**: The time derivation of static stability



World Coordinate System

Body Coordinate System

FIGURE 4.6: World coordinate system and body frontal coordinate system

- **Verticality**: The Euclidean distance between the x and y components of the pelvis and the neck joint position in the horizontal plane shows verticality of the trunk

- **Horizontality**: The mean absolute difference between the height (z) of the shoulders, hips and knee joints shows horizontality of the body

4.3.1.1.5.2 Joint alignments

- **Joint Vertical Alignments**: Euclidean distance between horizontal components of two joints

- **Shoulder-Wrist Frontal Alignment**: The Euclidean distance between the coordinate of the left/right wrist and left/right shoulder in the body frontal plane.

  The body frontal plane can be defined by three points which are the two hip joints and the neck joint. This will define a local coordinate system that is called the body frontal coordinate system.

- **Feet Alignment**: The absolute difference between the heels' distance and the toes' distance shows how much the feet are parallel

4.3.1.1.5.3 Favourable Angles The favourable angles of the joints are referred to the optimal joint flexion that is not fully stretched nor too bent. This can also be related to the optimal muscle length that produces the highest force.

- **Low Shoulders**: The angle between the shoulder, the neck and the thorax joints is extracted. The angle between the three joints j, k and i.

- **Elbow flexion deviation from the optimal angle**: The elbow flexion angle should be in range of 90° and 135°, the optimal angle is 112.5°.

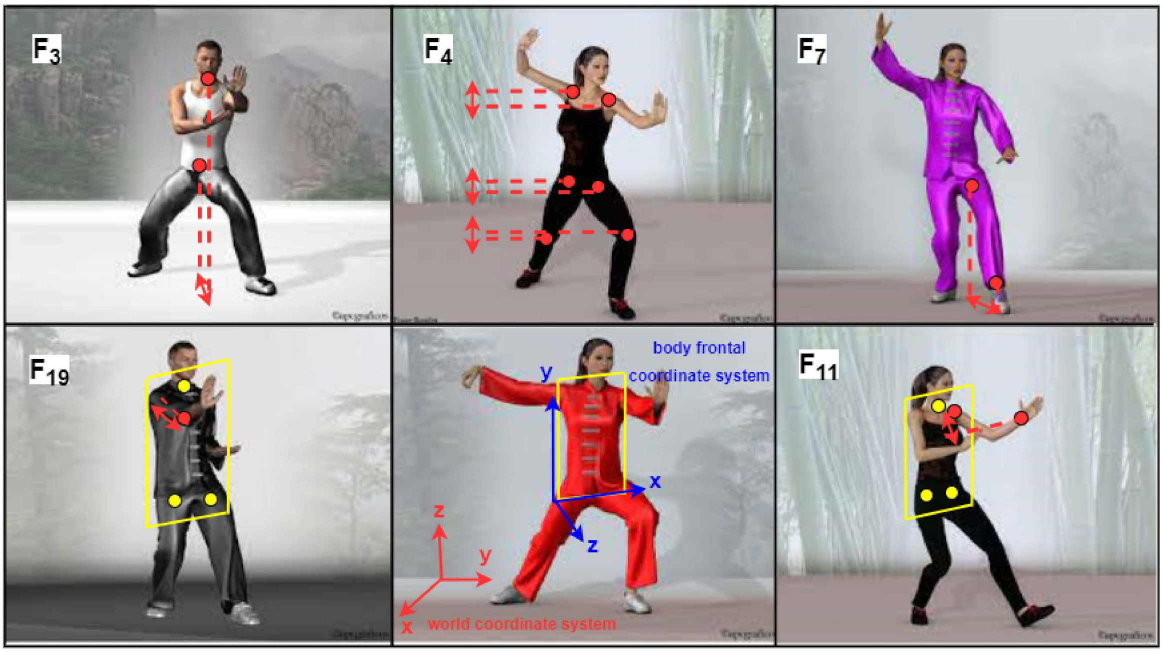


FIGURE 4.7: An illustration of ergonomic features and difference between world or global coordinate system and body frontal coordinate system

- **Elbows not behind body**: The z-coordinate in body frontal coordinate system is extracted to evaluate if the elbow is behind the body.
- **Elbow not too low/high**: The deviation of elbow abduction from the optimal angle (67.5°).

4.3.1.1.5.4 Fluidity The motion of the body should be smooth and should not have jerk. The fluidity of each limb and the trunk is found by computing the velocity (V), acceleration (A) and jerk (J) of the corresponding COMs.

4.3.2 Deep learning

In this section we demonstrate the pipeline for estimation and analysis of human motion in videos using deep neural networks (Figure 4.8). The first step is capturing and annotation of the video dataset of human motions. For the human motion evaluation step a set of complex features related to quality of martial art motions are computed as annotation.

The resulting moving 3D human model found from the monocular video input will not have perfect motion due to issues like pose ambiguity, occlusion, etc. Therefore, post-processing of the human motion result will be beneficial in increasing the accuracy. The 3D motion is then evaluated using deep neural networks and classic machine learning methods (such as regression and random forest). The result will be a number representing the quality of the motion.

4.3.2.1 Human mesh and pose estimation

In this section, we describe the method used for capturing 3D human motion from monocular videos. This is the first and second stage in Figure 4.8.

4.3.2.1.1 Monocular human pose estimation Human pose is usually represented by a set of 3D key points in space that are located on the main body such as joints and landmarks. An example of that can be seen in section 2.2.1.4.1. A more realistic 3D representation of the human body is SMPL (Skinned Multi-Person Linear Model) introduced by Loper et al. [104] (see section 2.2.1.4.2). It is a vertex-based model that can describe a wide variety of body shapes and its parameters are learned from the data. The pose parameters of the model describe the three rotation values of the joints and shape parameters can alter the shape of the human body. Figure 4.9 shows an example of an SMPL model and its joints locations.

4.3.2.1.2 Generating a 3D training dataset As mentioned in the previous section, one of the main challenges that limits research in the area of human motion analysis is the lack of data. Consequently, being able to produce a 3D dataset is of great importance. To produce large datasets, some commercial companies resort to generating synthetic data in a virtual environment [423] or random poses [424]. Whilst simulated data have an advantage of knowing the ground truth beforehand and have more control over motion and environment, they lack realism as compared to in the wild videos.

In order to train the pose and shape estimation model, there is a need for a training dataset that consists of videos and their corresponding 3D pose and body shape. Most of the currently available benchmark datasets include videos of a set of RGB cameras and 3D human data from an optical motion capture system that is synchronised with this multi-camera system - see Figure 4.10. Apart from the optical motion capture system, a multi-camera system can produce the desired 3D poses [425]. Inertial Measurement Units (IMUs) [112] and Kinect sensors [426] can also be used to produce the human 3D data.

4.3.2.2 Human movement analysis

In this section, we explain a method for prediction of some motion metrics for the human movements using deep neural networks. For training the network, we compute these metrics from a large dataset of human motion capture data using a mathematical explanation of these measurements. We use pose parameters of the SMPL model as a motion signal and input of the network and the output would be the prediction of some desirable aspects of the movement, such as stability, joint alignment, fluidity of motion or having desired joint angles.

Unlike many current models that rely on a complex hardware setup like optical motion capture, IMUs or depth sensors, this method makes use of standard video as a cheap and convenient alternative. In many scenarios of human movement analysis, designing a set of effective features that can predict a specific phenomenon can be very difficult. In the previous section, defined and computed a wide variety and types of features. In contrast to this designed features, deep neural networks use multiple layered artificial neural networks to learn a complicated relationship between input and output, which sometimes cannot be explained with human handcrafted features. Furthermore, neural networks can get the complete human motion signal as an input in the form of time-series while in the common statistical methods like regression and random forest only some motion signal statistics like the mean and standard deviation are assigned as the input of the model. This can be an advantage for neural networks as more details about the input movement signal can be preserved and used.

4.3.2.2.1 Generating the Annotation Dataset In order to predict the motion metrics from the human motion videos, an annotated dataset of motions and their corresponding measures are required. Generally, assigning a specific motion metric or score to a motion can be complicated and usually is done by different means and protocols depending on the subject of study and type of motions. We are doing automatic annotation of metrics that define the quality of the motion such as stability, fluidity, joint alignment and having favourable joint angles. As mentioned before, SMPL is a rigged body model (a standard skeletal representation and a fully rigged surface mesh). This similarity in format of the whole dataset, as well as its richness and quantity make it suitable for use in deep learning applications. To compute annotations for the human motion sequences in the dataset (Figure 4.12), we first converted the SMPL format to the common *.bvh* format of motion capture data and process the movement of the joints. Since we need a number associated with a sequence of frames, we have calculated the average of the measures computed for each frame, for example average of stability, fluidity values in a set of frames. It should also be noted that while converting the motion capture dataset to a training data, the frame rate of the data should be consistent with the future input data of the trained model which comes from a video source. The video data usually have a frame rate of 30 fps while optical motion capture data is recorded at higher frame rates, such as 60, 100 or 120 fps, so we have down sampled the MoCap data to match the video frame rate.

4.3.2.2.2 Preparation of automatic and hand-crafted timeseries The output of the human poses and shape estimation network is an SMPL model which contains information about 3D motion corresponding to the video of the input video. The motion of the model can be explained by the pose parameters (joint rotation vectors) of the SMPL model. To improve the performance of the neural network, we can add more handcrafted features that can explain the output of the neural network. The motion signal also should be pre-processed to provide a cleaner and more correct motion signal to the neural network. If there is missing data in the input signals, linear interpolation is used to fill these values. It is also necessary to cancel out the noise so a Gaussian filter (standard deviation $\sigma = 1$) is used for smoothing the signal. The human models in the dataset are the upgraded version of the SMPL model (SMPL-H) with more joints. Since the output of the video-based pose estimation is in the form of SMPL, we should convert the SMPL-H model to SMPL for compatibility.

4.3.2.2.3 Training Convolutional Neural Network CNNs are a type of deep learning models that are designed for processing data that has a grid pattern. Images are examples of this form of input data. In our case, each movement signal in a fixed time interval can be considered as a data vector and putting all the input movement signals together forms a 2D grid that can be used for training the CNN. The CNN is working as a mapping between the time-series movement data and the motion metric, so without needing to develop a large set of complicated handcrafted features the desired metric can be predicted.

We will use the CNN architecture from 2D human motion analysis research [429] for 3D analysis of the SMPL model motion. The input of the network is a multivariate time-series with fixed length. The main blocks of the network are 1-D convolutional layers with $T \times D$ neurons (T is in time dimension, D is the multivariate input dimension). Dense block is a multiple linear regression from input d_1 dimension to d_2 dimension (see Figure 4.12).

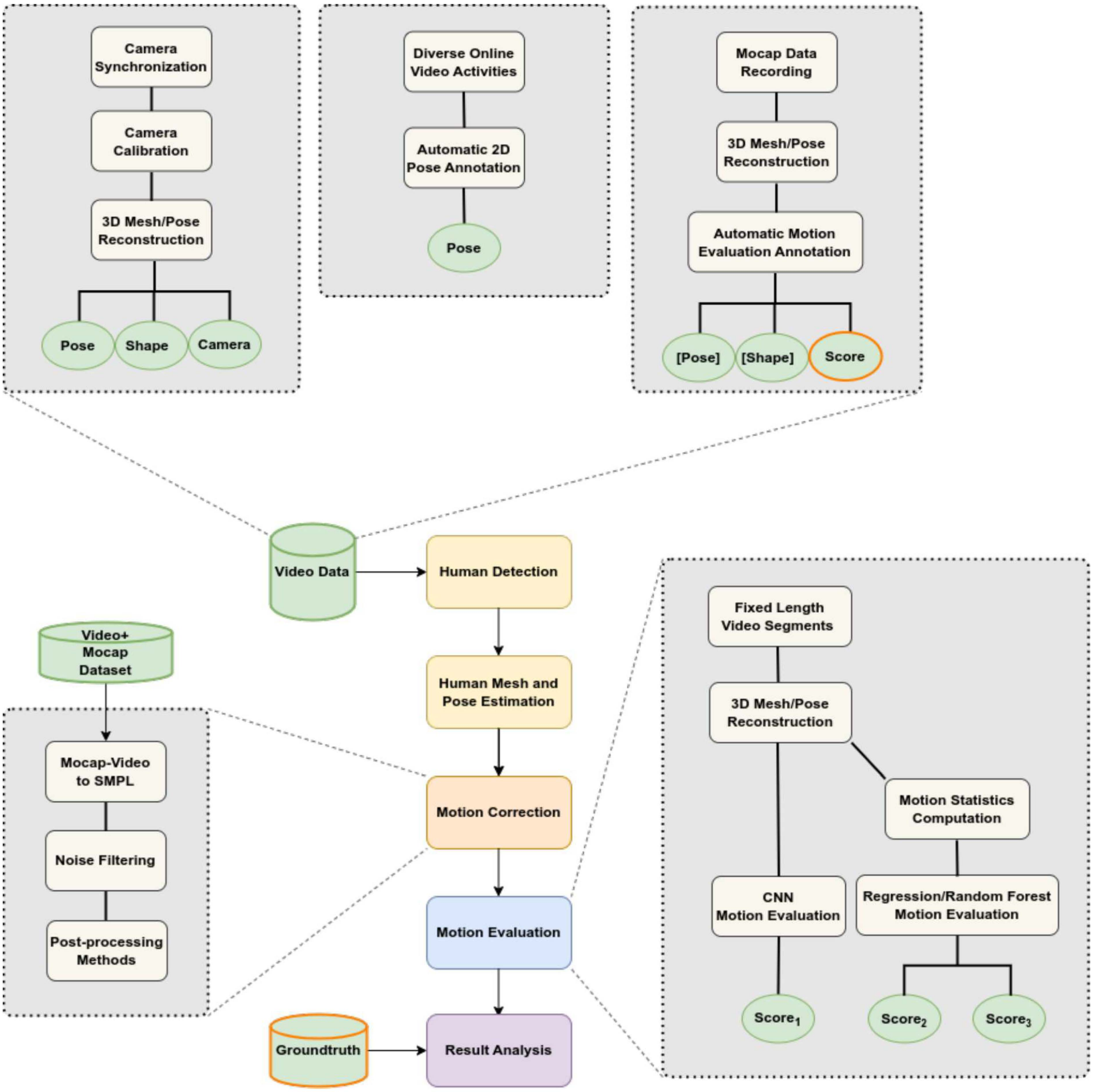


FIGURE 4.8: Deep learning based human motion estimation and analysis pipeline

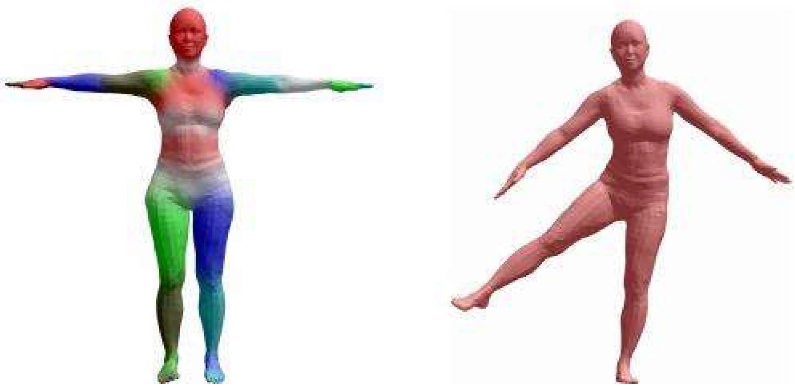


FIGURE 4.9: SMPL Human Body Model [104]

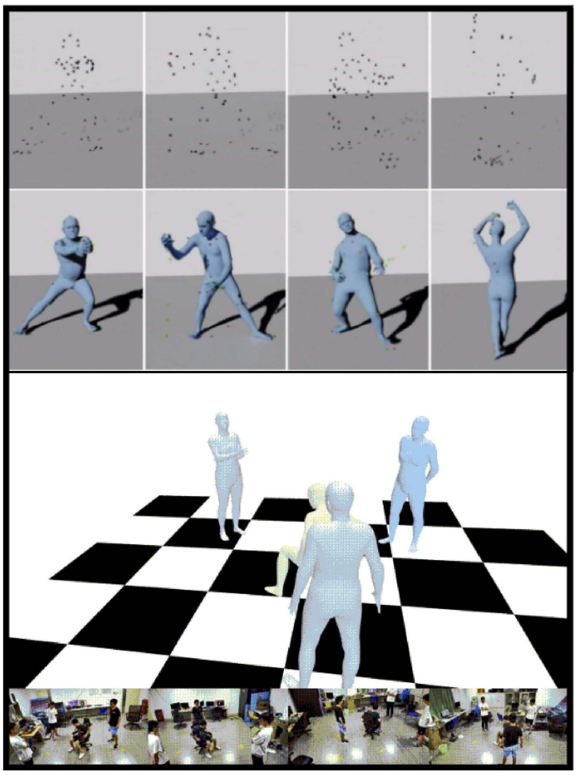
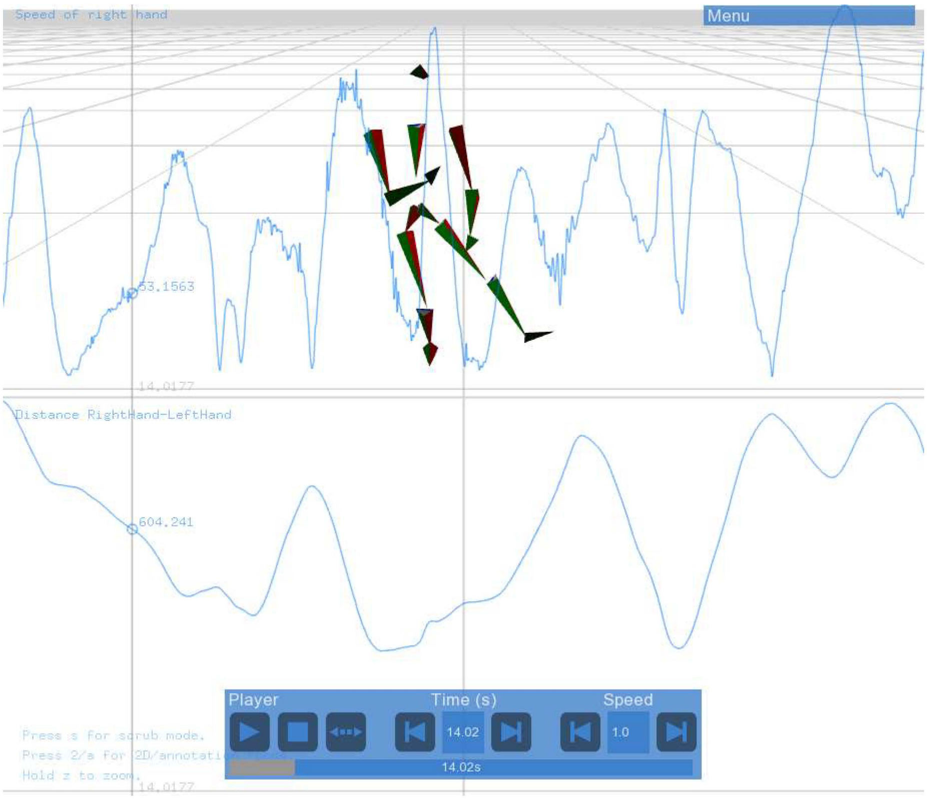


FIGURE 4.10: 3D Human model ground truth generation for the training dataset (using optical MoCap or multi-view camera system) [427], [428]



Press s for scrub mode.
Press 2/a for 2D/annotati
Hold z to zoom.

FIGURE 4.11: Motion evaluation annotation using AMASS dataset features

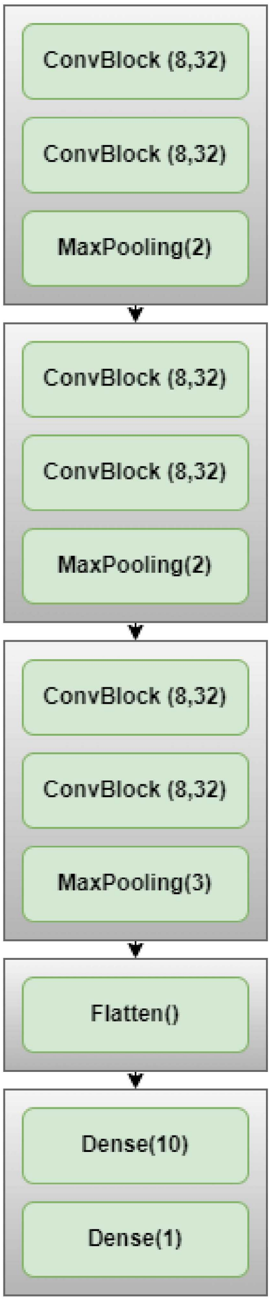


FIGURE 4.12: Convolutional neural network (CNN) architecture

4.4 Experimental Design

In this chapter the experimental design and the results regarding human motion evaluation are demonstrated. Improvements are made on the state-of-the-art methods in terms of increasing the accuracy. These improvements aim at reducing the error of human motion evaluation. For decreasing the error in human motion evaluation, we have studied the effect of each feature separately and using this method changed the pipeline to get a better result for prediction. The comparison between performance of deep learning based methods and classic machine learning methods is also demonstrated.

The output of the human motion analysis is the predicted score of the martial art movements. The error of score prediction is measured by computing the root mean square of the error between ground truth and predicted score. These errors are the the basis of the comparison between the different methods.

Figure 4.13 shows the pipeline of all experiments across human motion estimation and evaluation. After 3D motion estimation, evaluation of this motion using some example data sets is done. The effect of different combinations of features and classifiers and also comparison between deep learning and classical methods on a larger dataset is examined. The motion analysis is usually is done in the framework of specific application. The motion feature design (formulas), and dataset is based on the recent work on martial arts scoring [416]. We have changed the processing pipeline and further analyzed each feature separately to optimize the result.

The experiments in this research are designed according to the main objectives the research is aiming to achieve as stated in the introduction chapter. The designed methods are aiming to improve the error between the ground truth and prediction motion evaluation. Assuming we have a sequence of 3D human poses as a result of the human motion estimation part, the second part of project is motion evaluation. Motion evaluation is concerned with how well a certain action is performed. This is done by training a supervised machine learning model to predict the evaluation metric which is a single number assigned to the motion. Classic machine learning which is feature extraction combined with machine learning as well as deep learning methods are experimented. The features can be simple meaning derived from basic motion variables or complex and formulated using simple features. To understand the effect of each type of features, different combinations of each feature type and machine learning method is experimented with. This leads to finding the best feature and machine learning model to perform the motion evaluation task. Two types of experiments are done for training the motion evaluation for all the actions or for an individual action. For evaluating deep learning, a larger dataset of diverse motions is automatically annotated and performance of deep learning and classic machine learning methods are evaluated on the resulting dataset.

4.4.1 Datasets Details

The two main experiments in the human motion analysis part use different datasets with different types of input data.

Firstly, we have used the existing annotated dataset designed for evaluation of martial art motions. The input are the 3D joint positions and output will be the skill level of the martial artist based on how well they perform the gestures. The description of the dataset subjects and their skill level that is used as annotation is listed in Table 4.1.

ID	GN	Age	Weight	Height	Practice	Category	$Skill_1$	$Skill_2$	$Skill_3$	$Skill_\mu$
P01	M	56	95	196	32	Expert	9.3	9	10	9.43
P02	F	57	78	163	30	Expert	9.6	9.1	10	9.57
P03	F	62	58	162	24	Expert	8.5	8.5	9	8.67
P04	F	47	53	150	12	Advanced	8.2	8	8	8.07
P05	F	71	61	163	14	Advanced	6.8	7.4	7.5	8.07
P06	M	25	76	180	10	Advanced	8.4	8.6	8.5	7.23
P07	F	49	57	157	4	Intermediate	7	6.8	6.5	8.5
P08	F	34	56	158	3	Intermediate	8	7.3	7	6.77
P09	M	51	90	178	2.5	Intermediate	6.9	6.8	6.85	7.43
P10	F	59	55	163	1	Novice	6	5.8	6.5	6.1
P11	F	65	58	165	0.2	Novice	5	4.9	5	4.97
P12	M	28	96	181	0.6	Novice	5.8	6	5.75	5.85
M		50.33	69.42	168	11.11		7.46	7.35	7.55	7.45
SD		14	15.93	12.46	11.15		1.37	1.29	1.33	1.38

TABLE 4.1: Tai-chi dataset description of participants [416]

A set of Tai Chi motions from eight techniques (Bafa Techniques) of the Yang Taijiquan styles in the form of 3D human kinematic models are used for analysis. Each of the Tai Chi gestures has a certain name. They are numbered from 6 to 13

1. driving the monkey away
2. moving hands like clouds
3. part of the wild horse's mane
4. the golden rooster stands on one leg
5. fair lady work shuttles
6. kick with the heel
7. brush knee and twist step
8. grasp the bird's tail

A separate model is trained for each gesture, using motion sequences from 11 subjects for training and 1 subject for testing. We have also trained a model using all the gesture motions and reported the RMS error and correlation.

The second main experiment is the comparison of deep learning and classic machine learning models on SMPL-based dataset AMASS dataset [106] is an aggregation of a large set of human motion capture datasets with a total of more than 11000 motions. The dataset is annotated automatically by computing some evaluation metrics. Unlinke the previous dataset used for motion evaluation that had joint positions as input, this dataset has joint rotation values in axis-angle format as the input. The list of datasets that are used for creating the AMASS dataset is as follows:

Sub-Dataset	Markers	Subjects	Motions	Minutes
ACCAD	82	20	258	27.22
BioMotion	41	111	3130	541.82
CMU	41	97	2030	559.18
EKUT	46	4	349	30.74
Eyes Japan	37	12	795	385.42
HumanEva	39	3	28	8.48
KIT	50	55	4233	662.04
MPI HDM05	41	4	219	147.63
MPI Limits	53	3	40	24.14
MPI MoSh	87	20	78	16.65
SFU	53	7	44	15.23
SSM	86	3	30	1.87
TCD Hand	91	1	62	8.05
TotalCapture	53	5	40	43.71
Transitions	53	1	115	15.84
Total	-	346	11451	2488.01

TABLE 4.2: Datasets contained in AMASS [106], More than 42 hours of marker data is unified by converting to the SMPL format

4.4.2 Experimental Setup

Human motion evaluation of martial arts required feature extraction. Implementation of features is done in C++ using the Motion-Machine library and Armadillo C++ Linear Algebra Library which makes implementation of the features easier. These features are then processed in Python machine learning libraries to complete the pipeline. The motion evaluation of the SMPL motions in the AMASS dataset is done completely in Python and joint rotations are used directly as an input to the machine learning models without feature extraction.

4.4.3 Baseline Methods

In terms of human motion evaluation, the standard motion analysis pipeline with PCA after feature extraction is used in the main baseline. Therefore, in terms of comparison between our result and previous work, the existing work on a martial arts evaluation dataset UMONS-TAICHI used as a baseline [418].

4.4.4 Evaluation Metrics

For the human motion evaluation, the error is computed as a difference between the score assigned by the expert and the score found by the machine learning model. The root mean squared error as well as Pearson correlation coefficient is computed. For the pseudo-score ground truth annotations computed by formulas instead of experts, we only use the correlation as error metric because the pseudo-score are not normalized within an specific range and different formulas have different ranges.

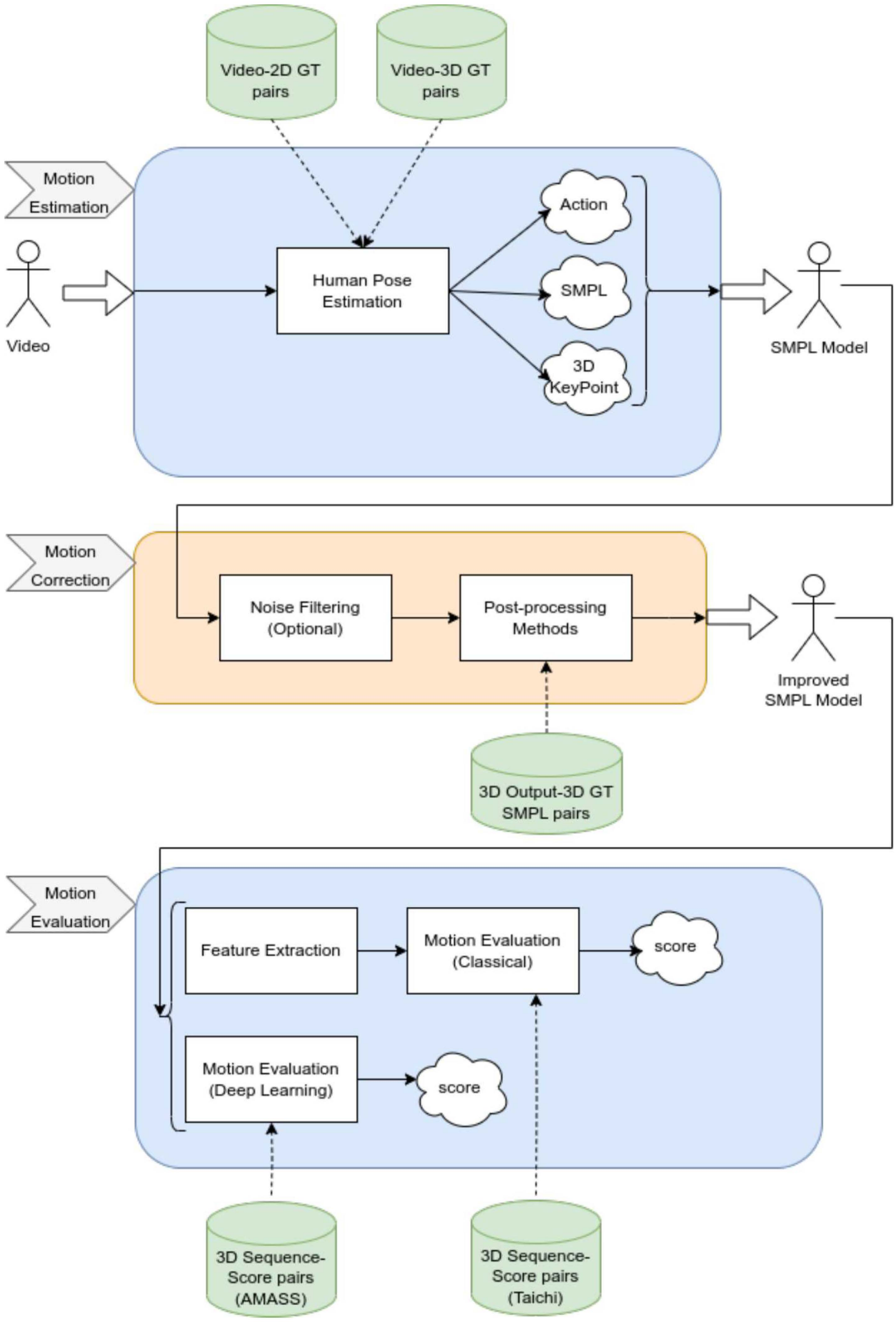


FIGURE 4.13: Human Motion Estimation and Evaluation Pipeline. The improved motion from the SMPL model after post-processing can be evaluated using classical machine learning or deep learning methods. Classic machine learning methods first extract the features from the motion, then statistics such as mean, standard deviation and quartiles of the feature signal is used to train the random forest and regression model. The deep learning method is using the input motion which is axis-angle joint rotational values directly.

4.5 Results

In this section, the result of human motion analysis using classic machine learning and deep learning methods are reported.

4.5.1 Classic Machine Learning

Classic Machine Learning for analysis of motion, consists of two consecutive stages of feature extraction and machine learning. As mentioned before we have used the different motion features and the dataset described in [416]. Examples of different features are shown in the Appendix section A.2. The baseline method combined different features together and ran several feature processing methods including PCA and morphology independence post-processing. While dimensionality reduction is a common practice when dealing with high dimensional data, we can show that it is possible to reduce the dimensionality of input by using less features as an input instead and get a better result. The result of experiments with single features are demonstrated in this section.

Different machine learning models (random forest and ridge regression) compared to the work in [416] are used. We have also further analyzed the effect of using each feature and each machine learning method in our implementation. It can be shown that specific combination of features and machine learning methods can lead to a better correlation between the predicted and actual gestures compared to the baseline method. We have also tried to train the models on specific gesture data or on all gestures and compared the results.

The best result from the baseline method is shown in the Figure 4.15. The related work [418] have used EN-Regression, 60 PCs and Morphology Independence feature post-processing. We have used separate features instead of combined features along with Random Forest and R-Regression (Ridge Regression).

By reducing the number of features in use, no PCA or feature post-processing needs to be used. Table 4.3 shows the result of using individual features combined with two new machine learning models (random forest and ridge regression) and training the models on all gestures. The RMSE error and correlation between the ground truth and the predicted score are reported. We could increase the best achievable correlation up to 0.99 with different combination of features and machine learning methods. The Muller features with the random forest had the best performance and Kinematic features with random forest is the second best combination. In average, it can be seen that random forest performed better than ridge regression for motion evaluation.

Tables 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 4.10, 4.11, 4.12, 4.13, 4.14 shows similar results to the Table 4.3 but with training on specific martial gestures (gesture 3 to gesture 13). The best combination of features and machine learning models for each gesture based on correlation value is as follows (RF is used for random forest and RG for ridge regression):

- Gesture 3: Quaternion and RG, Ergonomic and RF
- Gesture 4: Quaternion and RG, Euler and RG
- Gesture 5: Kinematic and RG, Euler and RG
- Gesture 6: Euler and RG, Muller and RF
- Gesture 7: Euler and RF, Quaternion and RF (or RG)
- Gesture 8: Muller and RF, Kinematic and RG
- Gesture 9: Euler and RF, Kinematic and RF
- Gesture 10: Kinematic and RF, Quaternion and RG
- Gesture 11: Muller and RG, Kinematic and RG
- Gesture 12: Quaternion and RG, Muller and RG
- Gesture 13: Ergonomic and RF, Kinematic and RG

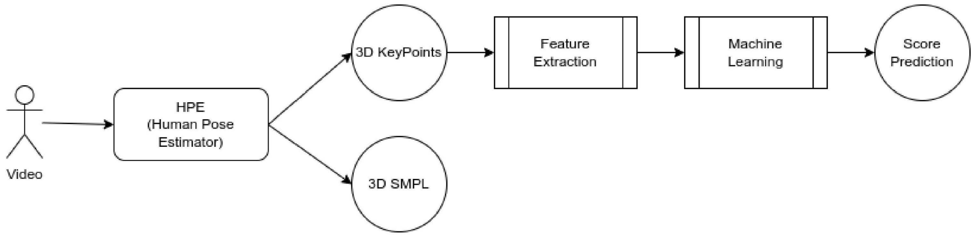


FIGURE 4.14: Human Motion Evaluation using Classic Machine Learning

4.5.2 Deep Learning

The main aim of this part is using CNN deep learning and comparing in to the other classic machine learning models random forest and ridge regression. No feature extraction is used and the input of both methods are joint angles of the SMPL output of human pose estimator. Since the previous dataset [416] ground truth in classic machine learning was in the form of 3D key points (joint positions) and the output of human pose estimator in this thesis is the SMPL model (joint rotations), we have used an SMPL-based dataset for this part.

Figure 4.16 shows the process of human motion evaluation using deep learning. The 3D SMPL model resulting from the human pose estimation is evaluated using classic machine learning and deep learning by analysing its joint rotations (pose parameter of the SMPL model).

The deep neural networks usually need more data compared to the classic machine learning models. Since we are using a supervised learning method, a large

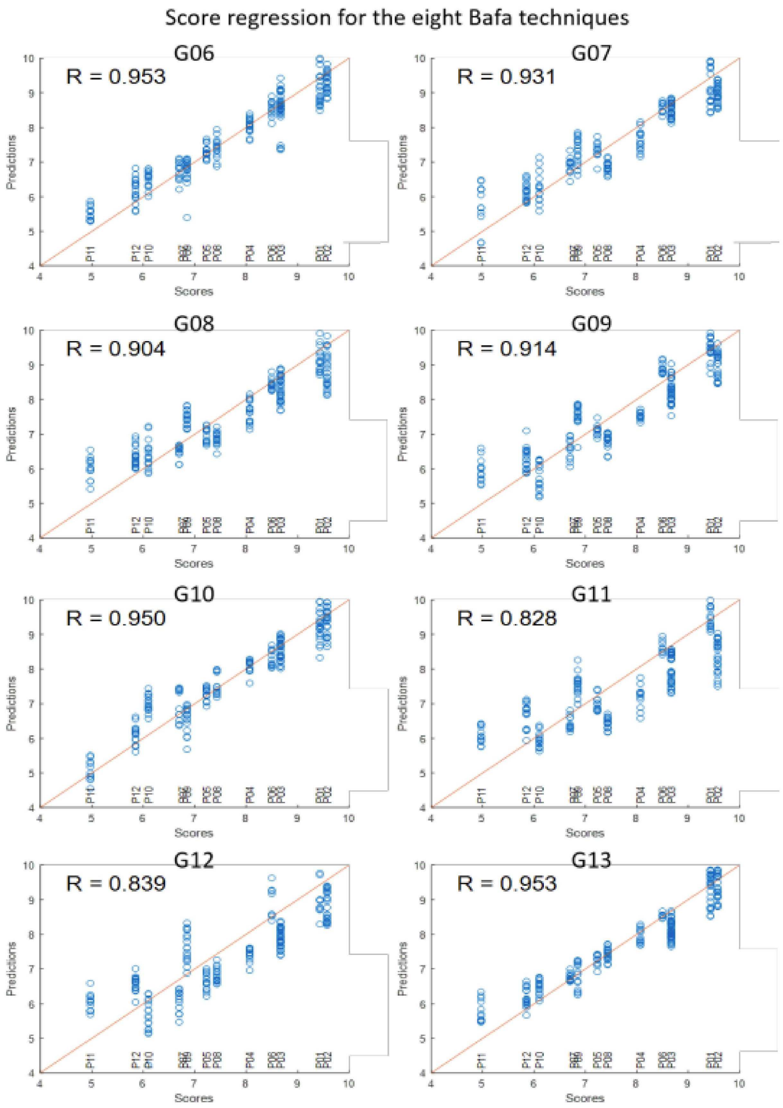


FIGURE 4.15: Baseline [416] Score Predictions using classic machine learning for all gestures with combined features, 60 PCs and Morphology Independence Feature Processing

SMPL based benchmark dataset (AMASS) [106] is annotated for score targets. Mathematical formulas for motion features that indicate a better score (for example desirable joint angles, joints alignment, stability and fluidity of motion), are used for annotating the AMASS dataset.

The results of human motion evaluation are shown in the Table 4.15 to 4.18. In Table 4.15, the dataset is annotated using the desirable angle formulas. In Table 4.16, the dataset is annotated using the motion fluidity formulas. In Table 4.17, the dataset is annotated using the joint alignment formulas. In Table 4.18 the dataset is annotated the movement stability formulas. Each of these four annotation categories has more than one criteria for computing the amount of stability, fluidity, joint alignment or desirable angles which can involve different parts of the body. Each row of the result tables show the result of using different annotation method (formula) within the

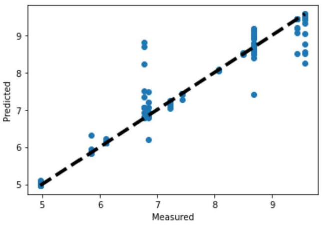
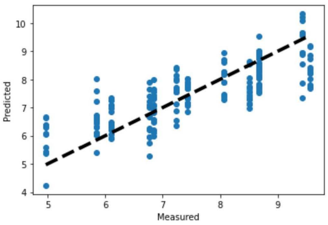
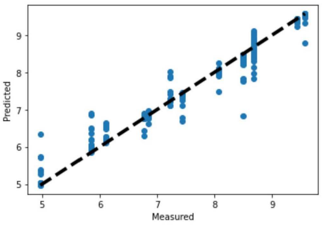
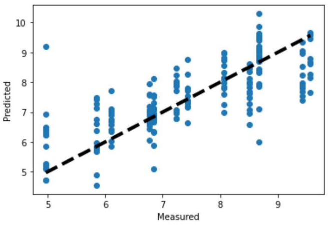
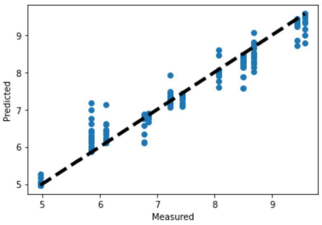
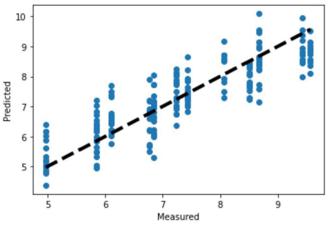
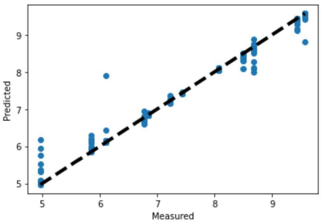
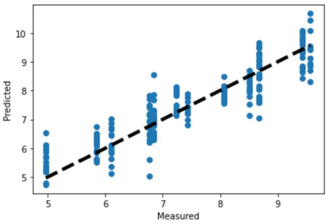
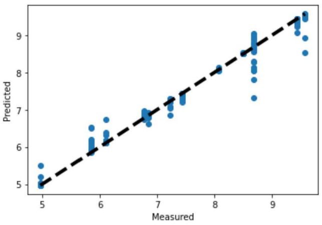
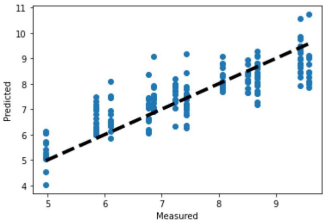
Features	Random Forest	Regression
Ergonomic	 correlation = 0.97 , rmse = 0.33	 correlation = 0.80 , rmse = 0.80
Euler	 correlation = 0.97 , rmse = 0.32	 correlation = 0.77 , rmse = 0.86
Quaternion	 correlation = 0.98 , rmse = 0.30	 correlation = 0.86 , rmse = 0.67
Kinematic	 correlation = 0.99 , rmse = 0.23	 correlation = 0.90 , rmse = 0.58
Muller	 correlation = 0.99 , rmse = 0.21	 correlation = 0.84 , rmse = 0.74

TABLE 4.3: Score Predictions for all Gestures

specified category. The result of all annotation formulas are shown in the Appendix section A.1 and more distinct result in each annotation category are mentioned in this section.

In motion evaluation using desirable angle (Table 4.15) and joint alignment criteria (Table 4.17) some annotations can be predicted easily by both deep learning

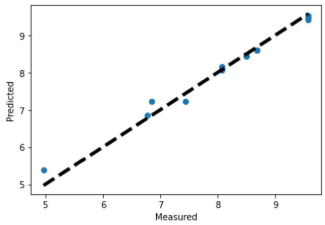
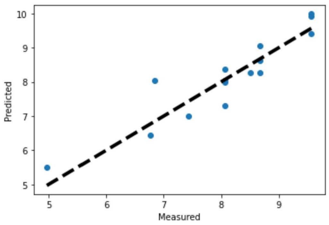
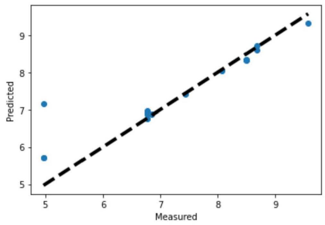
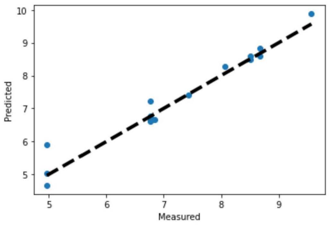
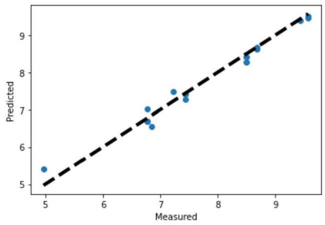
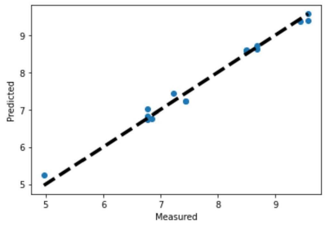
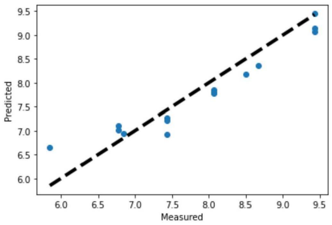
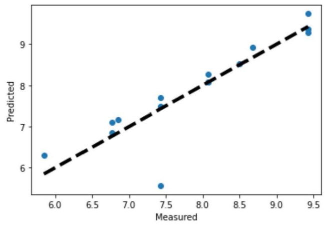
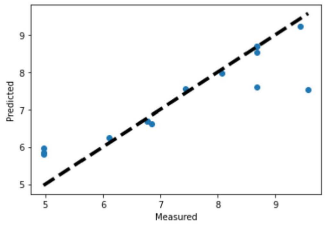
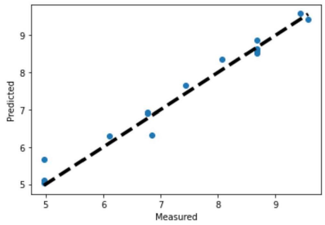
Features	Random Forest	Regression
Ergonomic	 correlation = 0.99 , rmse = 0.17	 correlation = 0.92 , rmse = 0.47
Euler	 correlation = 0.93 , rmse = 0.64	 correlation = 0.98 , rmse = 0.31
Quaternion	 correlation = 0.99 , rmse = 0.18	 correlation = 0.99 , rmse = 0.15
Kinematic	 correlation = 0.96 , rmse = 0.34	 correlation = 0.89 , rmse = 0.53
Muller	 correlation = 0.91 , rmse = 0.72	 correlation = 0.99 , rmse = 0.27

TABLE 4.4: Score Predictions for Gesture 3

and classic machine learning while in the more difficult ones CNN has the best performance, random forest is the second best and ridge regression has the worst performance. In motion evaluation using fluidity (Table 4.16) and stability (Table 4.18) criteria, CNN performs better in all the cases. Random forest comes after CNN in terms of performance and ridge regression in the last in the ranking.

Overall, the correlation between correct score and predicted score can show that in almost all scenarios CNN can perform better than the classic methods such as

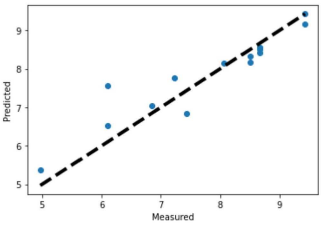
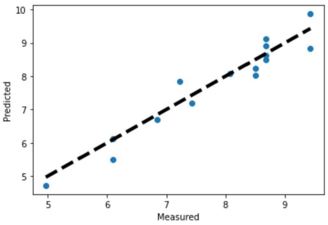
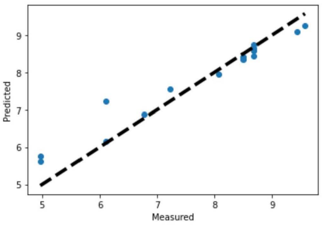
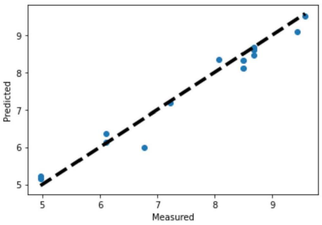
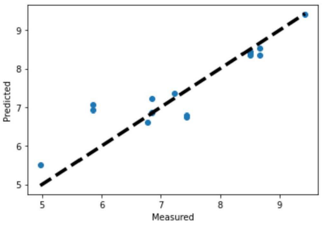
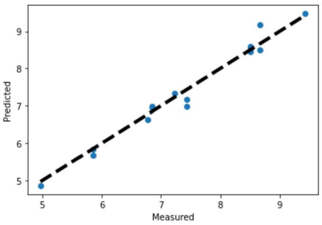
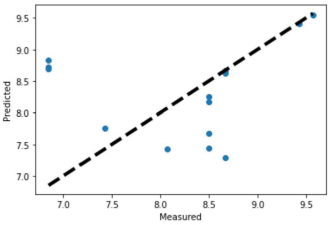
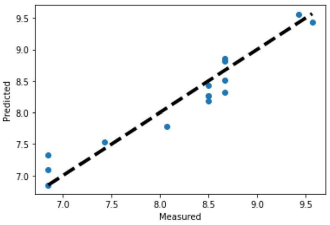
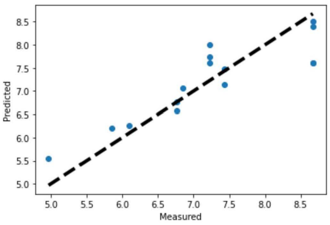
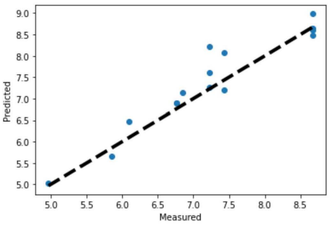
Features	Random Forest	Regression
Ergonomic	 correlation = 0.94 , rmse = 0.48	 correlation = 0.97 , rmse = 0.37
Euler	 correlation = 0.98 , rmse = 0.43	 correlation = 0.98 , rmse = 0.28
Quaternion	 correlation = 0.91 , rmse = 0.52	 correlation = 0.99 , rmse = 0.21
Kinematic	 correlation = 0.14 , rmse = 1.01	 correlation = 0.96 , rmse = 0.24
Muller	 correlation = 0.89 , rmse = 0.51	 correlation = 0.95 , rmse = 0.37

TABLE 4.5: Score Predictions for Gesture 4

regression and random forest. Between the classic machine learning methods, the random forest can perform better compared to regression method.

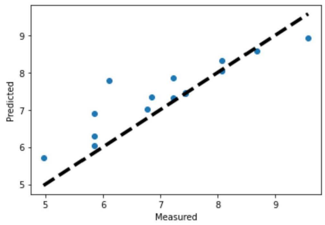
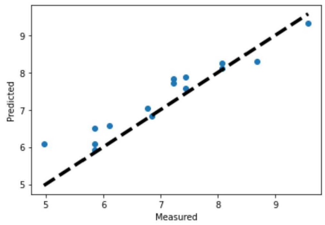
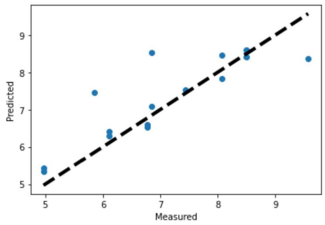
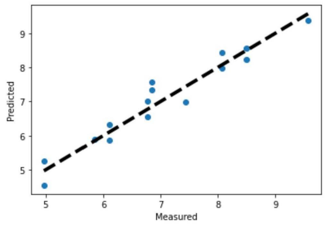
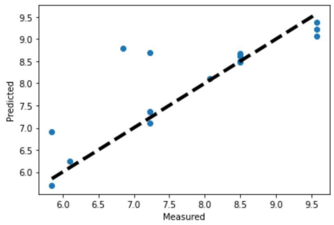
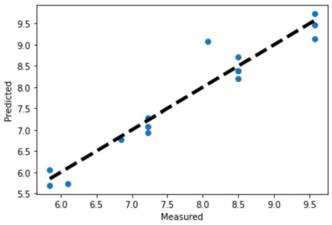
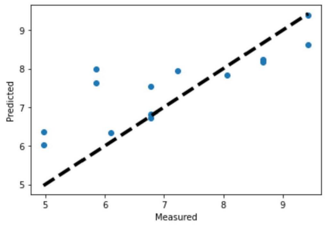
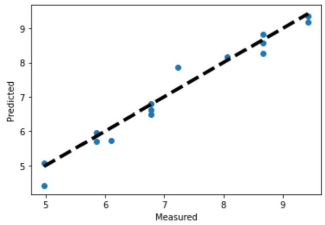
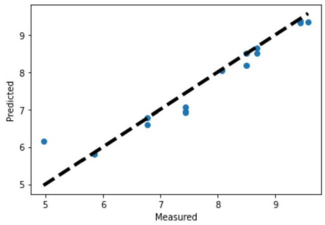
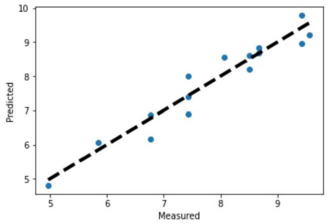
Features	Random Forest	Regression
Ergonomic	 correlation = 0.91 , rmse = 0.63	 correlation = 0.96 , rmse = 0.45
Euler	 correlation = 0.85 , rmse = 0.71	 correlation = 0.97 , rmse = 0.34
Quaternion	 correlation = 0.85 , rmse = 0.71	 correlation = 0.97 , rmse = 0.34
Kinematic	 correlation = 0.84 , rmse = 0.94	 correlation = 0.98 , rmse = 0.29
Muller	 correlation = 0.96 , rmse = 0.39	 correlation = 0.96 , rmse = 0.36

TABLE 4.6: Score Predictions for Gesture 5

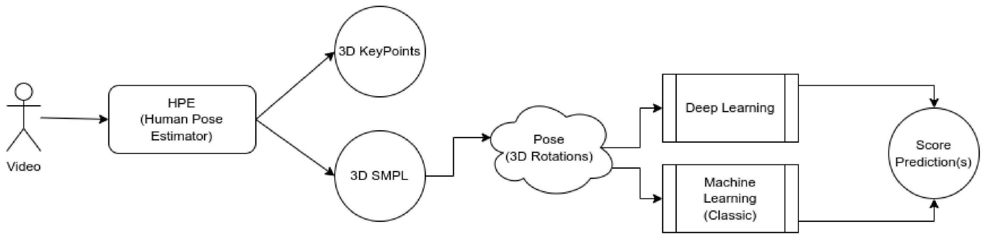


FIGURE 4.16: Human Motion Evaluation using Deep Learning

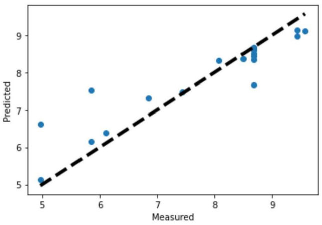
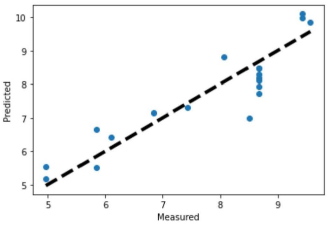
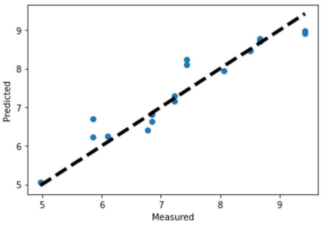
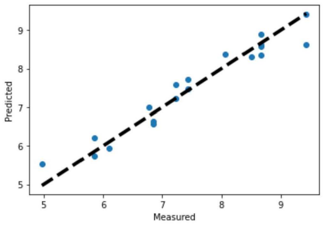
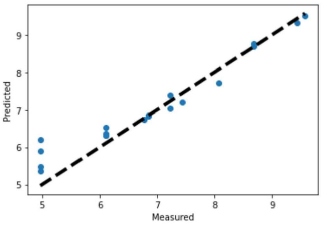
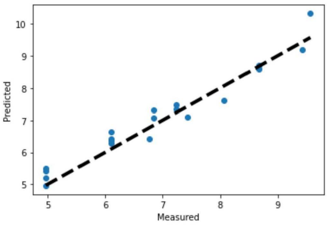
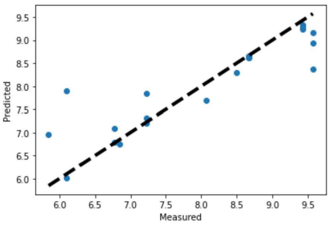
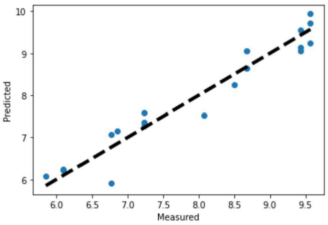
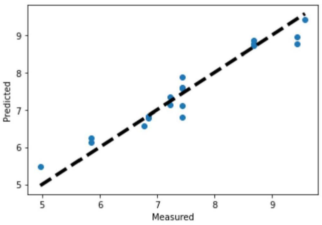
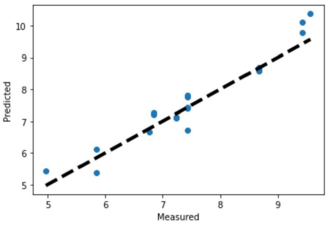
Features	Random Forest	Regression
Ergonomic	 correlation = 0.92 , rmse = 0.64	 correlation = 0.91 , rmse = 0.62
Euler	 correlation = 0.96 , rmse = 0.37	 correlation = 0.97 , rmse = 0.31
Quaternion	 correlation = 0.97 , rmse = 0.42	 correlation = 0.97 , rmse = 0.36
Kinematic	 correlation = 0.89 , rmse = 0.33	 correlation = 0.97 , rmse = 0.34
Muller	 correlation = 0.97 , rmse = 0.33	 correlation = 0.97 , rmse = 0.39

TABLE 4.7: Score Predictions for Gesture 6

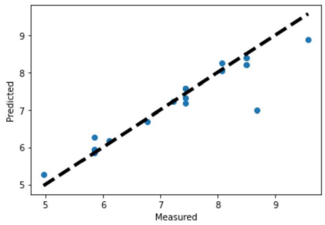
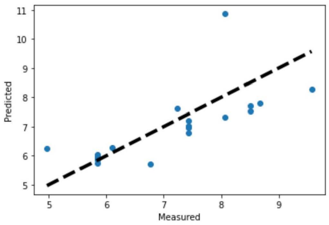
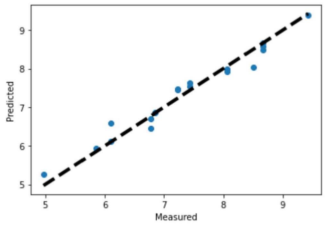
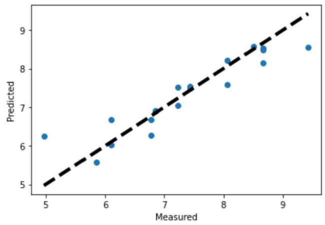
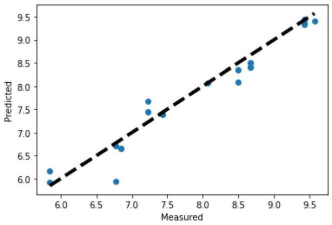
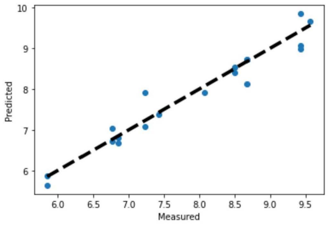
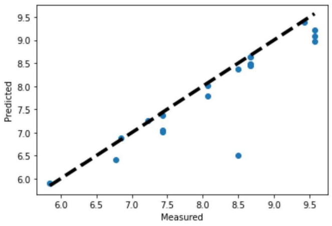
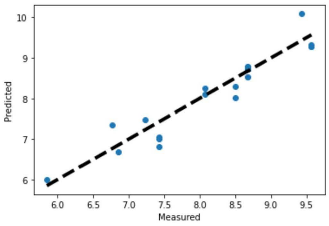
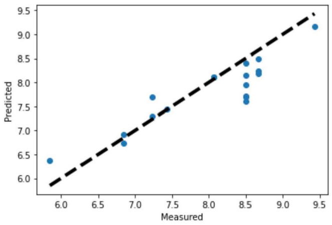
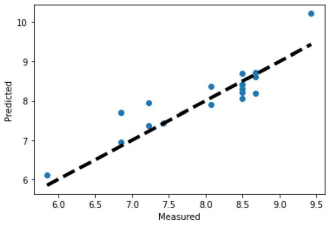
Features	Random Forest	Regression
Ergonomic	 correlation = 0.93 , rmse = 0.46	 correlation = 0.70 , rmse = 0.95
Euler	 correlation = 0.98 , rmse = 0.22	 correlation = 0.92 , rmse = 0.46
Quaternion	 correlation = 0.97 , rmse = 0.28	 correlation = 0.97 , rmse = 0.29
Kinematic	 correlation = 0.91 , rmse = 0.54	 correlation = 0.95 , rmse = 0.35
Muller	 correlation = 0.91 , rmse = 0.44	 correlation = 0.90 , rmse = 0.39

TABLE 4.8: Score Predictions for Gesture 7

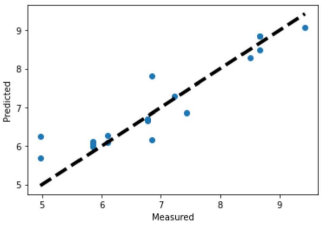
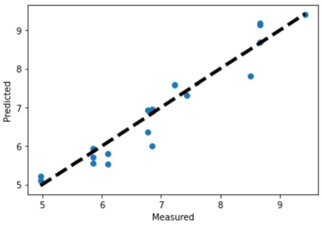
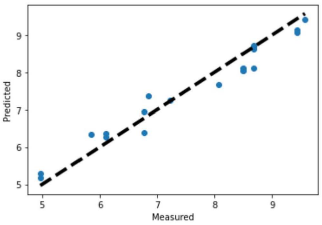
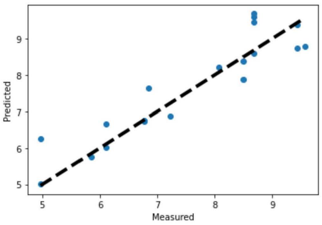
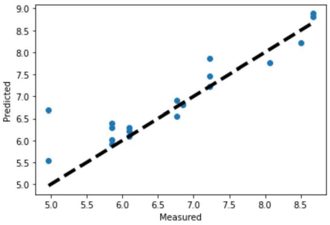
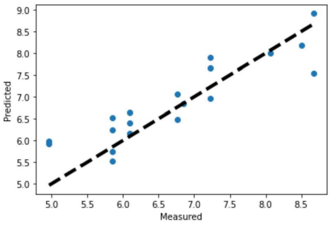
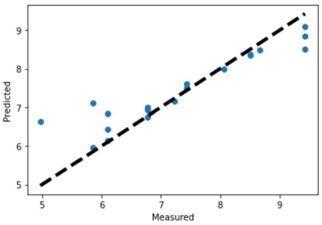
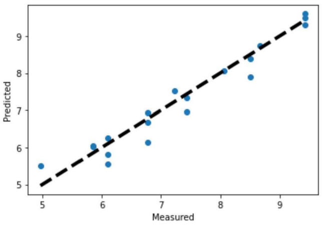
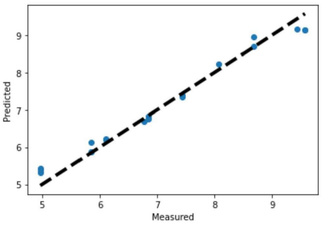
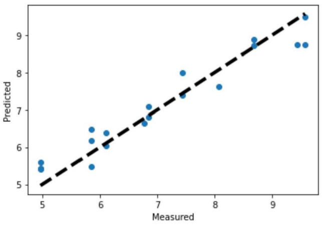
Features	Random Forest	Regression
Ergonomic	 correlation = 0.93 , rmse = 0.48	 correlation = 0.96 , rmse = 0.37
Euler	 correlation = 0.98 , rmse = 0.33	 correlation = 0.92 , rmse = 0.59
Quaternion	 correlation = 0.92 , rmse = 0.49	 correlation = 0.89 , rmse = 0.53
Kinematic	 correlation = 0.94 , rmse = 0.58	 correlation = 0.97 , rmse = 0.32
Muller	 correlation = 0.99 , rmse = 0.25	 correlation = 0.97 , rmse = 0.42

TABLE 4.9: Score Predictions for Gesture 8

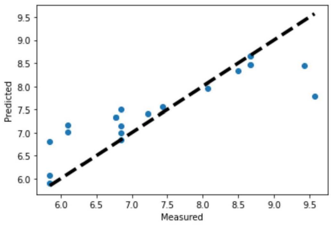
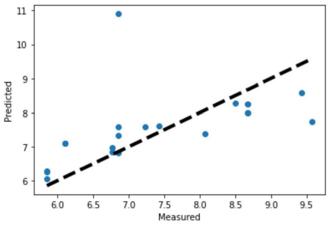
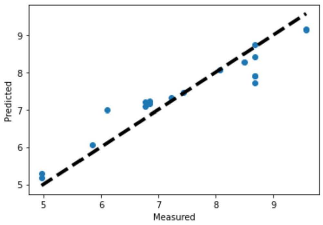
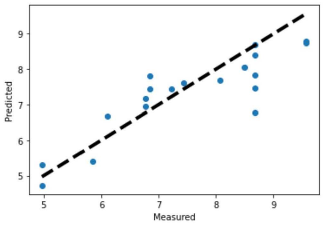
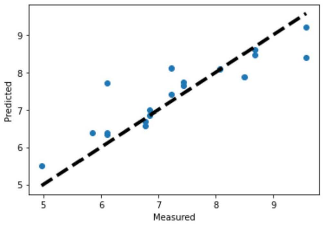
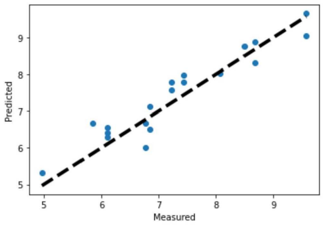
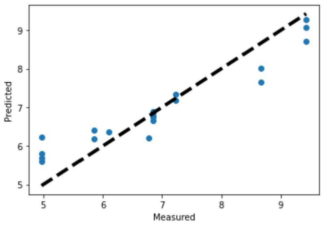
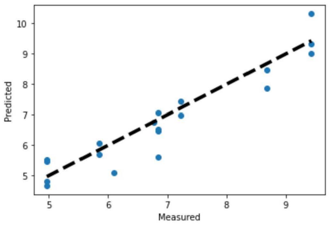
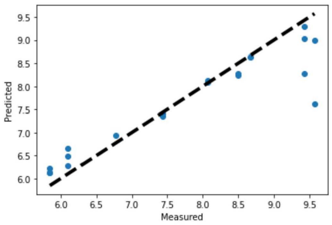
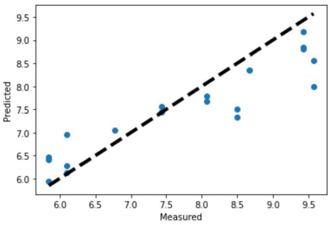
Features	**Random Forest**	**Regression**
Ergonomic	 correlation = 0.86 , rmse = 0.66	 correlation = 0.49 , rmse = 1.15
Euler	 correlation = 0.96 , rmse = 0.46	 correlation = 0.87 , rmse = 0.71
Quaternion	 correlation = 0.90 , rmse = 0.58	 correlation = 0.95 , rmse = 0.42
Kinematic	 correlation = 0.96 , rmse = 0.57	 correlation = 0.94 , rmse = 0.53
Muller	 correlation = 0.93 , rmse = 0.59	 correlation = 0.93 , rmse = 0.67

TABLE 4.10: Score Predictions for Gesture 9

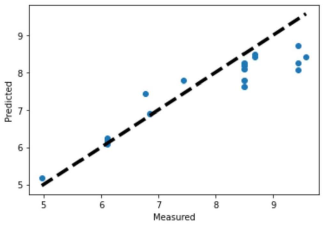
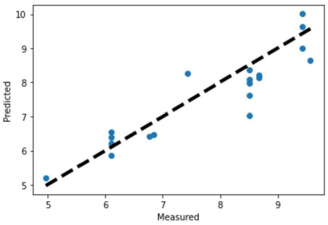
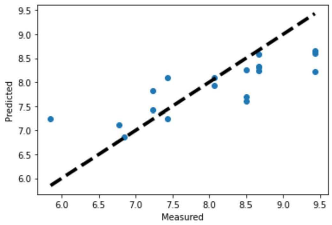
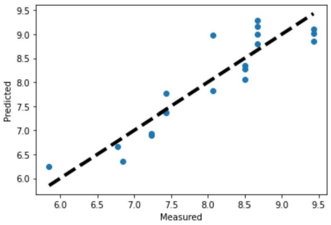
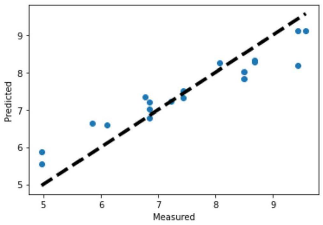
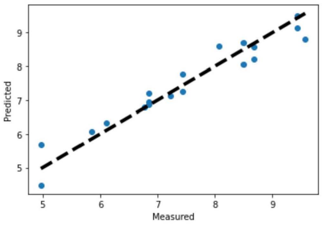
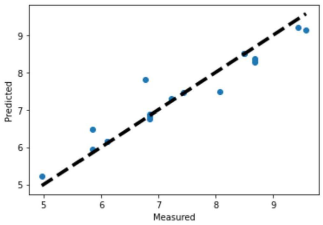
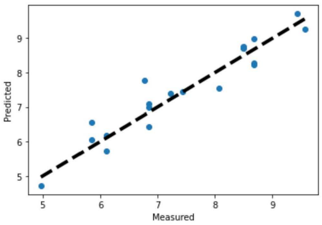
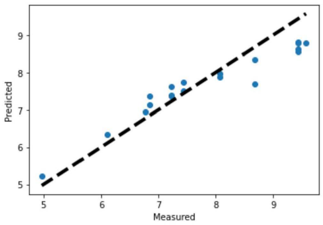
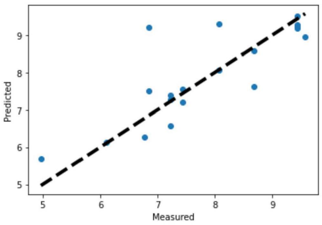
Features	Random Forest	Regression
Ergonomic	 correlation = 0.94 , rmse = 0.63	 correlation = 0.92 , rmse = 0.59
Euler	 correlation = 0.82 , rmse = 0.64	 correlation = 0.91 , rmse = 0.41
Quaternion	 correlation = 0.96 , rmse = 0.53	 correlation = 0.96 , rmse = 0.37
Kinematic	 correlation = 0.96 , rmse = 0.37	 correlation = 0.95 , rmse = 0.40
Muller	 correlation = 0.96 , rmse = 0.50	 correlation = 0.82 , rmse = 0.74

TABLE 4.11: Score Predictions for Gesture 10

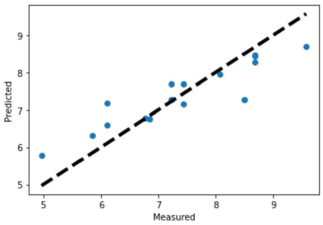
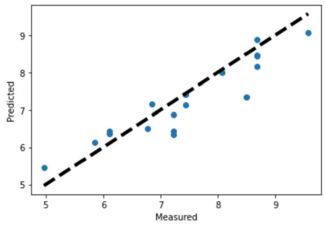
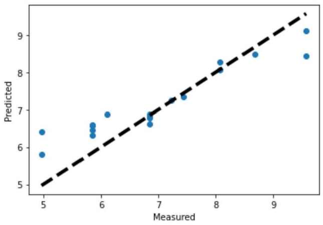
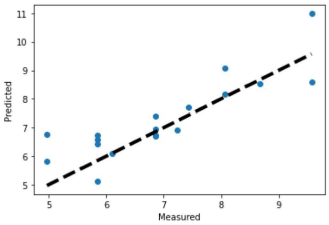
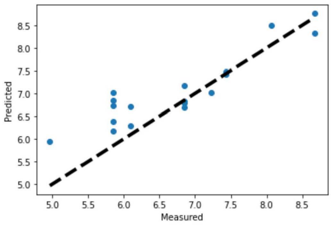
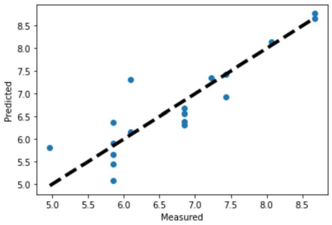
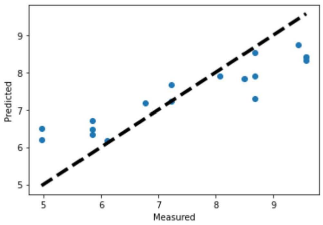
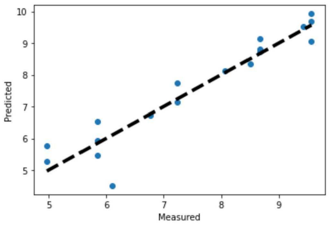
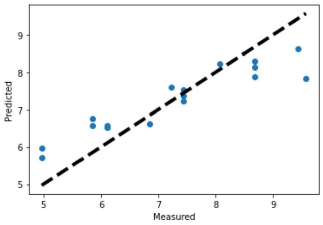
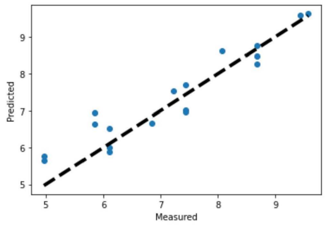
Features	Random Forest	Regression
Ergonomic	 correlation = 0.92 , rmse = 0.54	 correlation = 0.93 , rmse = 0.49
Euler	 correlation = 0.95 , rmse = 0.60	 correlation = 0.86 , rmse = 0.77
Quaternion	 correlation = 0.91 , rmse = 0.55	 correlation = 0.89 , rmse = 0.48
Kinematic	 correlation = 0.93 , rmse = 0.86	 correlation = 0.95 , rmse = 0.52
Muller	 correlation = 0.94 , rmse = 0.69	 correlation = 0.95 , rmse = 0.49

TABLE 4.12: Score Predictions for Gesture 11

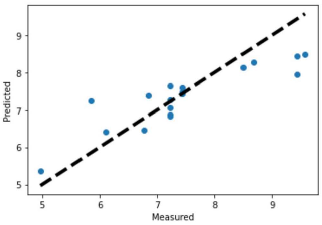
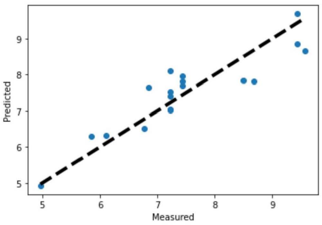
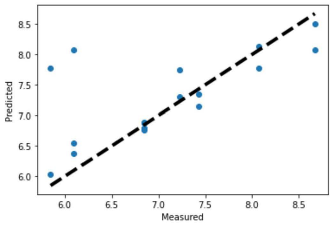
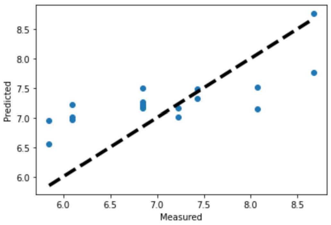
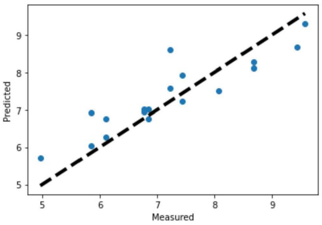
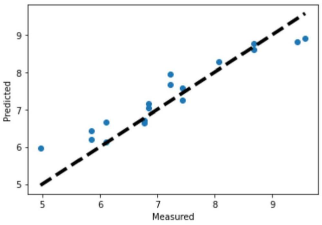
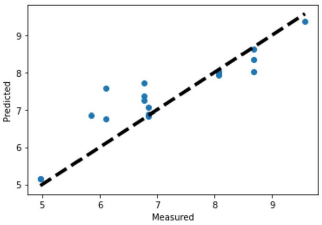
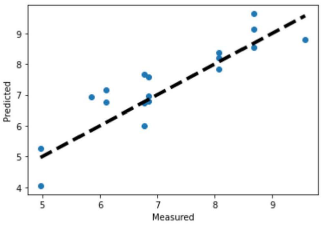
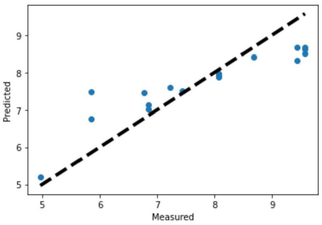
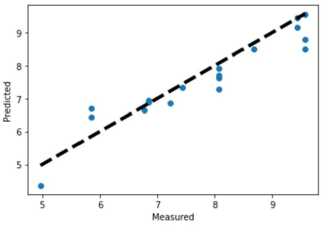
Features	Random Forest	Regression
Ergonomic	 correlation = 0.88 , rmse = 0.66	 correlation = 0.90 , rmse = 0.52
Euler	 correlation = 0.65 , rmse = 0.70	 correlation = 0.75 , rmse = 0.65
Quaternion	 correlation = 0.90 , rmse = 0.58	 correlation = 0.96 , rmse = 0.44
Kinematic	 correlation = 0.92 , rmse = 0.57	 correlation = 0.90 , rmse = 0.64
Muller	 correlation = 0.92 , rmse = 0.71	 correlation = 0.95 , rmse = 0.50

TABLE 4.13: Score Predictions for Gesture 12

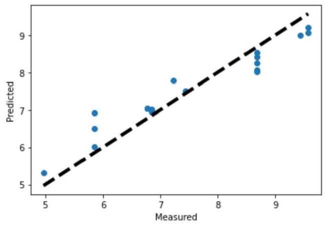
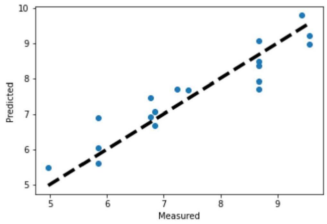
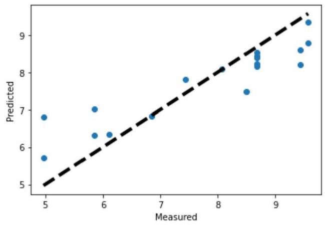
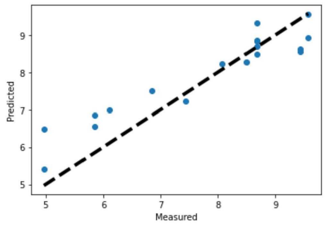
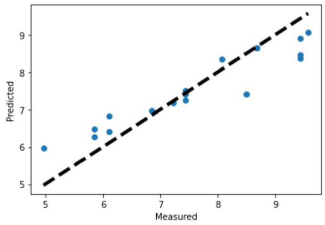
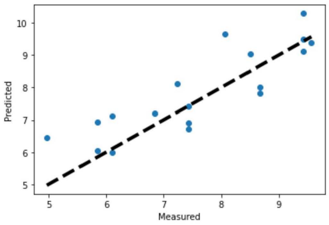
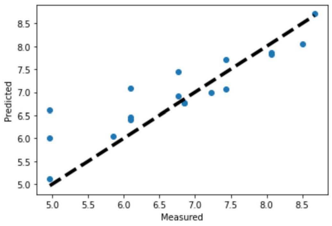
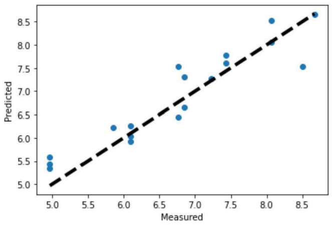
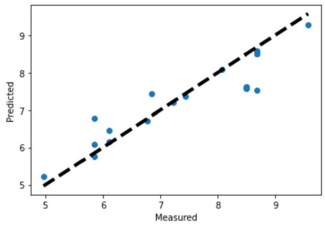
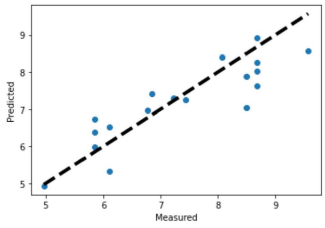
Features	Random Forest	Regression
Ergonomic	 correlation = 0.97 , rmse = 0.46	 correlation = 0.93 , rmse = 0.51
Euler	 correlation = 0.92 , rmse = 0.76	 correlation = 0.94 , rmse = 0.65
Quaternion	 correlation = 0.95 , rmse = 0.58	 correlation = 0.85 , rmse = 0.78
Kinematic	 correlation = 0.89 , rmse = 0.58	 correlation = 0.94 , rmse = 0.42
Muller	 correlation = 0.93 , rmse = 0.50	 correlation = 0.88 , rmse = 0.65

TABLE 4.14: Score Predictions for Gesture 13

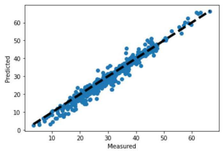
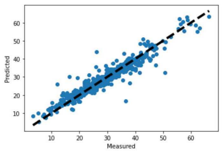
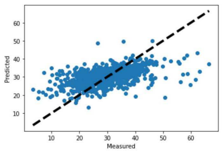
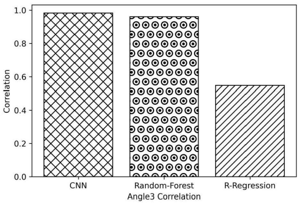
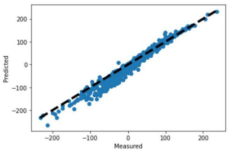
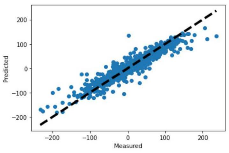
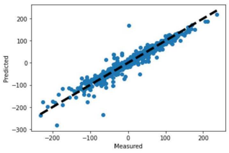
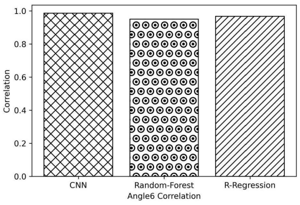
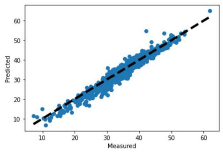
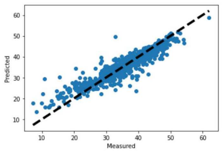
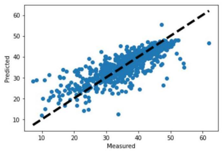
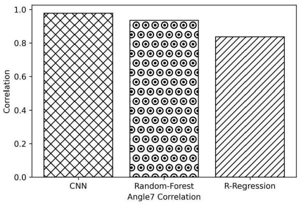
CNN	Random Forest	Regression	Angle Correlation
			
			
			

TABLE 4.15: Motion evaluation with angle criteria

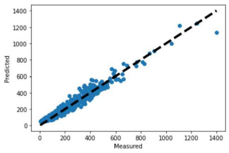
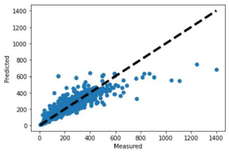
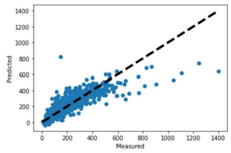
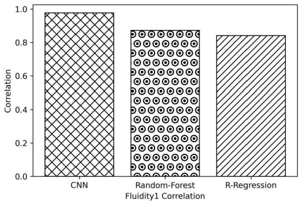
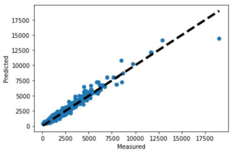
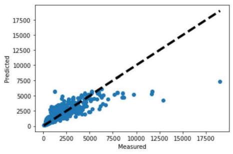
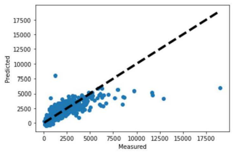
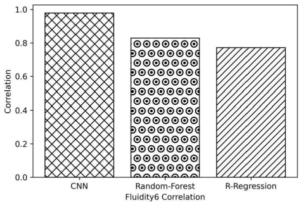
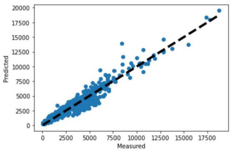
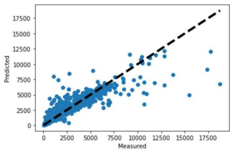
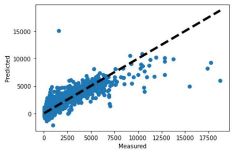
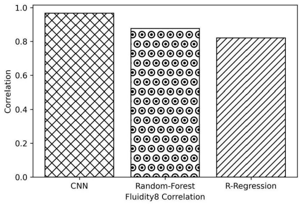
CNN	Random Forest	Regression	Fluidity Correlation
			
			
			

TABLE 4.16: Motion evaluation with fluidity criteria

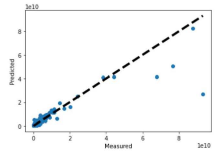
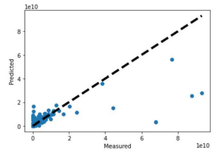
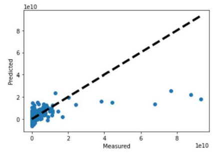
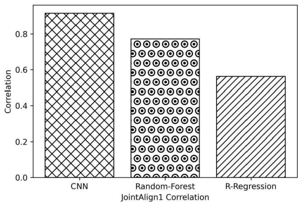
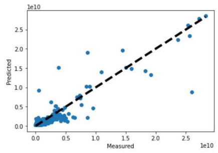
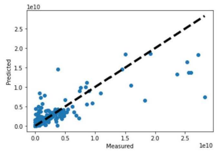
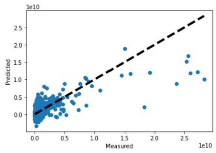
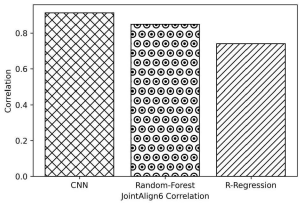
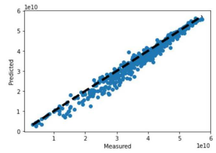
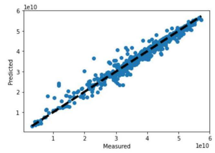
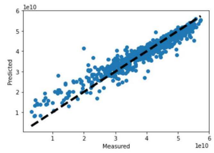
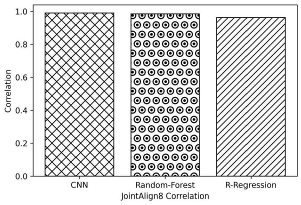
CNN	Random Forest	Regression	JointAlign Correlation
			
			
			

TABLE 4.17: Motion evaluation with joint align criteria

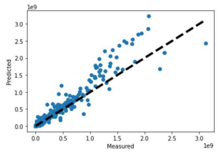
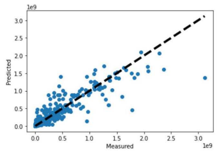
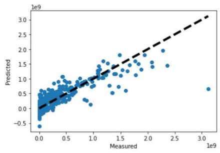
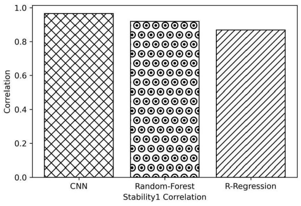
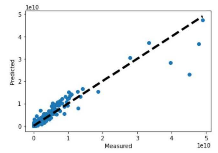
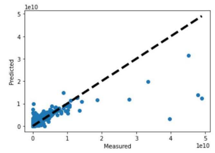
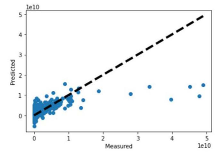
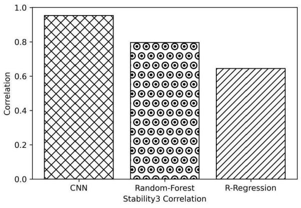
CNN	Random Forest	Regression	Stability Correlation
			
			

TABLE 4.18: Motion evaluation with stability criteria

4.6 Limitations

The motion evaluation part limitation is that it is a supervised method and needs an annotated dataset by experts to be able to evaluate the actions. Not having access to such knowledge or not having enough amount of annotated data makes the use of data demanding methods such as deep learning difficult. That is why our improvement to 99 percent correlation and low error by just using the simple and traditional machine learning methods is valuable. Another way of tackling this problem is automatic annotation.

4.7 Summary

In this chapter, several contributions were made to improve the accuracy and effectiveness of human motion estimation and evaluation. The primary focus was on **human motion evaluation** using different machine learning approaches, particularly comparing traditional machine learning models (Random Forest and Ridge Regression) with deep learning models (CNNs). These models were applied to human pose data obtained from the **SMPL model**, which provides a more detailed and accurate representation of human motion.

One of the key contributions of this work is the **Human Motion Evaluation**, which aims to improve the quality and accuracy of motion analysis by assessing the postures and movements of humans after motion estimation. This evaluation was conducted using a variety of motion features, including **joint rotations (SMPL pose parameters)**, which were directly fed into the machine learning models without any feature extraction.

The **Human Motion Evaluation** was validated using several datasets, including **AMASS**, a large-scale benchmark dataset annotated for motion features such as **desirable joint angles**, **motion fluidity**, **joint alignment**, and **movement stability**. These features were selected based on mathematical formulas representing the desired qualities of good human motion. The annotations for each feature category were manually prepared using expert knowledge or mathematical definition, and the evaluation was performed on both **martial arts** and **general human motion** datasets, offering a comprehensive assessment across various motion types.

4.8 Comparison of Features and Models for Classic Machine Learning and Martial Arts Dataset

The input features used for the **classic machine learning models** (Random Forest and Ridge Regression) were diverse and represented different motion characteristics:

- **Ergonomic Features**: These features focus on human biomechanics, including joint angles, limb lengths, and other anatomical parameters that represent the physical constraints of human movement.

- **Euler Angles**: These angles describe the orientation of joints and body segments in 3D space using rotation matrices.

- **Quaternions**: A more compact and computationally stable representation of rotations compared to Euler angles.

- **Kinematic Features**: These features describe the motion of the human body in terms of velocity, acceleration, and the movement of joints over time.

- **Müller Features**: This refers to a set of higher-order motion descriptors that capture the geometry of joint motion and the smoothness of transitions.

After evaluating different feature sets and machine learning models, it was found that the combination of the following performed the best in terms of motion evaluation accuracy:

-Best Feature Combination: The combination of **kinematic features** (which include velocity, acceleration, and joint movement over time) and **ergonomic features** (which represent the anatomical constraints and joint angles of the human body) showed the best performance for **Random Forest**. These features provided complementary information that allowed the model to effectively evaluate both the dynamic and static aspects of human motion.

-Best Machine Learning Model: Among the classic machine learning models, **Random Forest** consistently outperformed **Ridge Regression**. Random Forest's ability to handle non-linear relationships in the data and its robustness to overfitting made it particularly effective for motion evaluation, especially when combined with kinematic and ergonomic features. It was the top performer for tasks such as **joint alignment** and **desirable angles**.

Therefore, the **Random Forest model combined with kinematic and ergonomic features** achieved the best performance in motion evaluation. The model demonstrated superior accuracy and correlation with expert-annotated data, achieving higher performance than Ridge Regression in nearly all evaluation criteria.

4.9 Machine Learning Model Comparison for Deep Learning and General AMASS Dataset

The results from the experiments demonstrated that **deep learning (CNN)** outperformed traditional machine learning models (Random Forest and Ridge Regression) in all categories, especially for more complex motion characteristics, such as **fluidity** and **stability**. In contrast, **Random Forest** showed strong performance, particularly for features involving **joint alignment** and **desirable angles**, although it is still behind CNN in terms of performance. **Ridge Regression** generally performed the least well across all the criteria, especially in more challenging evaluations of **fluidity** and **stability**.

- **Random Forest**: This method showed strong performance when combined with **kinematic** and **ergonomic features**, particularly for evaluating **joint alignment** and **desirable angles**. The combination of these features with Random Forest resulted in the best performance among the traditional machine learning models.

- **Ridge Regression**: **Ridge Regression** had the least performance overall, particularly in the evaluation of **fluidity** and **stability**. It is a linear model, which limits its ability to capture the complex, nonlinear patterns that exist in human motion data.

- **Deep Learning (CNN)**: **CNNs** demonstrated the best performance across all motion evaluation categories. They were particularly effective at capturing the complexities of **fluidity** and **stability** that the traditional methods struggled with. The **CNN** model could learn complex representations of human motion directly from the raw joint rotation data, outperforming both **Random Forest** and **Ridge Regression** in almost every category.

4.10 Key Findings and Contributions

:

- **Improved Motion Evaluation**: The proposed method for human motion evaluation using the SMPL model significantly enhanced the accuracy of motion scoring, especially when combined with better feature selection and the application of deep learning models.

- **Feature Selection**: The evaluation demonstrated that a combination of **kinematic features** and **ergonomic features** were the most effective for traditional machine learning models. **Euler angles** and **quaternions** showed weaker performance but still contributed useful information to the evaluation.

- **Machine Learning Model Comparison**: The comparison between **Random Forest**, **Ridge Regression**, and **CNN** revealed that deep learning models were superior when dealing with raw motion signals, especially for complex features like **fluidity** and **stability**. **Random Forest** performed well on features such as **joint alignment** and **desirable angles**, when combined with **kinematic** and **ergonomic features**, while **Ridge Regression** struggled in most cases.

- **Performance Validation**: The methods were validated using the **AMASS dataset** and other datasets containing martial arts and general human motion, demonstrating the ability to evaluate diverse types of human motion effectively. The evaluation metrics, including **correlation** and **RMSE**, showed that CNNs consistently outperformed traditional methods across most of the criteria.

- **Improved Evaluation Accuracy**: By utilizing **deep learning models** with raw motion data and enhancing the feature selection process, significant improvements were achieved in human motion evaluation, surpassing the results from previous works that utilized similar datasets. This was achieved without the need for complicated feature engineering or hand-crafted features, showcasing the strength of deep learning models in human motion analysis.

- **Contributions to Motion Evaluation**: The main contribution of this chapter is the **Human Motion Evaluation**, which significantly enhanced the accuracy of motion scoring. This method uses joint rotations from the SMPL model, avoiding the need for extensive feature extraction, and uses machine learning models (both classic and deep learning) to assess the quality of human motion. The method was validated across several datasets and proved effective in improving evaluation accuracy. For classic machine learning effectiveness of feature selection combined with choice of appropriate machine learning model improved the motion evaluation error significantly.

Overall, this chapter contributes to the field by advancing human motion evaluation through both **improved machine learning methods** and the use of more accurate human body representations (SMPL model). The **Human Motion Evaluation** demonstrated clear benefits when using deep learning, achieving **high correlation scores** (up to 99%) and low error rates across various motion evaluation categories. The results shows the power of deep learning in human motion analysis, while also highlighting the limitations of traditional methods when dealing with complex motion dynamics like fluidity and stability.

Chapter 5

Conclusions

This chapter will conclude the study by summarising the key research findings in relation to the research aims, as well as the value and contribution thereof. It will also review the limitations of the study and propose opportunities for future reserach.

5.1 Research Summary

This research focused on three main stages of inferring and analysing the 3D human motion data from monocular video. These stages are capturing data and producing the dataset, human motion estimation and human motion evaluation. We aimed at improving accuracy and solving issues of baseline methods in motion estimation and motion evaluation by various means. Baseline human motion estimation methods are improved in terms of self-occlusion and unnatural poses. Improvements to the baseline motion estimation and evaluation methods are listed in the following paragraphs.

5.1.1 Human Pose Estimation

In the human pose estimation problem, instead of using the more common 3D human kinematic skeleton structure we used more accurate adaptive shape modeling that infers 3D parametric human models and is called SMPL.

The input data of the pose estimation system is monocular video footage which causes limitations on the resulting 3D inferences. Being view-independent and robust to occlusion is usually a common goal. We have addressed these limitations by adding another post-processing step after the pose estimator to compensate for such biases which maps incorrect poses/motions caused by self-occlusion to correct poses/motions.

Traditional online benchmark datasets for 3D human pose estimation tend to have 3D key points as their ground truth. Though only a handful of datasets have been created using the SMPL human model as the ground truth. In my thesis, I have focused on improving the estimation of self-occluded motions using SMPL models. I have also created a data set of self-occluded motions that were performed by three subjects whilst using the SMPL model as the ground truth. These data included monocular video capture and MoCap as the ground truth. Additionally, I also introduced a new error measurement method to evaluate the correct pose using SMPL rotation parameters.

The transition between predicted poses between each frame is not usually smooth. Use of different signal processing filters showed that many low-pass filters can give a good result.

The current accuracy metric for human pose estimation is the average 3D joint error between the prediction and ground truth. While this can be an indicator of relative quality of a specific method, it cannot provide more detailed explanation about which frames had better or worse predictions or if there is a common issue in all predicted poses in a sequence. The action specific error measurements and qualitative results can help to find the issues in pose estimation.

To enhance motion continuity and positional accuracy especially in the case of occlusion where the errors are high, two main post-processing techniques were applied: machine learning-based post-processing (using Random Forest and LSTM AutoEncoders) and inverse kinematics (IK). Each approach contributed differently to improving the SMPL pose predictions and was evaluated for effectiveness under various conditions.

5.1.1.1 Machine Learning-Based Post-Processing

The first post-processing approach utilized machine learning models, specifically frame-to-frame Random Forest models and motion-to-motion LSTM AutoEncoders, to refine pose predictions. Random Forest models were tuned iteratively for optimal parameters using a validation-based random search. This model excels in reducing positional noise on a frame-by-frame basis, allowing it to correct sudden shifts or outliers in joint positions without sacrificing temporal continuity. However, the LSTM AutoEncoder model was particularly effective in temporal smoothing, as its recurrent structure captured sequential dependencies in the motion data, resulting in smoother and more realistic transitions between frames. This model proved beneficial for complex poses involving significant limb overlap, where individual frame corrections alone may be insufficient.

The results of applying these machine learning models showed a marked reduction in both 3D joint positional errors and rotational discrepancies when compared to baseline predictions. Furthermore, in experiments using cross-validation, both models improved prediction accuracy, though the LSTM model displayed a greater capacity for long-sequence consistency, which is crucial in self-occluded actions. The RF method had access to the current frame motion data and showed better accuracy compared to the predictive autoencoder method.

5.1.1.2 Inverse Kinematics Post-Processing

Inverse kinematics (IK) was used as a secondary post-processing approach to improve SMPL model accuracy, especially for unnatural or challenging poses such as handstands and cartwheels. IK leveraged predicted 3D key points, which, while not precise in orientation, provided accurate spatial targets for limb endpoints (e.g., wrists and feet). IK improved joint position accuracy significantly, especially for upper body poses, as hands served as precise targets in cases of occlusion.

However, the results demonstrated a trade-off: while IK reduced positional errors, the orientation accuracy of joints (particularly rotational alignment) decreased due to the IK model's reliance solely on position data, without rotational targets. This limitation was evident in increased rotation errors, especially for limb movements requiring accurate angle alignment (e.g., arms in overhead motions).

5.1.1.3 Performance Analysis and Comparison

The two post-processing methods addressed different facets of the pose estimation problem. Machine learning-based post-processing offered improvements in both position and rotation accuracy and performed particularly well on actions with consistent motion patterns. Meanwhile, IK proved advantageous in refining joint positions in complex or self-occluded motions, with noticeable reductions in hand and arm positional errors. However, due to the lack of orientation control, IK was less effective in scenarios requiring strict rotational precision.

Overall, combining both methods demonstrated the potential for a complementary approach, where machine learning could handle frame-to-frame consistency and temporal smoothing, while IK could be selectively applied to refine endpoint joint positions in cases of extreme occlusion. The validation results across varied datasets support the effectiveness of these post-processing techniques in enhancing SMPL-based pose estimation accuracy, particularly in self-occluded and complex motion scenarios.

If the post-processing model is trained on a specific action, it performs better compared to the model trained on all the actions of the dataset. Action-specific modeling is introduced to leverage action recognition as to select the appropriate model trained on specific action. This provides a tailored approach for reconstructing self-occluded motions.

5.1.2 Human Motion Evaluation

The goal of human motion evaluation in this research is assigning a score value to the subject's movement. The motion sequence length can be as small as a gesture or a more complicated movement. We have used both classic and deep learning methods to evaluate a movement and compared their error and correlation values.

The classic machine learning methods using explicit feature extraction was implemented. We have found that adding different feature categories together does not help in better prediction of the output score. This causes a very large feature space that later should be reduced by methods such as PCA. Instead, we have used only one category of features for the model and used better classifiers such as random forest which improves the prediction result compared to the baseline method.

Separate use of each feature category also helps to compare the effectiveness of each feature and classifier in prediction of the score value. In a model trained on all gestures, the random forest performed better than regression and among the features Kinematic and Muller gained the best performance. We have also trained a separate model for scoring each gesture. When training on specific gestures, the highest achievable accuracy is usually lower compared to when we train the model on all gestures. The choice of the best performing feature extraction and classification method varies depending on the type of gesture.

The dataset used for classic motion evaluation [416] was relatively small and its 3D human data was in the form of a 3D kinematic skeleton. The dataset ground truth score values were labelled visually by some experts. To check the performance of deep neural networks for motion evaluation, a larger data set was needed. The format of the 3D human data should also be compatible with the pose estimator framework i.e. 3D human shape model such as SMPL. For this purpose, an unlabelled large dataset of SMPL motions [106] was labelled using mathematical quality metrics.

Comparing the result of motion evaluation using deep learning (CNN) and classic machine learning methods showed that the CNN model performed generally better than classic methods. Similar to the previous dataset, random forest prediction is better than regression method. The joint rotations of the SMPL model proved to be a good input for both random forest and CNN methods without the need for extra feature extraction.

5.2 Limitations and Future Research

5.2.1 Data Collection

The motion capture system in the lab consists of the optical MoCap cameras and a synchronised video camera that can record with the same high frame rate of 120 fps. Other sensors such as Kinect or multi-camera system can also be synchronised and connected to the existing motion capture system providing richer datasets and optimized ground truth through fusion of the available data.

Synchronization between multiple video cameras, and between motion capture and a single video camera, is done using hardware synchronization which has high precision. When this is not practical due to hardware limitations, it is possible to investigate synchronising by other means for example matching the 3D motion wave patterns resulting from the two systems or using sound.

5.2.2 Human Motion Estimation

In the human pose estimation, the predicted SMPL model can be rotated along one of the axes. While the results are aligned together to measure the error between joint prediction, the effect of such incompatibility in the coordinate axis can be further investigated.

The predicted human motion in SMPL format lacks global position prediction, therefore only the local body motion is tracked. More experiments can be done for inferring the global position of the subject in the 3D environment.

The predicted camera parameters from the pose estimator are parameters for a weak perspective model and used for orthographic projection of predicted 3D joints to predicted 2D joints and then comparing it with the 2D ground truth. It is possible to predict the full camera parameters using deep learning and use it to improve the result.

The loss function for the 3D human pose estimation method can consider the difference between SMPL parameters, 2D and 3D joint positions. An improved loss function can result in better final predictions.

In terms of the available 3D training data, there is an inconsistency in the structure of the ground truth 3D skeleton annotations between different available datasets. To be able to use the datasets together, it is good to use a standard format such as the SMPL model. Other available datasets can also be processed to assign a ground truth with the common format.

As mentioned before, many current 3D pose estimation methods are motivated by lack of 3D ground truth. Improving such models can be investigated by use of strong 3D supervision as opposed to using weak supervision by 2D joint locations along with small 3D datasets.

It should be noted that the place of key points in the 2D pose estimators such as Openpose sometimes does not match the corresponding position on the SMPL model, e.g. in the hip joints. This will have an effect on different stages of the HPE method. For example when reconstructing a 3D SMPL model from 2D joints prediction for creating multi-camera ground truth or when computing the 2D re-projection error in the pose estimator loss function.

The correspondence between the various 3D human models also should be taken into account when producing ground truth using MoCap data (SMPL-X). Precise conversion between SMPL-X and SMPL models requires re-computing the SMPL model not only choosing corresponding joints.

In capturing the data in the lab environment, the subjects used the black MoCap suit and a green screen is set up behind the subject. This might be different from the normal clothing in the training data of the video pose estimator. The effect of using different background and clothing on the result of the 3D pose estimator and ways to improve it can be investigated.

5.2.3 Human Motion Post-Processing

In human motion correction, we can denoise the output of the human pose estimator. The 3D pose estimator can predict similar joint rotations with different signs in x, y and z elements of the axis-angle representation. This change of sign in the elements is not visible in the human motion but causes rotation of the character in a few filtered frames while using our windowed noise filtering methods. The input motion can be pre-processed to cancel the effect of such an inconsistency between consequent frames, before using noise filtering methods.

In using machine learning for prediction of correct poses, the dataset of repeating similar actions by different people is needed. The amount of these data can be increased and other learning methods such as deep learning can be investigated.

The 2D pose estimator is used in different parts of the 3D pose estimator. In the SMPL estimation part, the initial human detection and tracking uses 2D pose estimation. In the 3D skeleton estimation part, the intermediate step of finding 2D joints is using the 2D pose estimation. While we have focused on improving self-occlusion in the 3D result in the final steps, it is good to remove the effect of occlusion on the 2D pose estimator too. We can use an improved 2D pose estimator which is robust to occlusion and see how it affects better 3D human prediction.

5.2.4 Human Motion Evaluation

In human motion evaluation, we have worked with supervised learning for predicting a score. In supervised learning the role of the annotated dataset is important. Annotation usually is done by experts for example in sports, healthcare or medical fields. Use of unsupervised learning for evaluating the motion can be further investigated with more focus on the specific application.

When creating annotation for motion evaluation purposes, we have assigned a number related to motion quality of a motion sequence. Various scoring criteria are used for automatic annotation. The annotation values can be normalised for example between 0 and 10 for easier interpretation.

We have used local joint rotations of the 3D SMPL human model as an input of the motion analysis. Unlike this method, use of 3D coordinates of the joints needs normalisation and scaling of the skeleton data. Simple normalisation of such data will cause unwanted change in motion features and is not recommended. The effect of using different motion standardization between different subjects on better human motion evaluation can be further investigated.

Production of synthetic data in the graphical environment using the realistic motions from MoCap and making the appearance of the avatar more realistic can also be investigated. It is also possible to use motion transfer and the SMPL model motion on a more realistic avatar in a photo-realistic 3D environment. This can be used to create more data related to the application area of the project.

Appendix A

Figures and Charts

TABLE A.1: HPE Results on Gymnastics Bar Motions

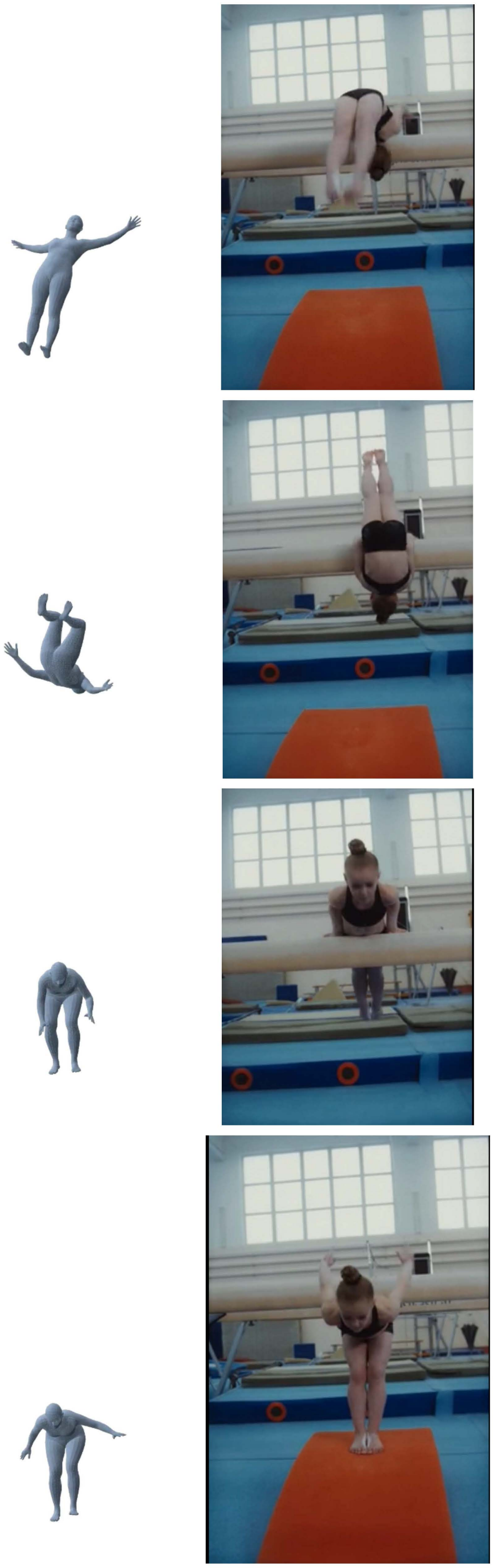


TABLE A.2: HPE Results on Gymnastics Ring Motions

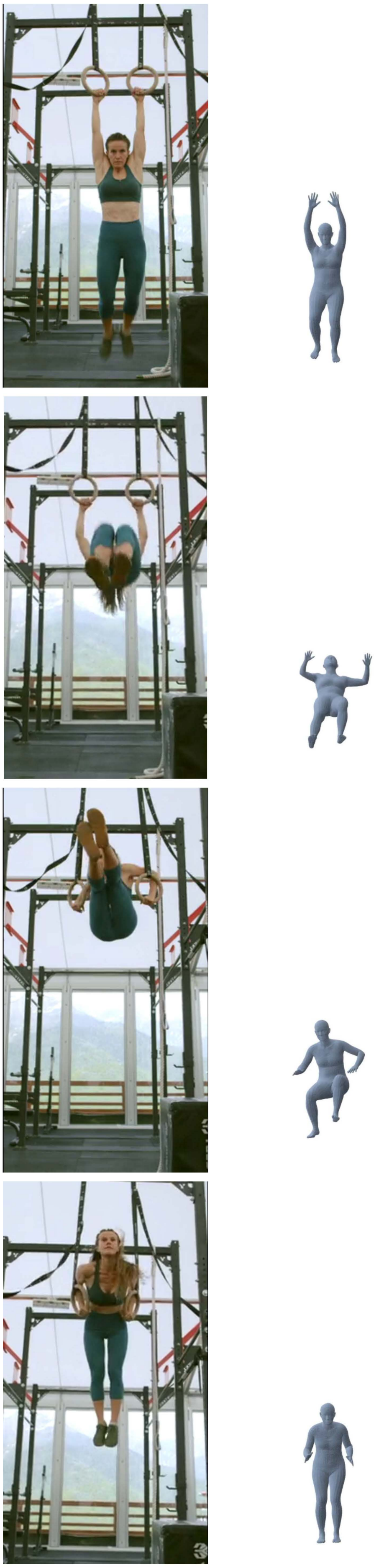


TABLE A.3: HPE Results on Exercise Motions

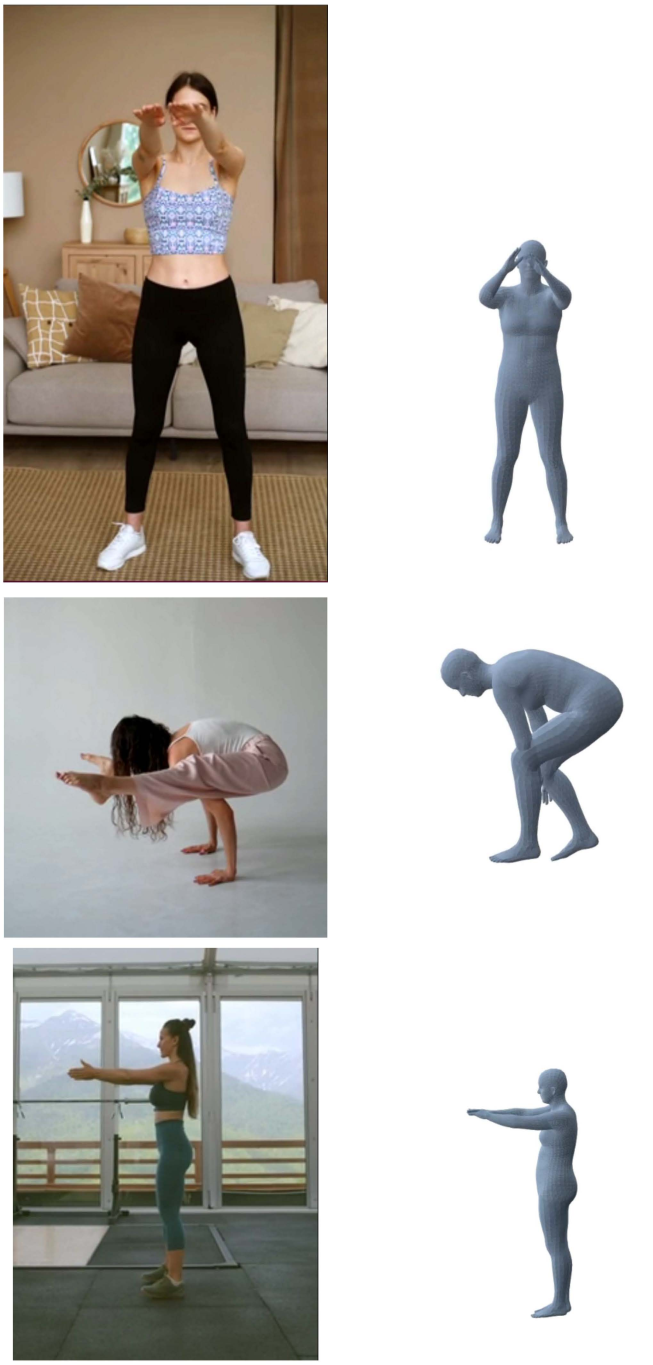


TABLE A.4: HPE Results on Handstand Motion

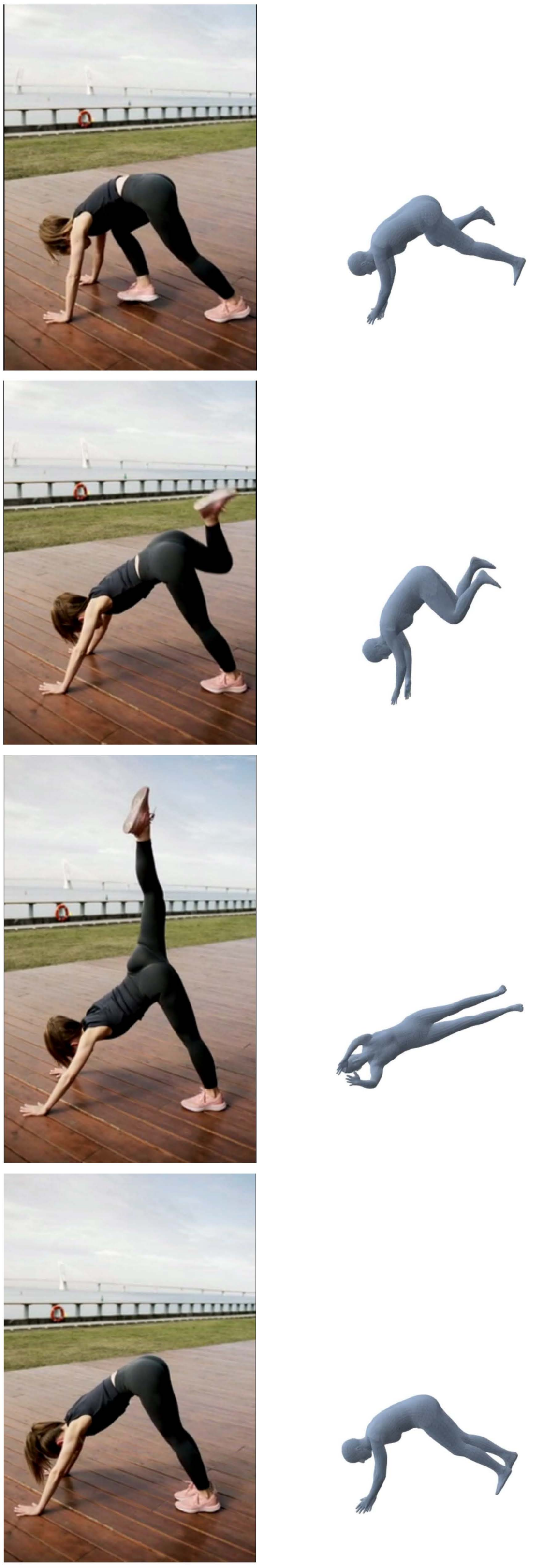


TABLE A.5: HPE Results on Handstand Motion

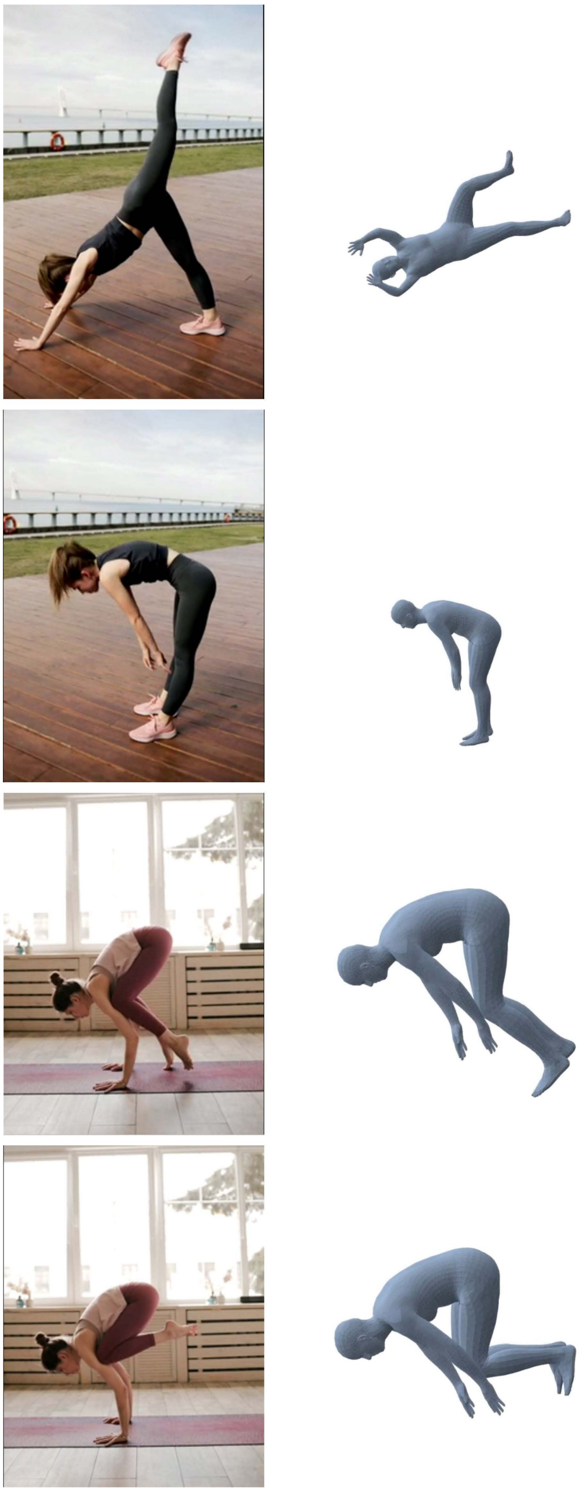


TABLE A.6: HPE Results on Yoga Motions

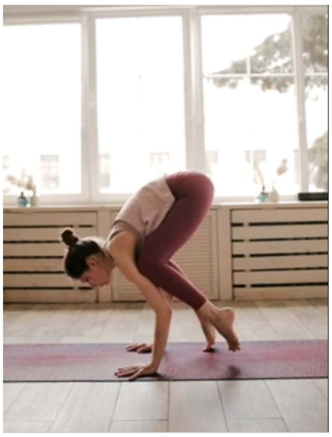
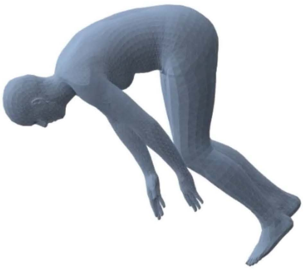
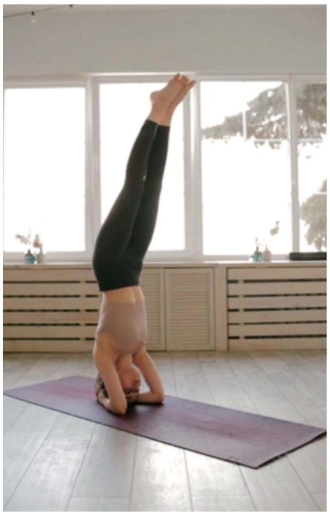
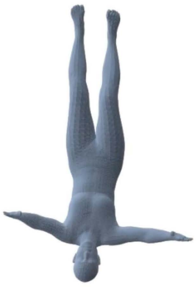
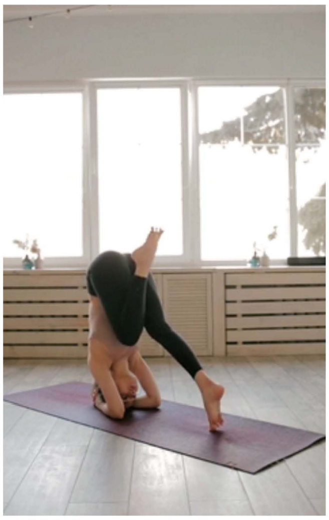
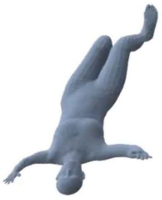
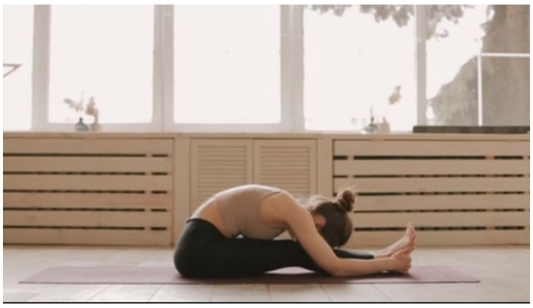
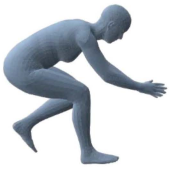
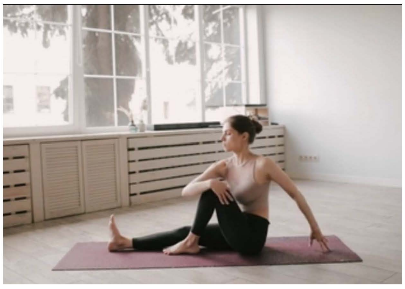
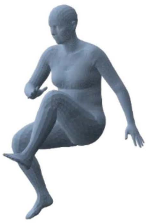
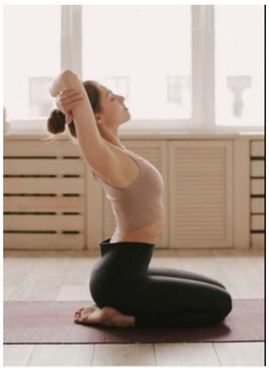
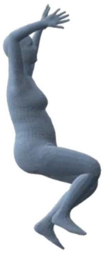
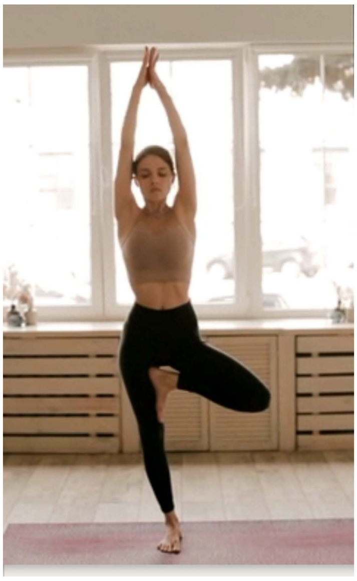
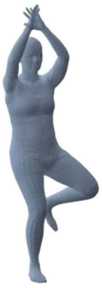
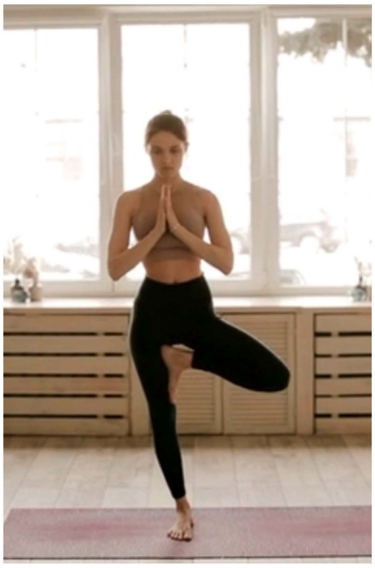
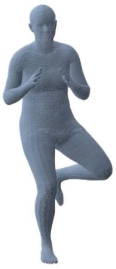
	
	
	
	

TABLE A.7: HPE Results on Yoga Motions

A.1 Deep Learning Extended Results

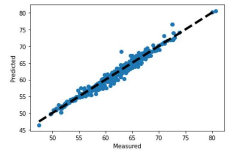
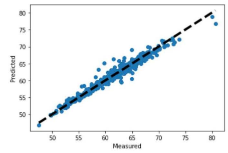
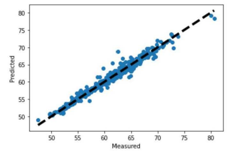
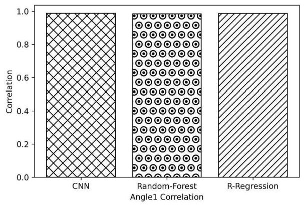
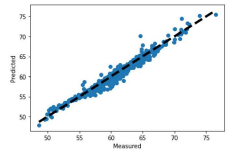
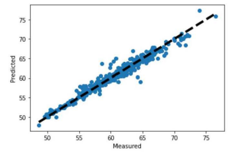
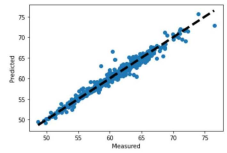
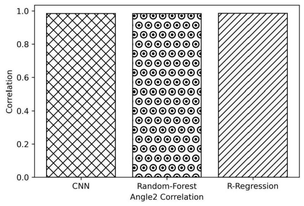
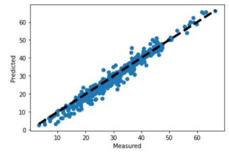
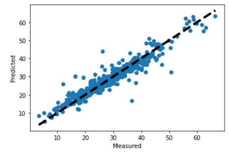
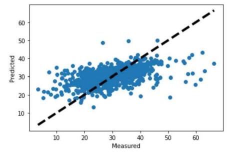
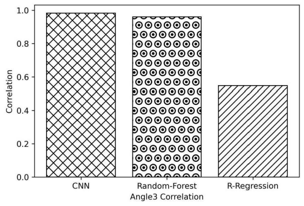
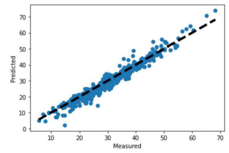
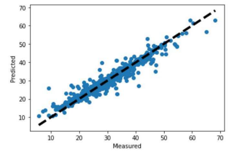
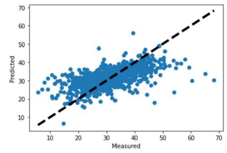
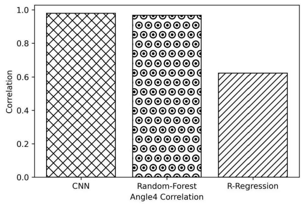
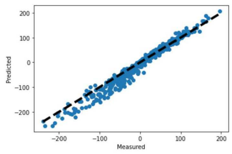
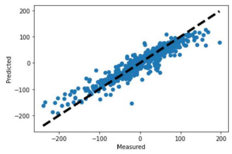
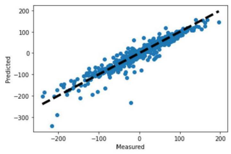
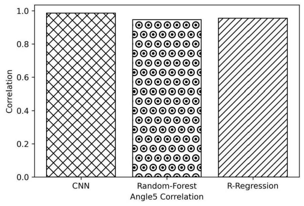
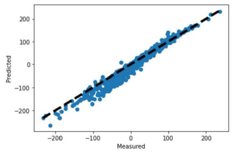
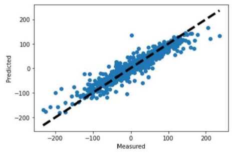
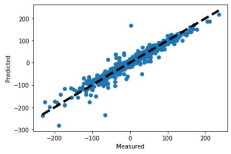
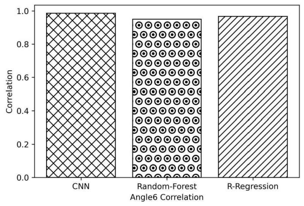
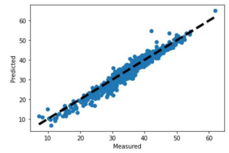
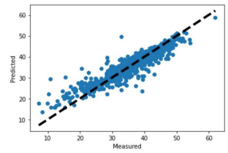
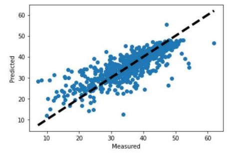
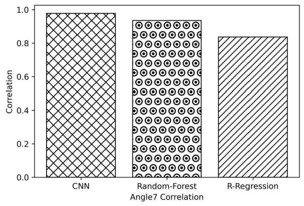
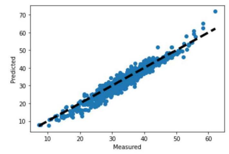
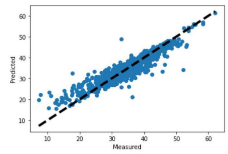
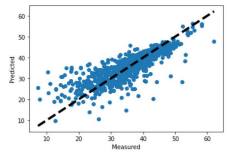
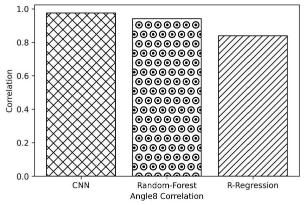
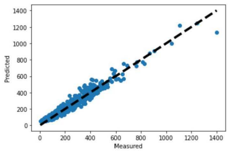
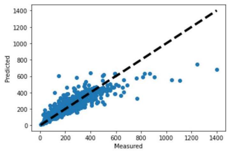
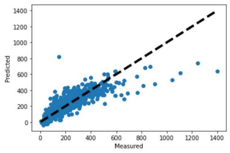
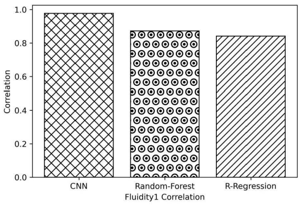
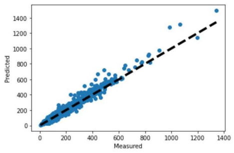
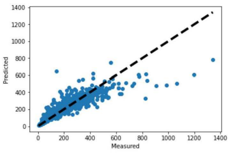
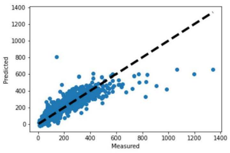
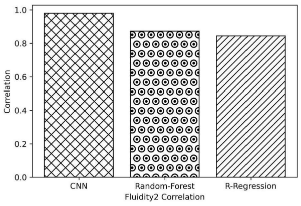
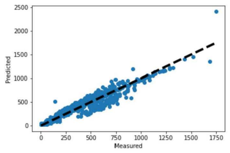
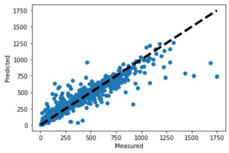
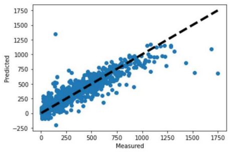
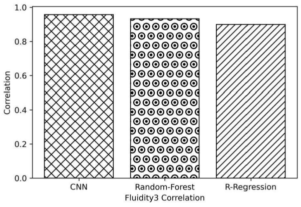
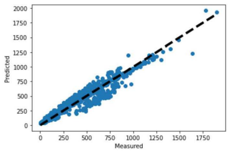
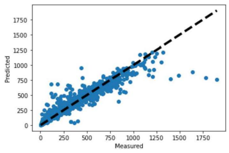
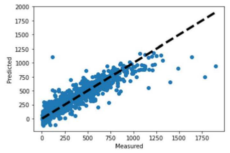
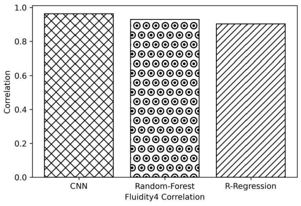
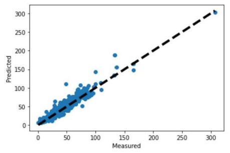
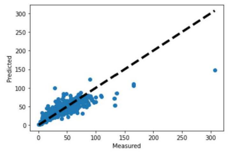
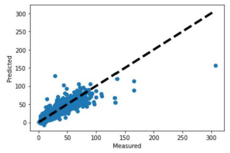
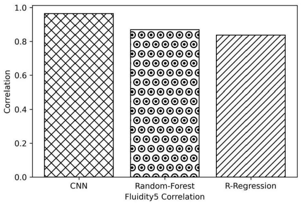
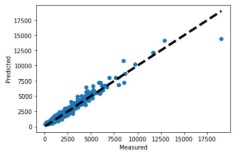
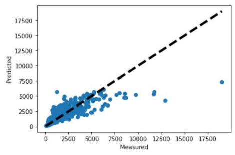
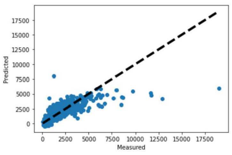
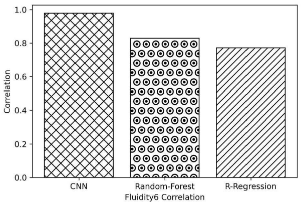
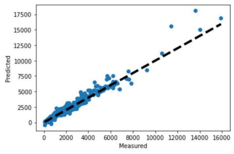
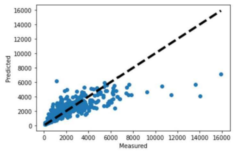
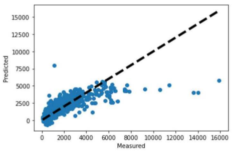
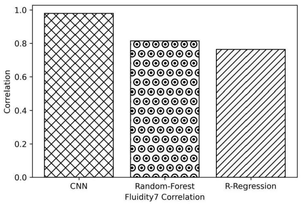
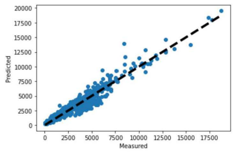
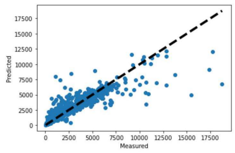
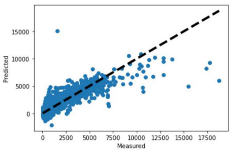
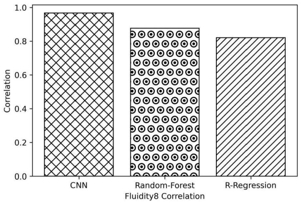
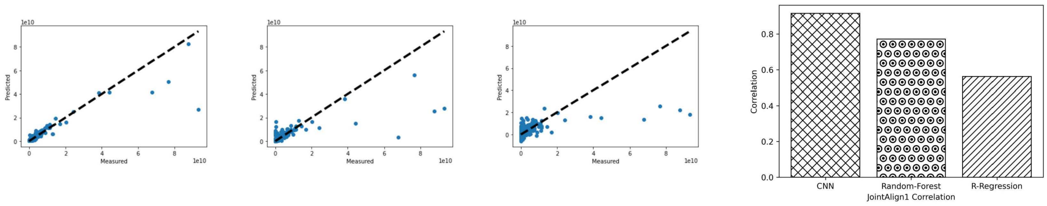
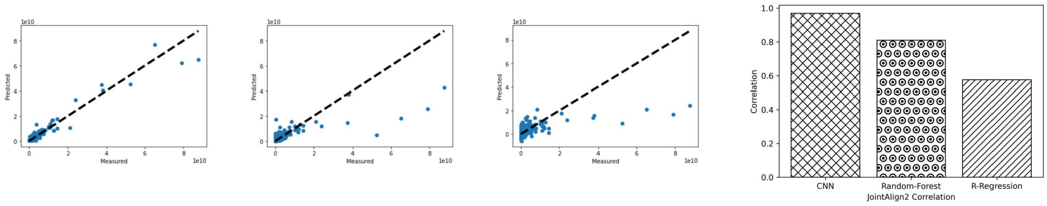
CNN	**Random Forest**	**Regression**	**Angle Correlation**
			
			
			
			
			
			
			
			

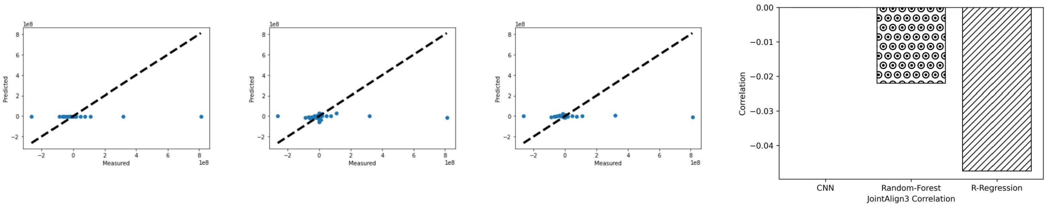
TABLE A.8: Motion evaluation with angle criteria

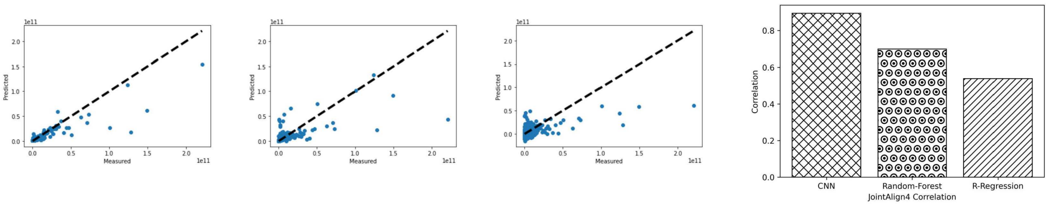
CNN	Random Forest	Regression	Fluidity Correlation
			
			
			
			
			
			
			
			

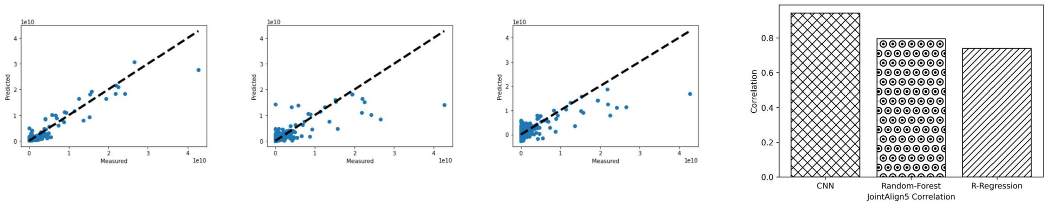
CNN | **Random Forest** | **Regression** | **JointAlign Correlation**

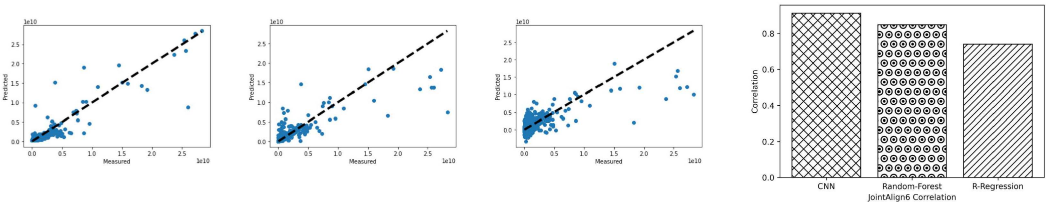


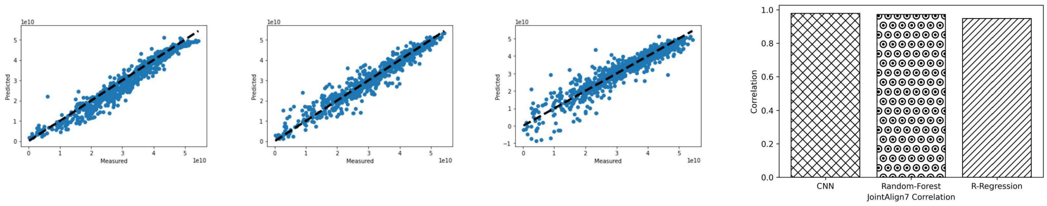


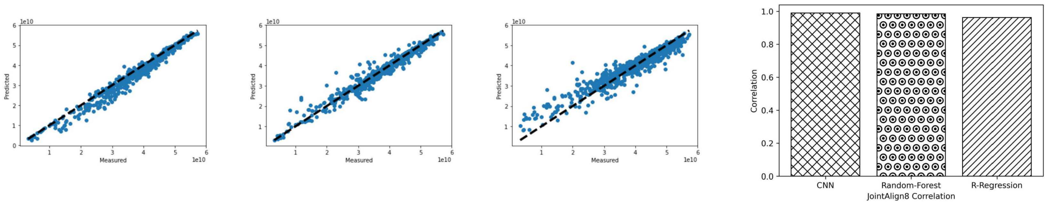


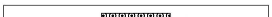












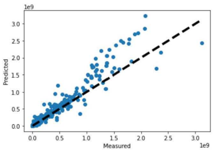
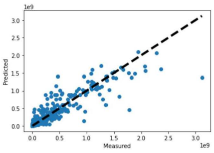
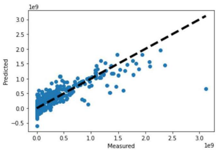
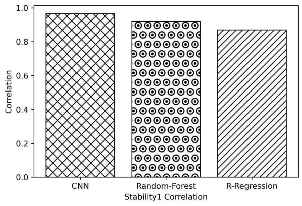
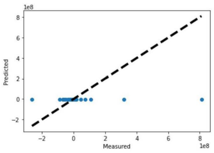
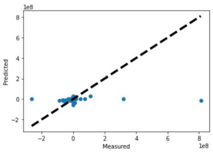
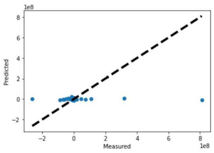
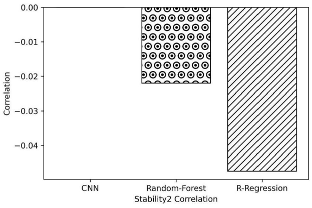
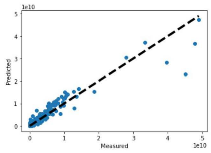
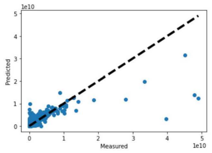
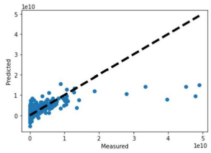
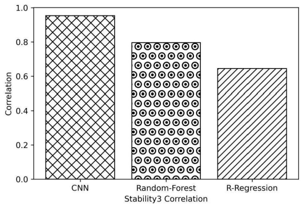
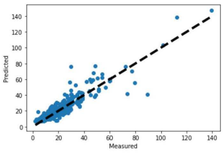
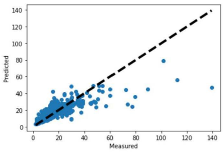
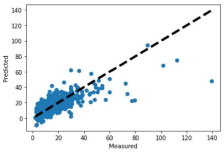
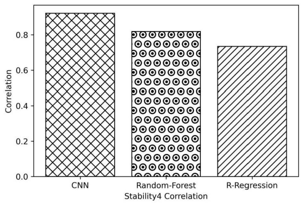
CNN	Random Forest	Regression	Stability Correlation
			
			
			
			

TABLE A.11: Motion evaluation with stability criteria

A.2 Classic Machine Learning Features

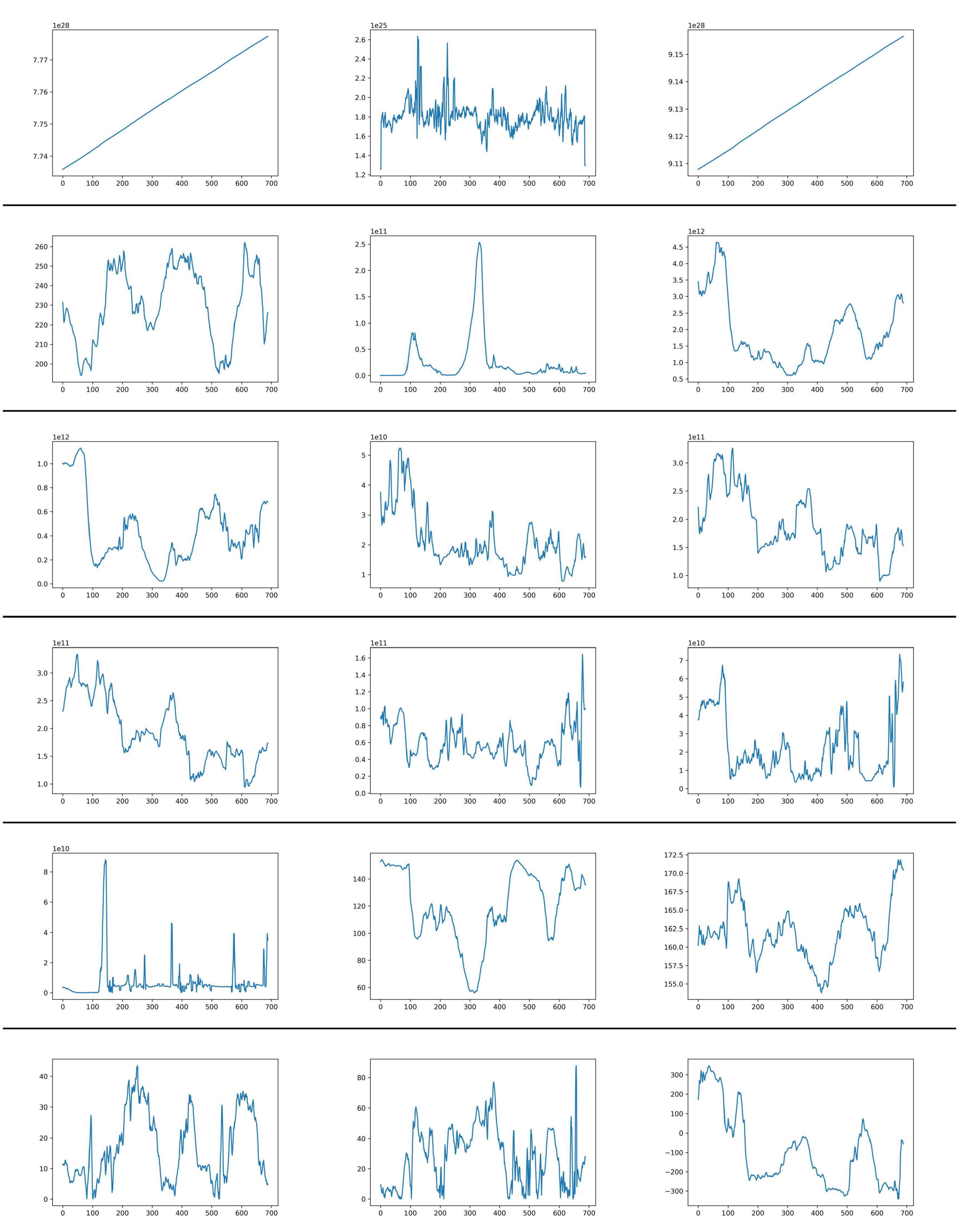


TABLE A.12: Ergonomic Features of a motion sequence of Person 12

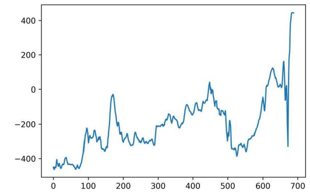
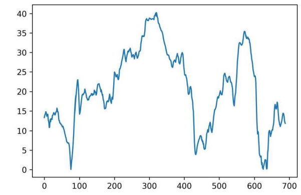
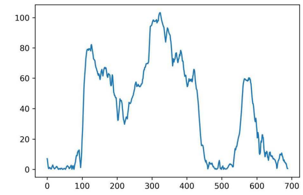
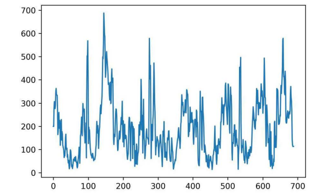
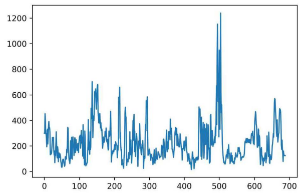
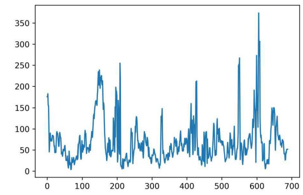
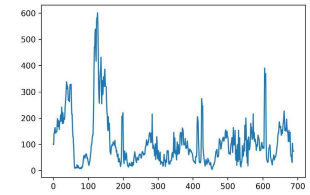
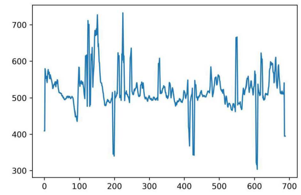
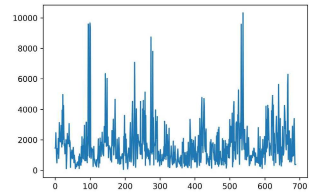
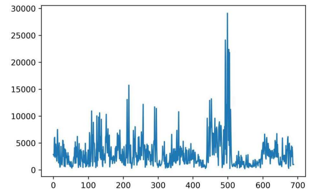
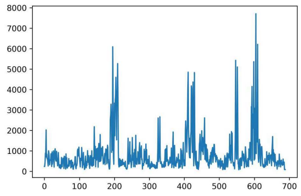
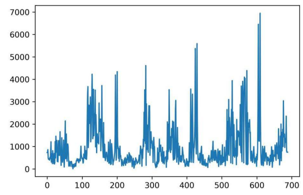
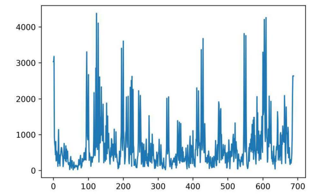
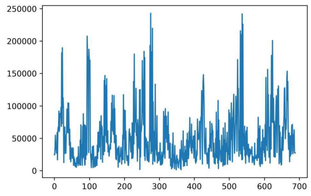
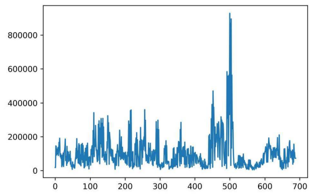
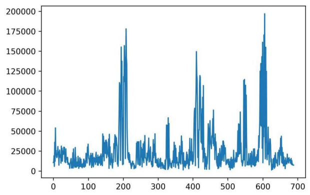
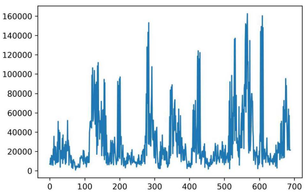
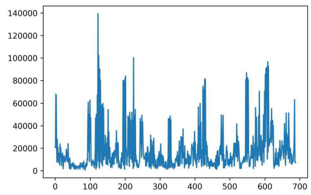
		
		
		
		
		
		

TABLE A.13: Ergonomic Features of a motion sequence of Person 12

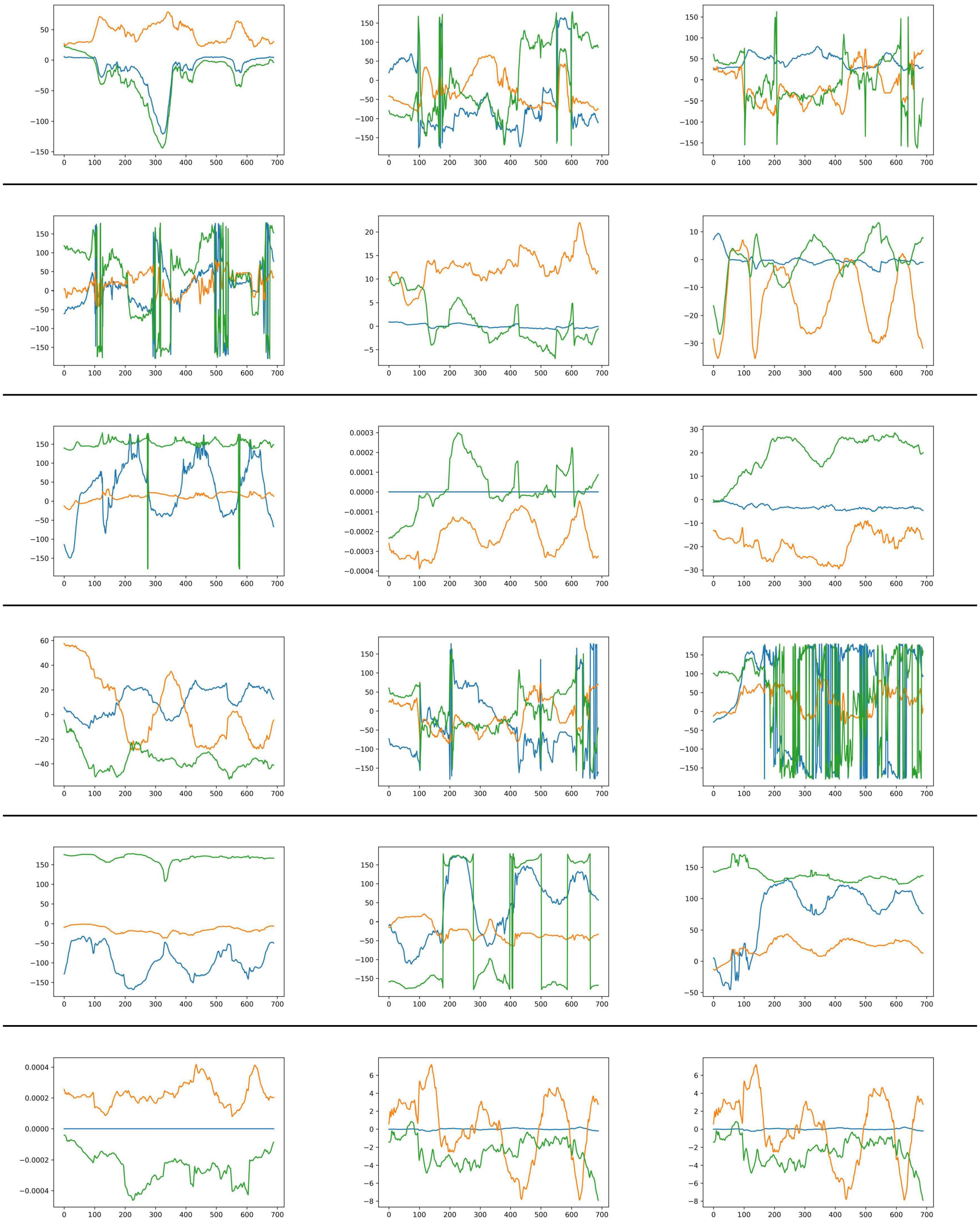


TABLE A.14: Euler Features of a motion sequence of Person 12

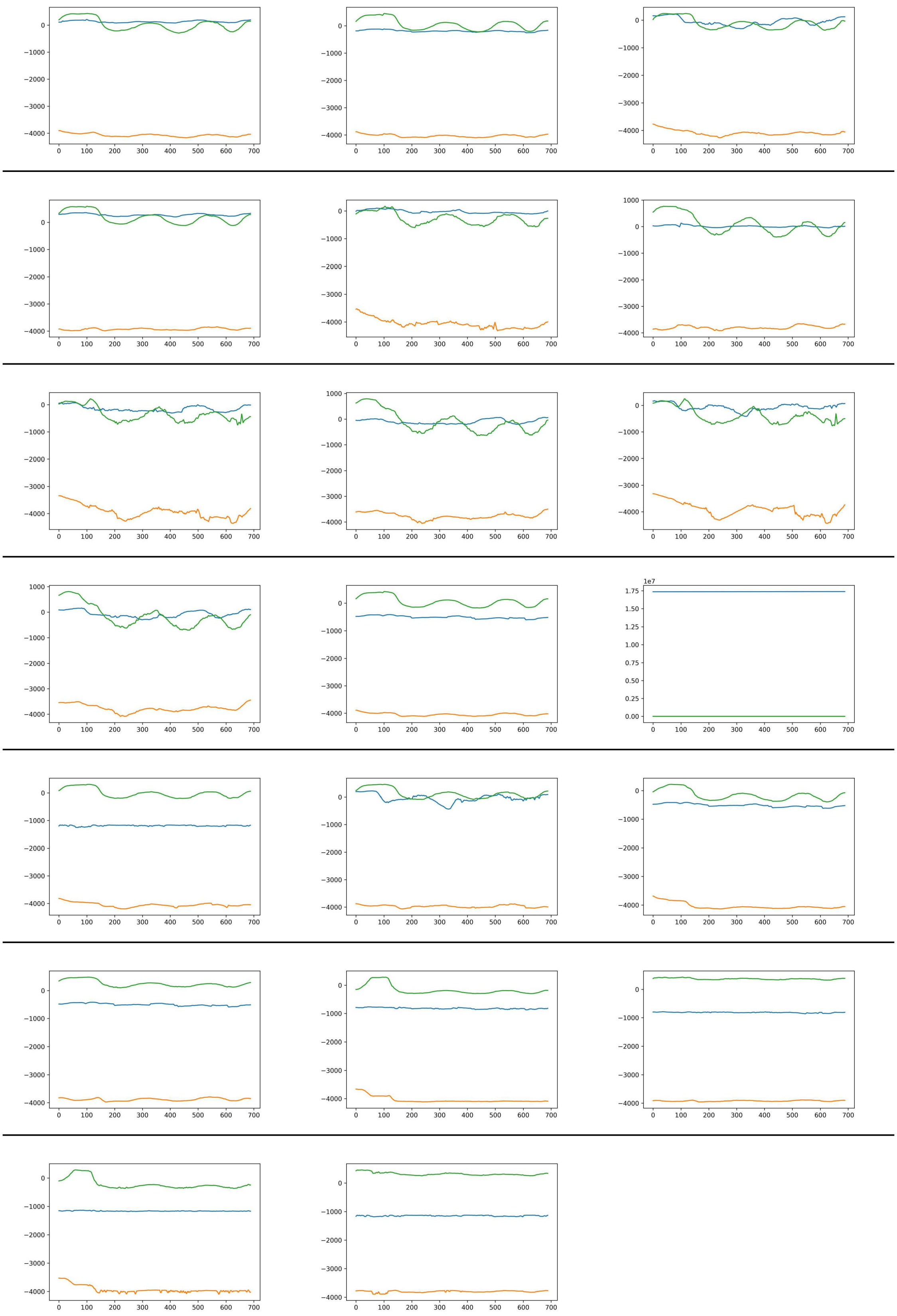


TABLE A.15: Kinematic Features of a motion sequence of Person 12

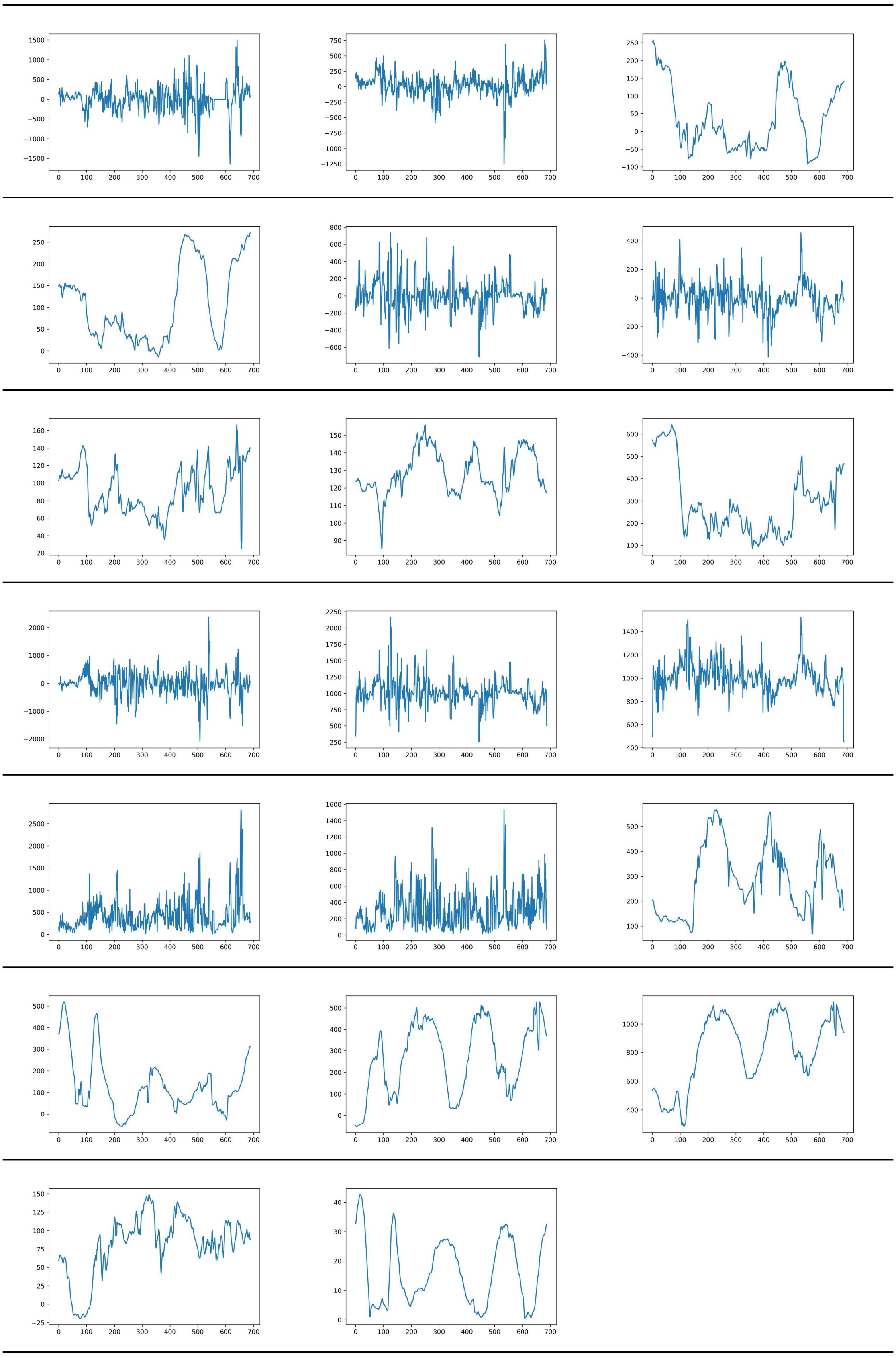


TABLE A.16: Muller Features of a motion sequence of Person 12

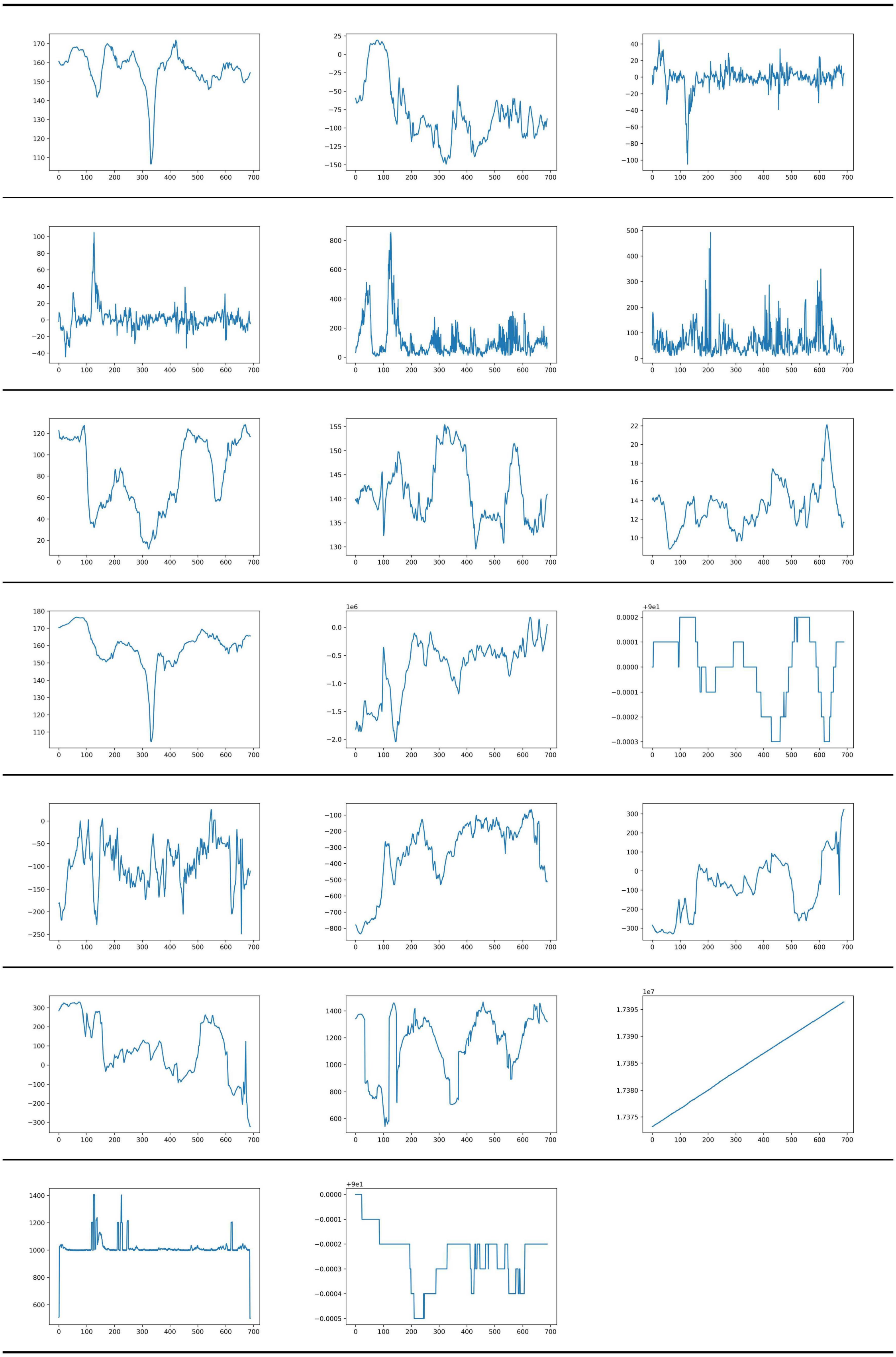
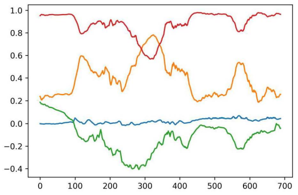
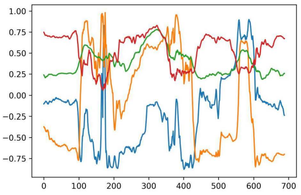
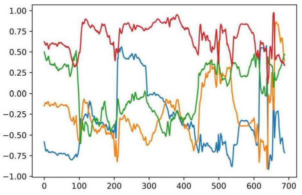
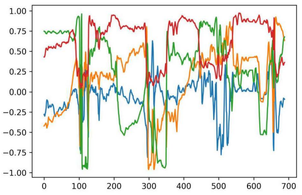


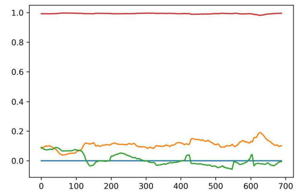
TABLE A.17: Muller Features of a motion sequence of Person 12

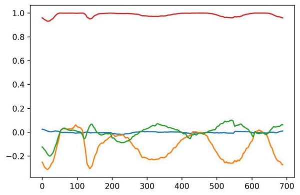


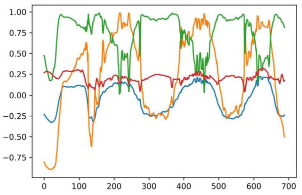


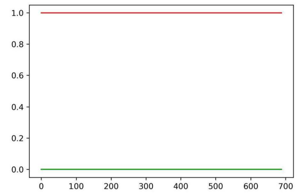


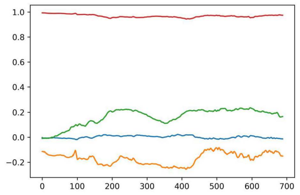


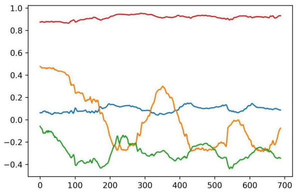


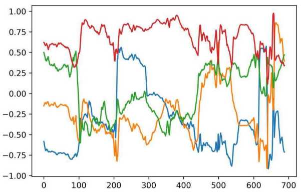


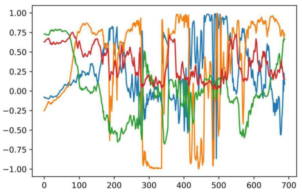


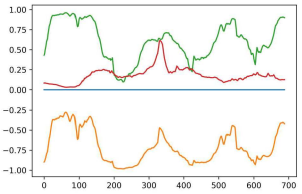


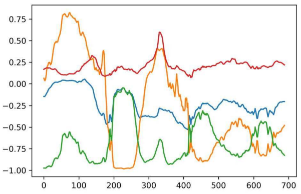


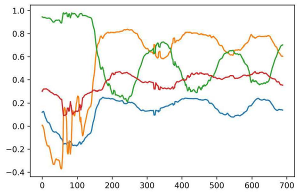


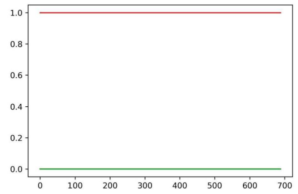


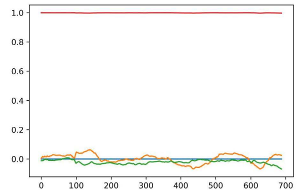












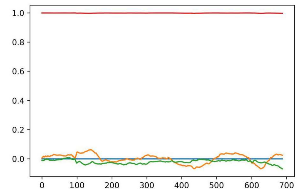


TABLE A.18: Quaternion Features of a motion sequence of Person 12

Bibliography

[1] J. Shotton, R. Girshick, A. Fitzgibbon, T. Sharp, M. Cook, M. Finocchio, R. Moore, P. Kohli, A. Criminisi, A. Kipman, *et al.*, "Efficient human pose estimation from single depth images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2821–2840, 2013.

[2] J. Han, L. Shao, D. Xu, and J. Shotton, "Enhanced computer vision with microsoft kinect sensor: A review", *IEEE transactions on cybernetics*, vol. 43, no. 5, pp. 1318–1334, 2013.

[3] F. Weichert, D. Bachmann, B. Rudak, and D. Fisseler, "Analysis of the accuracy and robustness of the leap motion controller", *Sensors*, vol. 13, no. 5, pp. 6380–6393, 2013.

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

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

[6] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn", in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

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

[8] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, "Rmpe: Regional multi-person pose estimation", in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2334–2343.

[9] B. Daubney, D. Gibson, and N. Campbell, "Estimating pose of articulated objects using low-level motion", *Computer Vision and Image Understanding*, vol. 116, no. 3, pp. 330–346, 2012.

[10] H. Ning, W. Xu, Y. Gong, and T. Huang, "Discriminative learning of visual words for 3d human pose estimation", in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, IEEE, 2008, pp. 1–8.

[11] C. Sminchisescu, "Estimation algorithms for ambiguous visual models: Three dimensional human modeling and motion reconstruction in monocular video sequences", PhD thesis, Institut National Polytechnique de Grenoble-INPG, 2002.

[12] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. Gehler, and B. Schiele, "Deepcut: Joint subset partition and labeling for multi person pose estimation", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2016, pp. 4929–4937. DOI: 10.1109/CVPR.2016.533.

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

[14] G. Papandreou, T. Zhu, L.-C. Chen, S. Gidaris, J. Tompson, and K. Murphy, "Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model", in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 269–286.

[15] U. Iqbal and J. Gall, "Multi-person pose estimation with local joint-to-person associations", in *Proceedings of the European Conference on Computer Vision (ECCV)*, Springer, 2016, pp. 627–642. DOI: 10.1007/978-3-319-46466-4_38.

[16] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele, "Deepercut: A deeper, stronger, and faster multi-person pose estimation model", in *European Conference on Computer Vision (ECCV)*, Springer, 2016, pp. 34–50. DOI: 10.1007/978-3-319-46466-4_3.

[17] A. Toshev and C. 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.

[18] T. Pfister, J. Charles, and A. Zisserman, "Flowing convnets for human pose estimation in videos", in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, IEEE, 2015, pp. 1913–1921. DOI: 10.1109/ICCV.2015.222.

[19] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler, "Joint training of a convolutional network and a graphical model for human pose estimation", *Advances in neural information processing systems*, vol. 27, 2014.

[20] G. Shakhnarovich, P. Viola, and T. Darrell, "Fast pose estimation with parameter-sensitive hashing", in *Computer Vision, IEEE International Conference on*, IEEE Computer Society, vol. 3, 2003, pp. 750–750.

[21] A. Agarwal and B. Triggs, "Recovering 3d human pose from monocular images", in *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 28, 2006, pp. 44–58. DOI: 10.1109/TPAMI.2006.16.

[22] V. Ramakrishna, T. Kanade, and Y. Sheikh, "Reconstructing 3d human pose from 2d image landmarks", in *Proceedings of the European Conference on Computer Vision (ECCV)*, Springer, 2012, pp. 573–586. DOI: `10.1007/978-3-642-33715-4_41`.

[23] L. Sigal, A. O. Balan, and M. J. Black, "Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion", *International journal of computer vision*, vol. 87, no. 1, pp. 4–27, 2010.

[24] L. Bo, C. Sminchisescu, A. Kanaujia, and D. Metaxas, "Fast algorithms for large scale conditional 3d prediction", in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, IEEE, 2008, pp. 1–8.

[25] L. Bo and C. Sminchisescu, "Structured output-associative regression", in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, IEEE, 2009, pp. 2403–2410.

[26] ——, "Twin gaussian processes for structured prediction", *Int J Comput Vis*, vol. 87, pp. 28–52, 2010.

[27] M. Bergtholdt, J. Kappes, S. Schmidt, and C. Schnörr, "A study of parts-based object class detection using complete graphs", *International journal of computer vision*, vol. 87, no. 1, pp. 93–117, 2010.

[28] C.-S. Lee and A Elgammal, "Coupled visual and kinematic manifold models for tracking", *International Journal of Computer Vision*, vol. 87, no. 1, pp. 118–139, 2010.

[29] M. A. Brubaker, D. J. Fleet, and A. Hertzmann, "Physics-based person tracking using the anthropomorphic walker", *International Journal of Computer Vision*, vol. 87, no. 1, pp. 140–155, 2010.

[30] S. Corazza, L. Mündermann, E. Gambaretto, G. Ferrigno, and T. P. Andriacchi, "Markerless motion capture through visual hull, articulated icp and subject specific model generation", *International journal of computer vision*, vol. 87, no. 1, pp. 156–169, 2010.

[31] J. Gall, B. Rosenhahn, T. Brox, and H.-P. Seidel, "Optimization and filtering for human motion capture", *International journal of computer vision*, vol. 87, no. 1, pp. 75–92, 2010.

[32] P. Peursum, S. Venkatesh, and G. West, "A study on smoothing for particle-filtered 3d human body tracking", *International Journal of Computer Vision*, vol. 87, no. 1, pp. 53–74, 2010.

[33] R. Li, T.-P. Tian, S. Sclaroff, and M.-H. Yang, "3d human motion tracking with a coordinated mixture of factor analyzers", *International Journal of Computer Vision*, vol. 87, no. 1, pp. 170–190, 2010.

[34] R. Rosales and S. Sclaroff, "Inferring body pose without tracking body parts", in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, IEEE, vol. 2, 2000, pp. 721–727.

[35] M. Brand, "Shadow puppetry", in *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, IEEE, vol. 2, 1999, pp. 1237–1244.

[36] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts", *IEEE transactions on pattern analysis and machine intelligence*, vol. 24, no. 4, pp. 509–522, 2002.

[37] N. R. Howe, "Silhouette lookup for automatic pose tracking", in *Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04. Conference on*, IEEE, 2004, pp. 15–22.

[38] G. Mori and J. Malik, "Recovering 3d human body configurations using shape contexts", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 7, pp. 1052–1062, 2006.

[39] A. Agarwal and B. Triggs, "Recovering 3d human pose from monocular images", *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 1, pp. 44–58, 2006.

[40] L Kakadiaris and D. Metaxas, "Model-based estimation of 3d human motion", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1453–1459, 2000.

[41] K. Rohr, "Towards model-based recognition of human movements in image sequences", *CVGIP: Image understanding*, vol. 59, no. 1, pp. 94–115, 1994.

[42] S. Wachter and H.-H. Nagel, "Tracking persons in monocular image sequences", *Computer Vision and Image Understanding*, vol. 74, no. 3, pp. 174–192, 1999.

[43] T. Drummond and R. Cipolla, "Real-time tracking of highly articulated structures in the presence of noisy measurements", in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, IEEE, vol. 2, 2001, pp. 315–320.

[44] D. Ramanan and D. A. Forsyth, "Finding and tracking people from the bottom up", in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, IEEE, vol. 2, 2003, pp. II–II.

[45] D. Ramanan, "Learning to parse images of articulated bodies", in *Advances in neural information processing systems*, 2007, pp. 1129–1136.

[46] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: Real-time tracking of the human body", *IEEE Transactions on pattern analysis and machine intelligence*, vol. 19, no. 7, pp. 780–785, 1997.

[47] C. Bregler, J. Malik, and K. Pullen, "Twist based acquisition and tracking of animal and human kinematics", *International Journal of Computer Vision*, vol. 56, no. 3, pp. 179–194, 2004.

[48] S. X. Ju, M. J. Black, and Y. Yacoob, "Cardboard people: A parameterized model of articulated image motion", in *Automatic Face and Gesture Recognition, 1996., Proceedings of the Second International Conference on*, IEEE, 1996, pp. 38–44.

[49] C. Sminchisescu and B. Triggs, "Estimating articulated human motion with covariance scaled sampling", *The International Journal of Robotics Research*, vol. 22, no. 6, pp. 371–391, 2003.

[50] D. G. Lowe, "Distinctive image features from scale-invariant keypoints", *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[51] J. Müller and M. Arens, "Human pose estimation with implicit shape models", in *Proceedings of the first ACM international workshop on Analysis and retrieval of tracked events and motion in imagery streams*, ACM, 2010, pp. 9–14.

[52] S. Belongie, J. Malik, and J. Puzicha, "Shape context: A new descriptor for shape matching and object recognition", in *Advances in neural information processing systems*, 2001, pp. 831–837.

[53] S. Amin, M. Andriluka, M. Rohrbach, and B. Schiele, "Multi-view pictorial structures for 3d human pose estimation.", in *Bmvc*, 2013.

[54] M. Andriluka, S. Roth, and B. Schiele, "Pictorial structures revisited: People detection and articulated pose estimation", in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, IEEE, 2009, pp. 1014–1021.

[55] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection", in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, IEEE, vol. 1, 2005, pp. 886–893.

[56] G. Gkioxari, P. Arbelaez, L. Bourdev, and J. Malik, "Articulated pose estimation using discriminative armlet classifiers", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3342–3349.

[57] Y. Yang and D. Ramanan, "Articulated pose estimation with flexible mixtures-of-parts", in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, IEEE, 2011, pp. 1385–1392.

[58] G. Pons-Moll, D. J. Fleet, and B. Rosenhahn, "Posebits for monocular human pose estimation", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2337–2344.

[59] A. Kanaujia, C. Sminchisescu, and D. Metaxas, "Semi-supervised hierarchical models for 3d human pose reconstruction", in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, IEEE, 2007, pp. 1–8.

[60] L. Sigal, "Human pose estimation", *Encyclopedia of Computer Vision*, 2011.

[61] M. Andriluka, S. Roth, and B. Schiele, "Monocular 3d pose estimation and tracking by detection", in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, IEEE, 2010, pp. 623–630.

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

[63] Q. Wang, G. Kurillo, F. Ofli, and R. Bajcsy, "Evaluation of pose tracking accuracy in the first and second generations of microsoft kinect", in *2015 international conference on healthcare informatics*, IEEE, 2015, pp. 380–389.

[64] A. P. Rocha, H. Choupina, J. M. Fernandes, M. J. Rosas, R. Vaz, and J. P. S. Cunha, "Kinect v2 based system for parkinson's disease assessment", in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, 2015, pp. 1279–1282.

[65] J. O'rourke and N. I. Badler, "Model-based image analysis of human motion using constraint propagation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 522–536, 1980.

[66] D. Hogg, "Model-based vision: A program to see a walking person", *Image and Vision computing*, vol. 1, no. 1, pp. 5–20, 1983.

[67] K. Rohr, "Towards model-based recognition of human movements in image sequences", *CVGIP: Image understanding*, vol. 59, no. 1, pp. 94–115, 1994.

[68] J. M. Rehg, D. D. Morris, and T. Kanade, "Ambiguities in visual tracking of articulated objects using two-and three-dimensional models", *The International Journal of Robotics Research*, vol. 22, no. 6, pp. 393–418, 2003.

[69] T. Sharp, C. Keskin, D. Robertson, J. Taylor, J. Shotton, D. Kim, C. Rhemann, I. Leichter, A. Vinnikov, Y. Wei, *et al.*, "Accurate, robust, and flexible real-time hand tracking", in *Proceedings of the 33rd annual ACM conference on human factors in computing systems*, 2015, pp. 3633–3642.

[70] J. Taylor, L. Bordeaux, T. Cashman, B. Corish, C. Keskin, T. Sharp, E. Soto, D. Sweeney, J. Valentin, B. Luff, *et al.*, "Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences", *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, pp. 1–12, 2016.

[71] C. Zimmermann and T. Brox, "Learning to estimate 3d hand pose from single rgb images", in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 4903–4911.

[72] F. Mueller, F. Bernard, O. Sotnychenko, D. Mehta, S. Sridhar, D. Casas, and C. Theobalt, "Ganerated hands for real-time 3d hand tracking from monocular rgb", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 49–59.

[73] Y. Hasson, G. Varol, D. Tzionas, I. Kalevatykh, M. J. Black, I. Laptev, and C. Schmid, "Learning joint reconstruction of hands and manipulated objects", in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 11 807–11 816.

[74] D. Shan, J. Geng, M. Shu, and D. F. Fouhey, "Understanding human hands in contact at internet scale", in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 9869–9878.

[75] G. Moon, T. Shiratori, and K. M. Lee, "Deephandmesh: A weakly-supervised deep encoder-decoder framework for high-fidelity hand mesh modeling", in *European Conference on Computer Vision*, Springer, 2020, pp. 440–455.

[76] G. Moon, S.-I. Yu, H. Wen, T. Shiratori, and K. M. Lee, "Interhand2. 6m: A dataset and baseline for 3d interacting hand pose estimation from a single rgb image", in *European Conference on Computer Vision*, Springer, 2020, pp. 548–564.

[77] A. Spurr, U. Iqbal, P. Molchanov, O. Hilliges, and J. Kautz, "Weakly supervised 3d hand pose estimation via biomechanical constraints", in *European Conference on Computer Vision*, Springer, 2020, pp. 211–228.

[78] O. Taheri, N. Ghorbani, M. J. Black, and D. Tzionas, "Grab: A dataset of whole-body human grasping of objects", in *European conference on computer vision*, Springer, 2020, pp. 581–600.

[79] J. Romero, D. Tzionas, and M. J. Black, "Embodied hands: Modeling and capturing hands and bodies together", *arXiv preprint arXiv:2201.02610*, 2022.

[80] H. Joo, T. Simon, and Y. Sheikh, "Total capture: A 3d deformation model for tracking faces, hands, and bodies", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8320–8329.

[81] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. J. Black, "Expressive body capture: 3d hands, face, and body from a single image", in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 10 975–10 985.

[82] Y. Rong, T. Shiratori, and H. Joo, "Frankmocap: Fast monocular 3d hand and body motion capture by regression and integration", *arXiv preprint arXiv:2008.08324*, 2020.

[83] D. M. Gavrila and L. S. Davis, "3-d model-based tracking of humans in action: A multi-view approach", in *Proceedings cvpr ieee computer society conference on computer vision and pattern recognition*, IEEE, 1996, pp. 73–80.

[84] L. Kakadiaris and D. Metaxas, "Model-based estimation of 3d human motion", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1453–1459, 2000.

[85] C. Bregler, J. Malik, and K. Pullen, "Twist based acquisition and tracking of animal and human kinematics", *International Journal of Computer Vision*, vol. 56, no. 3, pp. 179–194, 2004.

[86] R. Kehl and L. Van Gool, "Markerless tracking of complex human motions from multiple views", *Computer Vision and Image Understanding*, vol. 104, no. 2-3, pp. 190–209, 2006.

[87] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images", in *CVPR 2011*, Ieee, 2011, pp. 1297–1304.

[88] J. Taylor, J. Shotton, T. Sharp, and A. Fitzgibbon, "The vitruvian manifold: Inferring dense correspondences for one-shot human pose estimation", in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2012, pp. 103–110.

[89] J. Shotton, R. Girshick, A. Fitzgibbon, T. Sharp, M. Cook, M. Finocchio, R. Moore, P. Kohli, A. Criminisi, A. Kipman, *et al.*, "Efficient human pose estimation from single depth images", *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 12, pp. 2821–2840, 2012.

[90] A. Baak, M. Müller, G. Bharaj, H.-P. Seidel, and C. Theobalt, "A data-driven approach for real-time full body pose reconstruction from a depth camera", in *Consumer depth cameras for computer vision*, Springer, 2013, pp. 71–98.

[91] D. Mehta, S. Sridhar, O. Sotnychenko, H. Rhodin, M. Shafiei, H.-P. Seidel, W. Xu, D. Casas, and C. Theobalt, "Vnect: Real-time 3d human pose estimation with a single rgb camera", *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, pp. 1–14, 2017.

[92] M. Habermann, W. Xu, M. Zollhoefer, G. Pons-Moll, and C. Theobalt, "Livecap: Real-time human performance capture from monocular video", *ACM Transactions On Graphics (TOG)*, vol. 38, no. 2, pp. 1–17, 2019.

[93] R. Polana and R. C. Nelson, "Detection and recognition of periodic, nonrigid motion", *International Journal of Computer Vision*, vol. 23, no. 3, pp. 261–282, 1997.

[94] S. M. Seitz and C. R. Dyer, "View-invariant analysis of cyclic motion", *International Journal of Computer Vision*, vol. 25, no. 3, pp. 231–251, 1997.

[95] R. Cutler and L. S. Davis, "Robust real-time periodic motion detection, analysis, and applications", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 781–796, 2000.

[96] H. Sidenbladh, M. J. Black, and D. J. Fleet, "Stochastic tracking of 3d human figures using 2d image motion", in *European conference on computer vision*, Springer, 2000, pp. 702–718.

[97] R. Urtasun, D. J. Fleet, and P. Fua, "Temporal motion models for monocular and multiview 3d human body tracking", *Computer vision and image understanding*, vol. 104, no. 2-3, pp. 157–177, 2006.

[98] B. Allen, B. Curless, and Z. Popović, "The space of human body shapes: Reconstruction and parameterization from range scans", *ACM transactions on graphics (TOG)*, vol. 22, no. 3, pp. 587–594, 2003.

[99] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis, "Scape: Shape completion and animation of people", in *ACM SIGGRAPH 2005 Papers*, 2005, pp. 408–416.

[100] A. O. Bălan and M. J. Black, "The naked truth: Estimating body shape under clothing", in *European Conference on Computer Vision*, Springer, 2008, pp. 15–29.

[101] D. Vlasic, I. Baran, W. Matusik, and J. Popović, "Articulated mesh animation from multi-view silhouettes", in *ACM SIGGRAPH 2008 papers*, 2008, pp. 1–9.

[102] P. Guan, A. Weiss, A. O. Balan, and M. J. Black, "Estimating human shape and pose from a single image", in *2009 IEEE 12th International Conference on Computer Vision*, IEEE, 2009, pp. 1381–1388.

[103] D. A. Hirshberg, M. Loper, E. Rachlin, and M. J. Black, "Coregistration: Simultaneous alignment and modeling of articulated 3d shape", in *European conference on computer vision*, Springer, 2012, pp. 242–255.

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

[105] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black, "Keep it smpl: Automatic estimation of 3d human pose and shape from a single image", in *European conference on computer vision*, Springer, 2016, pp. 561–578.

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

[107] M. Kocabas, N. Athanasiou, and M. J. Black, "Vibe: Video inference for human body pose and shape estimation", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 5253–5263.

[108] V. Choutas, G. Pavlakos, T. Bolkart, D. Tzionas, and M. J. Black, "Monocular expressive body regression through body-driven attention", in *European Conference on Computer Vision*, Springer, 2020, pp. 20–40.

[109] M. Loper, N. Mahmood, and M. J. Black, "Mosh: Motion and shape capture from sparse markers", *ACM Transactions on Graphics (ToG)*, vol. 33, no. 6, pp. 1–13, 2014.

[110] A. A. Osman, T. Bolkart, and M. J. Black, "Star: Sparse trained articulated human body regressor", in *European Conference on Computer Vision*, Springer, 2020, pp. 598–613.

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

[112] T. von Marcard, R. Henschel, M. J. Black, B. Rosenhahn, and G. Pons-Moll, "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.

[113] J. Song, X. Chen, and O. Hilliges, "Human body model fitting by learned gradient descent", in *European Conference on Computer Vision*, Springer, 2020, pp. 744–760.

[114] H. Joo, N. Neverova, and A. Vedaldi, "Exemplar fine-tuning for 3d human pose fitting towards in-the-wild 3d human pose estimation", 2020.

[115] M. Isard and A. Blake, "Condensation—conditional density propagation for visual tracking", *International journal of computer vision*, vol. 29, no. 1, pp. 5–28, 1998.

[116] J. Deutscher, A. Blake, and I. Reid, "Articulated body motion capture by annealed particle filtering", in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, IEEE, vol. 2, 2000, pp. 126–133.

[117] J. Deutscher and I. Reid, "Articulated body motion capture by stochastic search", *International Journal of Computer Vision*, vol. 61, no. 2, pp. 185–205, 2005.

[118] E.-J. Ong, A. S. Micilotta, R. Bowden, and A. Hilton, "Viewpoint invariant exemplar-based 3d human tracking", *Computer Vision and Image Understanding*, vol. 104, no. 2-3, pp. 178–189, 2006.

[119] T.-J. Cham and J. M. Rehg, "A multiple hypothesis approach to figure tracking", in *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, IEEE, vol. 2, 1999, pp. 239–244.

[120] C. Sminchisescu and B. Triggs, "Covariance scaled sampling for monocular 3d body tracking", in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, IEEE, vol. 1, 2001, pp. I–I.

[121] L. Sigal, S. Bhatia, S. Roth, M. J. Black, and M. Isard, "Tracking loose-limbed people", in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, IEEE, vol. 1, 2004, pp. I–I.

[122] A. Hilton, P. Fua, and R. Ronfard, "Modeling people: Vision-based understanding of a person's shape, appearance, movement, and behaviour", *Computer Vision and Image Understanding*, vol. 104, no. 2, pp. 87–89, 2006.

[123] D. A. Forsyth, O. Arikan, L. Ikemoto, J. O'Brien, D. Ramanan, *et al.*, "Computational studies of human motion: Part 1, tracking and motion synthesis", *Foundations and Trends® in Computer Graphics and Vision*, vol. 1, no. 2–3, pp. 77–254, 2006.

[124] T. B. Moeslund, A. Hilton, and V. Krüger, "A survey of advances in vision-based human motion capture and analysis", *Computer vision and image understanding*, vol. 104, no. 2-3, pp. 90–126, 2006.

[125] L. Sigal, A. O. Balan, and M. J. Black, "Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion", *International journal of computer vision*, vol. 87, no. 1, pp. 4–27, 2010.

[126] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "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.

[127] F. Bogo, J. Romero, M. Loper, and M. J. Black, "Faust: Dataset and evaluation for 3d mesh registration", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 3794–3801.

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

[129] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments", *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 7, pp. 1325–1339, 2014.

[130] H. Yasin, B. Krüger, and A. Weber, "Model based full body human motion reconstruction from video data", in *Proceedings of the 6th International Conference on Computer Vision/Computer Graphics Collaboration Techniques and Applications*, ACM, 2013, p. 1.

[131] V. Ramakrishna, T. Kanade, and Y. Sheikh, "Reconstructing 3d human pose from 2d image landmarks", *Computer Vision–ECCV 2012*, pp. 573–586, 2012.

[132] C. Wang, Y. Wang, Z. Lin, A. L. Yuille, and W. Gao, "Robust estimation of 3d human poses from a single image", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2361–2368.

[133] E. Simo-Serra, A. Ramisa, G. Alenyà, C. Torras, and F. Moreno-Noguer, "Single image 3d human pose estimation from noisy observations", in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, IEEE, 2012, pp. 2673–2680.

[134] E. Simo-Serra, A. Quattoni, C. Torras, and F. Moreno-Noguer, "A joint model for 2d and 3d pose estimation from a single image", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3634–3641.

[135] R. Urtasun, D. J. Fleet, and P. Fua, "3d people tracking with gaussian process dynamical models", in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, IEEE, vol. 1, 2006, pp. 238–245.

[136] C. Stoll, N. Hasler, J. Gall, H.-P. Seidel, and C. Theobalt, "Fast articulated motion tracking using a sums of gaussians body model", in *Computer Vision (ICCV), 2011 IEEE International Conference on*, IEEE, 2011, pp. 951–958.

[137] J. Tong, J. Zhou, L. Liu, Z. Pan, and H. Yan, "Scanning 3d full human bodies using kinects", *IEEE transactions on visualization and computer graphics*, vol. 18, no. 4, pp. 643–650, 2012.

[138] "Openni", *http://www.openni.org/*, 2011.

[139] T. Shiratori, H. S. Park, Y. Sheikh, J. K. Hodgins, *et al.*, *Motion capture from body mounted cameras*, US Patent 8,786,680, 2014.

[140] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, "Wallflower: Principles and practice of background maintenance", in *Proceedings of the seventh IEEE international conference on computer vision*, IEEE, vol. 1, 1999, pp. 255–261.

[141] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking", in *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, IEEE, vol. 2, 1999, pp. 246–252.

[142] H. Sidenbladh and M. J. Black, "Learning the statistics of people in images and video", *International Journal of Computer Vision*, vol. 54, no. 1, pp. 183–209, 2003.

[143] P. F. Felzenszwalb and D. P. Huttenlocher, "Pictorial structures for object recognition", *International journal of computer vision*, vol. 61, no. 1, pp. 55–79, 2005.

[144] M. Eichner, M. Marin-Jimenez, A. Zisserman, and V. Ferrari, "2d articulated human pose estimation and retrieval in (almost) unconstrained still images", *International journal of computer vision*, vol. 99, no. 2, pp. 190–214, 2012.

[145] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts", in *ACM transactions on graphics (TOG)*, ACM, vol. 23, 2004, pp. 309–314.

[146] S. Sengupta, V. Jayaram, B. Curless, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Background matting: The world is your green screen", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2291–2300.

[147] S. Lin, A. Ryabtsev, S. Sengupta, B. L. Curless, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Real-time high-resolution background matting", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8762–8771.

[148] A. Baumberg and D. Hogg, "Generating spatiotemporal models from examples", *Image and Vision Computing*, vol. 14, no. 8, pp. 525–532, 1996.

[149] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: Real-time tracking of the human body", *IEEE Transactions on pattern analysis and machine intelligence*, vol. 19, no. 7, pp. 780–785, 1997.

[150] I Haritaoglu, D Harwood, and L Davis, *W4: Real-time surveillance of people and their activities. 22 (8): 809–830*, 2000.

[151] A. Mittal and L. S. Davis, "M2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene", *International Journal of Computer Vision*, vol. 51, no. 3, pp. 189–203, 2003.

[152] M. Dimitrijevic, V. Lepetit, and P. Fua, "Human body pose detection using bayesian spatio-temporal templates", *Computer vision and image understanding*, vol. 104, no. 2-3, pp. 127–139, 2006.

[153] G. Mori, X. Ren, A. A. Efros, and J. Malik, "Recovering human body configurations: Combining segmentation and recognition", in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, IEEE, vol. 2, 2004, pp. II–II.

[154] P. F. Felzenszwalb and D. P. Huttenlocher, "Pictorial structures for object recognition", *International journal of computer vision*, vol. 61, no. 1, pp. 55–79, 2005.

[155] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model", in *2008 IEEE conference on computer vision and pattern recognition*, Ieee, 2008, pp. 1–8.

[156] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art", *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 4, pp. 743–761, 2011.

[157] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection", *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 8, pp. 1532–1545, 2014.

[158] P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. LeCun, "Pedestrian detection with unsupervised multi-stage feature learning", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 3626–3633.

[159] W. Ouyang and X. Wang, "Joint deep learning for pedestrian detection", in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 2056–2063.

[160] Y. Tian, P. Luo, X. Wang, and X. Tang, "Deep learning strong parts for pedestrian detection", in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1904–1912.

[161] L. Zhang, L. Lin, X. Liang, and K. He, "Is faster r-cnn doing well for pedestrian detection?", in *European conference on computer vision*, Springer, 2016, pp. 443–457.

[162] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models", *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[163] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features", in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, IEEE, vol. 1, 2001, pp. I–I.

[164] B. Sapp, D. Weiss, and B. Taskar, "Parsing human motion with stretchable models", in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, IEEE, 2011, pp. 1281–1288.

[165] M. Eichner and V. Ferrari, "We are family: Joint pose estimation of multiple persons", *Computer Vision–ECCV 2010*, pp. 228–242, 2010.

[166] N. Howe, M. Leventon, and W. Freeman, "Bayesian reconstruction of 3d human motion from single-camera video", *Advances in neural information processing systems*, vol. 12, 1999.

[167] R. Rosales and S. Sclaroff, "Inferring body pose without tracking body parts", in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662)*, IEEE, vol. 2, 2000, pp. 721–727.

[168] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas, "Discriminative density propagation for 3d human motion estimation", in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, IEEE, vol. 1, 2005, pp. 390–397.

[169] A. Agarwal and B. Triggs, "Recovering 3d human pose from monocular images", *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 1, pp. 44–58, 2005.

[170] M. W. Lee and I. Cohen, "A model-based approach for estimating human 3d poses in static images", *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 6, pp. 905–916, 2006.

[171] L. Sigal and M. J. Black, "Predicting 3d people from 2d pictures", in *International Conference on Articulated Motion and Deformable Objects*, Springer, 2006, pp. 185–195.

[172] B. Stenger, A. Thayananthan, P. H. Torr, and R. Cipolla, "Model-based hand tracking using a hierarchical bayesian filter", *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 9, pp. 1372–1384, 2006.

[173] D. Ramanan, D. A. Forsyth, and A. Zisserman, "Strike a pose: Tracking people by finding stylized poses", in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, IEEE, vol. 1, 2005, pp. 271–278.

[174] G. Rogez, J. Rihan, S. Ramalingam, C. Orrite, and P. H. Torr, "Randomized trees for human pose detection", in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2008, pp. 1–8.

[175] L. Bourdev and J. Malik, "Poselets: Body part detectors trained using 3d human pose annotations", in *2009 IEEE 12th International Conference on Computer Vision*, IEEE, 2009, pp. 1365–1372.

[176] R. A. Güler, N. Neverova, and I. Kokkinos, "Densepose: Dense human pose estimation in the wild", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7297–7306.

[177] A. Fossati, M. Dimitrijevic, V. Lepetit, and P. Fua, "Bridging the gap between detection and tracking for 3d monocular video-based motion capture", in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2007, pp. 1–8.

[178] M. Andriluka, S. Roth, and B. Schiele, "People-tracking-by-detection and people-detection-by-tracking", in *2008 IEEE Conference on computer vision and pattern recognition*, IEEE, 2008, pp. 1–8.

[179] V. Ferrari, M. Marin-Jimenez, and A. Zisserman, "Progressive search space reduction for human pose estimation", in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2008, pp. 1–8.

[180] S. X. Ju, M. J. Black, and Y. Yacoob, "Cardboard people: A parameterized model of articulated image motion", in *Proceedings of the Second International Conference on Automatic Face and Gesture Recognition*, IEEE, 1996, pp. 38–44.

[181] A. A. Efros, A. C. Berg, G. Mori, and J. Malik, "Recognizing action at a distance", in *Computer Vision, IEEE International Conference on*, IEEE Computer Society, vol. 3, 2003, pp. 726–726.

[182] R. White, K. Crane, and D. A. Forsyth, "Capturing and animating occluded cloth", *ACM Transactions on Graphics (TOG)*, vol. 26, no. 3, 34–es, 2007.

[183] J. Pilet, V. Lepetit, and P. Fua, "Fast non-rigid surface detection, registration and realistic augmentation", *International Journal of Computer Vision*, vol. 76, no. 2, pp. 109–122, 2008.

[184] Y. Furukawa and J. Ponce, "Dense 3d motion capture from synchronized video streams", in *Image and Geometry Processing for 3-D Cinematography*, Springer, 2010, pp. 193–211.

[185] M. Salzmann and P. Fua, "Deformable surface 3d reconstruction from monocular images", *Synthesis Lectures on Computer Vision*, vol. 2, no. 1, pp. 1–113, 2010.

[186] A. Bozic, M. Zollhofer, C. Theobalt, and M. Nießner, "Deepdeform: Learning non-rigid rgb-d reconstruction with semi-supervised data", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7002–7012.

[187] A. Bozic, P. Palafox, M. Zollhöfer, A. Dai, J. Thies, and M. Nießner, "Neural non-rigid tracking", *Advances in Neural Information Processing Systems*, vol. 33, pp. 18 727–18 737, 2020.

[188] E. De Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H.-P. Seidel, and S. Thrun, "Performance capture from sparse multi-view video", in *ACM SIGGRAPH 2008 papers*, 2008, pp. 1–10.

[189] D. Stavens and S. Thrun, "Unsupervised learning of invariant features using video", in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, IEEE, 2010, pp. 1649–1656.

[190] B. Sapp, A. Toshev, and B. Taskar, "Cascaded models for articulated pose estimation", *Computer Vision–ECCV 2010*, pp. 406–420, 2010.

[191] A. Toshev and C. 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.

[192] X. Chen and A. L. Yuille, "Articulated pose estimation by a graphical model with image dependent pairwise relations", in *Advances in Neural Information Processing Systems*, 2014, pp. 1736–1744.

[193] W. Ouyang, X. Chu, and X. Wang, "Multi-source deep learning for human pose estimation", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2329–2336.

[194] J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, and R. Moore, "Real-time human pose recognition in parts from single depth images", *Communications of the ACM*, vol. 56, no. 1, pp. 116–124, 2013.

[195] V. Ramakrishna, D. Munoz, M. Hebert, J. A. Bagnell, and Y. Sheikh, "Pose machines: Articulated pose estimation via inference machines", in *European Conference on Computer Vision*, Springer, 2014, pp. 33–47.

[196] D. Tran and D. Forsyth, "Improved human parsing with a full relational model", *Computer Vision–ECCV 2010*, pp. 227–240, 2010.

[197] L. Karlinsky and S. Ullman, "Using linking features in learning non-parametric part models", in *European Conference on Computer Vision*, Springer, 2012, pp. 326–339.

[198] M. A. Fischler and R. A. Elschlager, "The representation and matching of pictorial structures", *IEEE Transactions on computers*, vol. 100, no. 1, pp. 67–92, 1973.

[199] M. Eichner, V. Ferrari, and S Zurich, "Better appearance models for pictorial structures.", in *BMVC*, vol. 2, 2009, p. 5.

[200] B. Sapp and B. Taskar, "Modec: Multimodal decomposable models for human pose estimation", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3674–3681.

[201] M. Kiefel and P. V. Gehler, "Human pose estimation with fields of parts", in *European Conference on Computer Vision*, Springer, 2014, pp. 331–346.

[202] L. Pishchulin, M. Andriluka, P. Gehler, and B. Schiele, "Poselet conditioned pictorial structures", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 588–595.

[203] ——, "Strong appearance and expressive spatial models for human pose estimation", in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 3487–3494.

[204] D. Mehta, O. Sotnychenko, F. Mueller, W. Xu, S. Sridhar, G. Pons-Moll, and C. Theobalt, "Single-shot multi-person 3d pose estimation from monocular rgb", in *2018 International Conference on 3D Vision (3DV)*, IEEE, 2018, pp. 120–130.

[205] D. Mehta, O. Sotnychenko, F. Mueller, W. Xu, M. Elgharib, P. Fua, H.-P. Seidel, H. Rhodin, G. Pons-Moll, and C. Theobalt, "Xnect: Real-time multi-person 3d human pose estimation with a single rgb camera", *arXiv preprint arXiv:1907.00837*, 2019.

[206] J. Zhen, Q. Fang, J. Sun, W. Liu, W. Jiang, H. Bao, and X. Zhou, "Smap: Single-shot multi-person absolute 3d pose estimation", in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16*, Springer, 2020, pp. 550–566.

[207] R. De Bem, A. Arnab, S. Golodetz, M. Sapienza, and P. Torr, "Deep fully-connected part-based models for human pose estimation", in *Asian conference on machine learning*, PMLR, 2018, pp. 327–342.

[208] I. Radwan, A. Dhall, and R. Goecke, "Monocular image 3d human pose estimation under self-occlusion", in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1888–1895.

[209] G. Rogez, P. Weinzaepfel, and C. Schmid, "Lcr-net: Localization-classification-regression for human pose", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3433–3441.

[210] R. Gu, G. Wang, and J.-N. Hwang, "Exploring severe occlusion: Multi-person 3d pose estimation with gated convolution", in *2020 25th International Conference on Pattern Recognition (ICPR)*, IEEE, 2021, pp. 8243–8250.

[211] M. Kocabas, C.-H. P. Huang, O. Hilliges, and M. J. Black, "Pare: Part attention regressor for 3d human body estimation", in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 127–11 137.

[212] L. Zhou, Y. Chen, Y. Gao, J. Wang, and H. Lu, "Occlusion-aware siamese network for human pose estimation", in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*, Springer, 2020, pp. 396–412.

[213] Y. Cheng, B. Yang, B. Wang, W. Yan, and R. T. Tan, "Occlusion-aware networks for 3d human pose estimation in video", in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 723–732.

[214] B. Artacho and A. Savakis, "Unipose: Unified human pose estimation in single images and videos", in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 7035–7044.

[215] Y. Cai, L. Ge, J. Liu, J. Cai, T.-J. Cham, J. Yuan, and N. M. Thalmann, "Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks", in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 2272–2281.

[216] J. Liu, J. Rojas, Y. Li, Z. Liang, Y. Guan, N. Xi, and H. Zhu, "A graph attention spatio-temporal convolutional network for 3d human pose estimation in video", in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2021, pp. 3374–3380.

[217] M. Parger, C. Tang, Y. Xu, C. D. Twigg, L. Tao, Y. Li, R. Wang, and M. Steinberger, "Unoc: Understanding occlusion for embodied presence in virtual reality", *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 12, pp. 4240–4251, 2021.

[218] M. Véges and A Lőrincz, "Temporal smoothing for 3d human pose estimation and localization for occluded people", in *Neural Information Processing: 27th International Conference, ICONIP 2020, Bangkok, Thailand, November 23–27, 2020, Proceedings, Part I 27*, Springer, 2020, pp. 557–568.

[219] J. Wang, E. Xu, K. Xue, and L. Kidzinski, "3d pose detection in videos: Focusing on occlusion", *arXiv preprint arXiv:2006.13517*, 2020.

[220] J. Xu, Z. Yu, B. Ni, J. Yang, X. Yang, and W. Zhang, "Deep kinematics analysis for monocular 3d human pose estimation", in *Proceedings of the IEEE/CVF Conference on computer vision and Pattern recognition*, 2020, pp. 899–908.

[221] S. Park and J. Park, "Localizing human keypoints beyond the bounding box", in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 1602–1611.

[222] Y. Cheng, B. Yang, B. Wang, and R. T. Tan, "3d human pose estimation using spatio-temporal networks with explicit occlusion training", in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 10 631–10 638.

[223] X. Peng, Z. Tang, F. Yang, R. S. Feris, and D. Metaxas, "Jointly optimize data augmentation and network training: Adversarial data augmentation in human pose estimation", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 2226–2234.

[224] Y. Xu, W. Wang, T. Liu, X. Liu, J. Xie, and S.-C. Zhu, "Monocular 3d pose estimation via pose grammar and data augmentation", *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 10, pp. 6327–6344, 2021.

[225] B. Biggs, D. Novotny, S. Ehrhardt, H. Joo, B. Graham, and A. Vedaldi, "3d multi-bodies: Fitting sets of plausible 3d human models to ambiguous image data", *Advances in Neural Information Processing Systems*, vol. 33, pp. 20 496–20 507, 2020.

[226] E. Jahangiri and A. L. Yuille, "Generating multiple diverse hypotheses for human 3d pose consistent with 2d joint detections", in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pp. 805–814.

[227] C. Li and G. H. Lee, "Generating multiple hypotheses for 3d human pose estimation with mixture density network", in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 9887–9895.

[228] T. Wehrbein, M. Rudolph, B. Rosenhahn, and B. Wandt, "Probabilistic monocular 3d human pose estimation with normalizing flows", in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 11 199–11 208.

[229] A. Qammaz and A. Argyros, "Occlusion-tolerant and personalized 3d human pose estimation in rgb images", in *2020 25th International Conference on Pattern Recognition (ICPR)*, IEEE, 2021, pp. 6904–6911.

[230] Q. Liu, Y. Zhang, S. Bai, and A. Yuille, "Explicit occlusion reasoning for multi-person 3d human pose estimation", in *European Conference on Computer Vision*, Springer, 2022, pp. 497–517.

[231] C. Yu, B. Wang, B. Yang, and R. T. Tan, "Multi-scale networks for 3d human pose estimation with inference stage optimization", *arXiv preprint arXiv:2010.06844*, 2020.

[232] S. Banik, P. Gschoßmann, A. M. Garcia, and A. Knoll, "Occlusion robust 3d human pose estimation with stridedposegraphformer and data augmentation", *arXiv preprint arXiv:2304.12069*, 2023.

[233] A. A. Dursun and T. E. Tuncer, "Estimation of partially occluded 2d human joints with a bayesian approach", *Digital Signal Processing*, vol. 114, p. 103 056, 2021.

[234] J. N. Kundu, S. Seth, P. YM, V. Jampani, A. Chakraborty, and R. V. Babu, "Uncertainty-aware adaptation for self-supervised 3d human pose estimation", in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 20 448–20 459.

[235] B. Huang, Y. Shu, J. Ju, and Y. Wang, "Occluded human body capture with self-supervised spatial-temporal motion prior", *arXiv preprint arXiv:2207.05375*, 2022.

[236] M. Ghafoor and A. Mahmood, "Quantification of occlusion handling capability of 3d human pose estimation framework", *IEEE Transactions on Multimedia*, 2022.

[237] X. Guo and Y. Dai, "Occluded joints recovery in 3d human pose estimation based on distance matrix", in *2018 24th International Conference on Pattern Recognition (ICPR)*, IEEE, 2018, pp. 1325–1330.

[238] N.-G. Cho, A. L. Yuille, and S.-W. Lee, "Adaptive occlusion state estimation for human pose tracking under self-occlusions", *Pattern Recognition*, vol. 46, no. 3, pp. 649–661, 2013.

[239] Y. Yu, B. Yang, and P. C. Yuen, "Torso orientation: A new clue for occlusion-aware human pose estimation", in *2016 24th European Signal Processing Conference (EUSIPCO)*, IEEE, 2016, pp. 908–912.

[240] L. Fu, J. Zhang, and K. Huang, "Beyond tree structure models: A new occlusion aware graphical model for human pose estimation", in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1976–1984.

[241] L. Sigal and M. J. Black, "Measure locally, reason globally: Occlusion-sensitive articulated pose estimation", in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, IEEE, vol. 2, 2006, pp. 2041–2048.

[242] Y. Yang and G. Sundaramoorthi, "Modeling self-occlusions in dynamic shape and appearance tracking", in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 201–208.

[243] K. Lee, I. Lee, and S. Lee, "Propagating lstm: 3d pose estimation based on joint interdependency", in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 119–135.

[244] M. R. I. Hossain and J. J. Little, "Exploiting temporal information for 3d human pose estimation", in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 68–84.

[245] I. Sárándi, T. Linder, K. O. Arras, and B. Leibe, “Metrabs: Metric-scale truncation-robust heatmaps for absolute 3d human pose estimation”, *IEEE Transactions on Biometrics, Behavior, and Identity Science*, vol. 3, no. 1, pp. 16–30, 2020.

[246] J. Cha, M. Saqlain, G. Kim, M. Shin, and S. Baek, “Multi-person 3d pose and shape estimation via inverse kinematics and refinement”, in *European Conference on Computer Vision*, Springer, 2022, pp. 660–677.

[247] J. Li, C. Xu, Z. Chen, S. Bian, L. Yang, and C. Lu, “Hybrik: A hybrid analytical-neural inverse kinematics solution for 3d human pose and shape estimation”, in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 3383–3393.

[248] J. Li, S. Bian, Q. Liu, J. Tang, F. Wang, and C. Lu, “Niki: Neural inverse kinematics with invertible neural networks for 3d human pose and shape estimation”, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12 933–12 942.

[249] C. Doersch and A. Zisserman, “Sim2real transfer learning for 3d human pose estimation: Motion to the rescue”, *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[250] Z. Li, B. Xu, H. Huang, C. Lu, and Y. Guo, “Deep two-stream video inference for human body pose and shape estimation”, in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022, pp. 430–439.

[251] T. Alldieck, M. Kassubeck, B. Wandt, B. Rosenhahn, and M. Magnor, “Optical flow-based 3d human motion estimation from monocular video”, in *Pattern Recognition: 39th German Conference, GCPR 2017, Basel, Switzerland, September 12–15, 2017, Proceedings 39*, Springer, 2017, pp. 347–360.

[252] A. Jain, J. Tompson, Y. LeCun, and C. Bregler, “Modeep: A deep learning framework using motion features for human pose estimation”, in *Computer Vision–ACCV 2014: 12th Asian Conference on Computer Vision, Singapore, Singapore, November 1-5, 2014, Revised Selected Papers, Part II 12*, Springer, 2015, pp. 302–315.

[253] B. Ji, C. Yang, Y. Shunyu, and Y. Pan, “Hpof: 3d human pose recovery from monocular video with optical flow”, in *Proceedings of the 2021 International Conference on Multimedia Retrieval*, 2021, pp. 144–154.

[254] D. Zhang, G. Guo, D. Huang, and J. Han, “Poseflow: A deep motion representation for understanding human behaviors in videos”, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6762–6770.

[255] R. Feng, Y. Gao, X. Ma, T. H. E. Tse, and H. J. Chang, “Mutual information-based temporal difference learning for human pose estimation in video”, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 17 131–17 141.

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

[257] O. Arikan and D. A. Forsyth, "Interactive motion generation from examples", *ACM Transactions on Graphics (TOG)*, vol. 21, no. 3, pp. 483–490, 2002.

[258] L. Kovar, M. Gleicher, and F. Pighin, "Motion graphs", in *ACM SIGGRAPH 2008 classes*, 2008, pp. 1–10.

[259] J. Lee, J. Chai, P. S. Reitsma, J. K. Hodgins, and N. S. Pollard, "Interactive control of avatars animated with human motion data", in *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, 2002, pp. 491–500.

[260] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features", in *2005 IEEE international workshop on visual surveillance and performance evaluation of tracking and surveillance*, IEEE, 2005, pp. 65–72.

[261] I. Laptev, "On space-time interest points", *International journal of computer vision*, vol. 64, no. 2, pp. 107–123, 2005.

[262] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies", in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2008, pp. 1–8.

[263] P. Scovanner, S. Ali, and M. Shah, "A 3-dimensional sift descriptor and its application to action recognition", in *Proceedings of the 15th ACM international conference on Multimedia*, 2007, pp. 357–360.

[264] H. Wang and C. Schmid, "Action recognition with improved trajectories", in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 3551–3558.

[265] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints", in *Workshop on statistical learning in computer vision, ECCV*, Prague, vol. 1, 2004, pp. 1–2.

[266] I. S. Vicente, V. Kyrki, D. Kragic, and M. Larsson, "Action recognition and understanding through motor primitives", *Advanced Robotics*, vol. 21, no. 15, pp. 1687–1707, 2007.

[267] J. Kovač and P. Peer, "Human skeleton model based dynamic features for walking speed invariant gait recognition", *Mathematical Problems in Engineering*, vol. 2014, 2014.

[268] A. F. Bobick and A. Y. Johnson, "Gait recognition using static, activity-specific parameters", in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, IEEE, vol. 1, 2001, pp. I–I.

[269] K. Yamauchi, B. Bhanu, and H. Saito, "Recognition of walking humans in 3d: Initial results", in *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, IEEE, 2009, pp. 45–52.

[270] J. Preis, M. Kessel, M. Werner, and C. Linnhoff-Popien, "Gait recognition with kinect", in *1st international workshop on kinect in pervasive computing*, New Castle, UK, 2012, pp. 1–4.

[271] A. Behara and M Raghunadh, "Person recognition system using model based gaitface fusion technique", in *International Conference on Electrical Engineering and Computer Science (EECS 2013)*, 2013, pp. 65–70.

[272] M. D. Bengalur, "Human activity recognition using body pose features and support vector machine", in *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, IEEE, 2013, pp. 1970–1975.

[273] P. J. Phillips, S. Sarkar, I. Robledo, P. Grother, and K. Bowyer, "Baseline results for the challenge problem of humanid using gait analysis", in *Proceedings of Fifth IEEE International Conference on Automatic Face Gesture Recognition*, IEEE, 2002, pp. 137–142.

[274] N. V. Boulgouris and Z. X. Chi, "Human gait recognition based on matching of body components", *Pattern recognition*, vol. 40, no. 6, pp. 1763–1770, 2007.

[275] P. J. Phillips, S. Sarkar, I. Robledo, P. Grother, and K. Bowyer, "The gait identification challenge problem: Data sets and baseline algorithm", in *Object recognition supported by user interaction for service robots*, IEEE, vol. 1, 2002, pp. 385–388.

[276] L. Wang, W. Hu, and T. Tan, "A new attempt to gait-based human identification", in *Object recognition supported by user interaction for service robots*, IEEE, vol. 1, 2002, pp. 115–118.

[277] J. Wang, M. She, S. Nahavandi, and A. Kouzani, "A review of vision-based gait recognition methods for human identification", in *2010 international conference on digital image computing: techniques and applications*, IEEE, 2010, pp. 320–327.

[278] L. Wang, T. Tan, H. Ning, and W. Hu, "Silhouette analysis-based gait recognition for human identification", *IEEE transactions on pattern analysis and machine intelligence*, vol. 25, no. 12, pp. 1505–1518, 2003.

[279] F. Dadashi, B. N. Araabi, and H. Soltanian-Zadeh, "Gait recognition using wavelet packet silhouette representation and transductive support vector machines", in *2009 2nd International Congress on Image and Signal Processing*, IEEE, 2009, pp. 1–5.

[280] N. V. Boulgouris, K. N. Plataniotis, and D. Hatzinakos, "An angular transform of gait sequences for gait assisted recognition", in *2004 International Conference on Image Processing, 2004. ICIP'04.*, IEEE, vol. 2, 2004, pp. 857–860.

[281] A. Kale, A. Rajagopalan, N. Cuntoor, and V. Kruger, "Gait-based recognition of humans using continuous hmms", in *Proceedings of Fifth IEEE International Conference on Automatic Face Gesture Recognition*, IEEE, 2002, pp. 336–341.

[282] J Kang, B Badi, Y Zhao, and D. Wright, "Human motion modeling and simulation", in *6th International Conference on Robotics, Control and Manufacturing Technology (ROCOM 2006)*, 2006, pp. 62–67.

[283] V. M. Zatsiorsky and B. I. Prilutsky, *Biomechanics of skeletal muscles*. Human Kinetics, 2012.

[284] D. Cunado, M. S. Nixon, and J. N. Carter, "Using gait as a biometric, via phase-weighted magnitude spectra", in *International conference on audio-and video-based biometric person authentication*, Springer, 1997, pp. 93–102.

[285] ——, "Automatic extraction and description of human gait models for recognition purposes", *Computer vision and image understanding*, vol. 90, no. 1, pp. 1–41, 2003.

[286] M. P. Murray, "Gait as a total pattern of movement: Including a bibliography on gait", *American Journal of Physical Medicine & Rehabilitation*, vol. 46, no. 1, pp. 290–333, 1967.

[287] M. P. Murray, A. B. Drought, and R. C. Kory, "Walking patterns of normal men", *JBJS*, vol. 46, no. 2, pp. 335–360, 1964.

[288] C. Yam, M. S. Nixon, and J. N. Carter, "Automated person recognition by walking and running via model-based approaches", *Pattern recognition*, vol. 37, no. 5, pp. 1057–1072, 2004.

[289] J.-H. Yoo, D. Hwang, K.-Y. Moon, and M. S. Nixon, "Automated human recognition by gait using neural network", in *2008 First Workshops on Image Processing Theory, Tools and Applications*, IEEE, 2008, pp. 1–6.

[290] L. Wang, H. Ning, T. Tan, and W. Hu, "Fusion of static and dynamic body biometrics for gait recognition", *IEEE Transactions on circuits and systems for video technology*, vol. 14, no. 2, pp. 149–158, 2004.

[291] R. Tanawongsuwan and A. Bobick, "Gait recognition from time-normalized joint-angle trajectories in the walking plane", in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, IEEE, vol. 2, 2001, pp. II–II.

[292] A. Ball, D. Rye, F. Ramos, and M. Velonaki, "Unsupervised clustering of people from'skeleton'data", in *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, 2012, pp. 225–226.

[293] Y.-C. Lin, B.-S. Yang, Y.-T. Lin, and Y.-T. Yang, "Human recognition based on kinematics and kinetics of gait", *Journal of medical and biological engineering*, vol. 31, no. 4, pp. 255–263, 2011.

[294] M. Luštrek and B. Kaluža, "Fall detection and activity recognition with machine learning", *Informatica*, vol. 33, no. 2, 2009.

[295] J. Sedmidubsky and J. Valcik, "Retrieving similar movements in motion capture data", in *International Conference on Similarity Search and Applications*, Springer, 2013, pp. 325–330.

[296] M. Müller, T. Röder, and M. Clausen, "Efficient content-based retrieval of motion capture data", in *ACM SIGGRAPH 2005 Papers*, 2005, pp. 677–685.

[297] J. Xiao, Y. Zhuang, T. Yang, and F. Wu, "An efficient keyframe extraction from motion capture data", in *Computer Graphics International Conference*, Springer, 2006, pp. 494–501.

[298] W. Gong, A. D. Bagdanov, F. X. Roca, and J. Gonzalez, "Automatic key pose selection for 3d human action recognition", in *International Conference on Articulated Motion and Deformable Objects*, Springer, 2010, pp. 290–299.

[299] L. Ren, A. Patrick, A. A. Efros, J. K. Hodgins, and J. M. Rehg, "A data-driven approach to quantifying natural human motion", *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 1090–1097, 2005.

[300] B. Krüger, J. Tautges, M. Müller, and A. Weber, "Multi-mode tensor representation of motion data", *JVRB-Journal of Virtual Reality and Broadcasting*, vol. 5, no. 5, 2008.

[301] J. Sedmidubsky, J. Valcik, M. Balazia, and P. Zezula, "Gait recognition based on normalized walk cycles", in *International Symposium on Visual Computing*, Springer, 2012, pp. 11–20.

[302] J. Valcik, J. Sedmidubsky, M. Balazia, and P. Zezula, "Identifying walk cycles for human recognition", in *Pacific-Asia Workshop on Intelligence and Security Informatics*, Springer, 2012, pp. 127–135.

[303] X. Zhao, X. Li, C. Pang, X. Zhu, and Q. Z. Sheng, "Online human gesture recognition from motion data streams", in *Proceedings of the 21st ACM international conference on Multimedia*, 2013, pp. 23–32.

[304] E. P. Ijjina and C. K. Mohan, "Human action recognition based on mocap information using convolution neural networks", in *2014 13th international conference on machine learning and applications*, IEEE, 2014, pp. 159–164.

[305] M. Müller and T. Röder, "Motion templates for automatic classification and retrieval of motion capture data", in *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 2006, pp. 137–146.

[306] B. K. Horn and B. G. Schunck, "Determining optical flow", *Artificial intelligence*, vol. 17, no. 1-3, pp. 185–203, 1981.

[307] S. N. Vitaladevuni, V. Kellokumpu, and L. S. Davis, "Action recognition using ballistic dynamics", in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 2008, pp. 1–8.

[308] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf, "Parametric correspondence and chamfer matching: Two new techniques for image matching", SRI INTERNATIONAL MENLO PARK CA ARTIFICIAL INTELLIGENCE CENTER, Tech. Rep., 1977.

[309] M. Ahmad and S.-W. Lee, "Human action recognition using shape and clg-motion flow from multi-view image sequences", *Pattern Recognition*, vol. 41, no. 7, pp. 2237–2252, 2008.

[310] M.-K. Hu, "Visual pattern recognition by moment invariants", *IRE transactions on information theory*, vol. 8, no. 2, pp. 179–187, 1962.

[311] M. Azimi, "Skeletal joint smoothing white paper", *MSDN digital library*, 2012.

[312] T. Röder, "Similarity, retrieval, and classification of motion capture data", PhD thesis, Bonn Univ., Diss., 2007, 2006.

[313] T. T. Thanh, F. Chen, K. Kotani, and B. Le, "Automatic extraction of semantic action features", in *2013 International Conference on Signal-Image Technology & Internet-Based Systems*, IEEE, 2013, pp. 148–155.

[314] J. W. Davis and A. F. Bobick, "The representation and recognition of human movement using temporal templates", in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, 1997, pp. 928–934.

[315] J. Han and B. Bhanu, "Individual recognition using gait energy image", *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 2, pp. 316–322, 2005.

[316] J. Liu and N. Zheng, "Gait history image: A novel temporal template for gait recognition", in *2007 IEEE international conference on multimedia and expo*, IEEE, 2007, pp. 663–666.

[317] K. Bashir, T. Xiang, and S. Gong, "Feature selection on gait energy image for human identification", in *2008 IEEE international conference on acoustics, speech and signal processing*, IEEE, 2008, pp. 985–988.

[318] L. Chunli and W. Kejun, "A behavior classification based on enhanced gait energy image", in *2010 International Conference on Networking and Digital Society*, IEEE, vol. 2, 2010, pp. 589–592.

[319] X. Huang and N. V. Boulgouris, "Gait recognition using linear discriminant analysis with artificial walking conditions", in *2010 IEEE International Conference on Image Processing*, IEEE, 2010, pp. 2461–2464.

[320] ——, "Gait recognition with shifted energy image and structural feature extraction", *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 2256–2268, 2011.

[321] Q. Ma, S. Wang, D. Nie, and J. Qiu, "Recognizing humans based on gait moment image", in *Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD 2007)*, IEEE, vol. 2, 2007, pp. 606–610.

[322] E. Zhang, Y. Zhao, and W. Xiong, "Active energy image plus 2dlpp for gait recognition", *Signal Processing*, vol. 90, no. 7, pp. 2295–2302, 2010.

[323] C. Chen, J. Liang, H. Zhao, H. Hu, and J. Tian, "Frame difference energy image for gait recognition with incomplete silhouettes", *Pattern Recognition Letters*, vol. 30, no. 11, pp. 977–984, 2009.

[324] D. Weinland, R. Ronfard, and E. Boyer, "Free viewpoint action recognition using motion history volumes", *Computer vision and image understanding*, vol. 104, no. 2-3, pp. 249–257, 2006.

[325] S. Sivapalan, D. Chen, S. Denman, S. Sridharan, and C. Fookes, "Gait energy volumes and frontal gait recognition using depth images", in *2011 International Joint Conference on Biometrics (IJCB)*, IEEE, 2011, pp. 1–6.

[326] A. F. Bobick and J. W. Davis, "The recognition of human movement using temporal templates", *IEEE Transactions on pattern analysis and machine intelligence*, vol. 23, no. 3, pp. 257–267, 2001.

[327] F. A. Sadjadi and E. L. Hall, "Three-dimensional moment invariants", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 2, pp. 127–136, 1980.

[328] J. E. Boyd, "Video phase-locked loops in gait recognition", in *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, IEEE, vol. 1, 2001, pp. 696–703.

[329] L. Wang, T. Tan, W. Hu, and H. Ning, "Automatic gait recognition based on statistical shape analysis", *IEEE transactions on image processing*, vol. 12, no. 9, pp. 1120–1131, 2003.

[330] C. BenAbdelkader, R. Cutler, and L. Davis, "Stride and cadence as a biometric in automatic person identification and verification", in *Proceedings of Fifth IEEE international conference on automatic face gesture recognition*, IEEE, 2002, pp. 372–377.

[331] Y. Sheikh, M. Sheikh, and M. Shah, "Exploring the space of a human action", in *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, IEEE, vol. 1, 2005, pp. 144–149.

[332] M. Milovanović, M. Minović, and D. Starcević, "New gait recognition method using kinect stick figure and cbir", in *2012 20th Telecommunications Forum (TELFOR)*, IEEE, 2012, pp. 1323–1326.

[333] S. Carlsson, "Combinatorial geometry for shape representation and indexing", in *International Workshop on Object Representation in Computer Vision*, Springer, 1996, pp. 53–78.

[334] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis", *Chemometrics and intelligent laboratory systems*, vol. 2, no. 1-3, pp. 37–52, 1987.

[335] O. Arikan, "Compression of motion capture databases", in *ACM SIGGRAPH 2006 Papers*, 2006, pp. 890–897.

[336] A. Baak, M. Müeller, and H.-P. Seidel, "An efficient algorithm for keyframe-based motion retrieval in the presence of temporal deformations", in *Proceedings of the 1st ACM international conference on Multimedia information retrieval*, 2008, pp. 451–458.

[337] J. Leskovec, A. Rajaraman, and J. D. Ullman, *Mining of massive data sets*. Cambridge university press, 2020.

[338] W. Shen, K. Deng, X. Bai, T. Leyvand, B. Guo, and Z. Tu, "Exemplar-based human action pose correction", *IEEE transactions on cybernetics*, vol. 44, no. 7, pp. 1053–1066, 2013.

[339] M. Vrigkas, C. Nikou, and I. A. Kakadiaris, "A review of human activity recognition methods", *Frontiers in Robotics and AI*, vol. 2, p. 28, 2015.

[340] W. Lin, M.-T. Sun, R. Poovendran, and Z. Zhang, "Group event detection with a varying number of group members for video surveillance", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 8, pp. 1057–1067, 2010.

[341] I. Rodomagoulakis, N. Kardaris, V. Pitsikalis, E Mavroudi, A. Katsamanis, A. Tsiami, and P. Maragos, "Multimodal human action recognition in assistive human-robot interaction", in *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, 2016, pp. 2702–2706.

[342] L. Piyathilaka and S. Kodagoda, "Human activity recognition for domestic robots", in *Field and Service Robotics*, Springer, 2015, pp. 395–408.

[343] J. K. Tang, J. C. Chan, and H. Leung, "Interactive dancing game with real-time recognition of continuous dance moves from 3d human motion capture", in *Proceedings of the 5th international conference on ubiquitous information management and communication*, 2011, pp. 1–9.

[344] H. Nicolau, T. Guerreiro, R. Pereira, D. Gonçalves, and J. Jorge, "Computer-assisted rehabilitation: Towards effective evaluation", *International Journal of Cognitive Performance Support*, vol. 1, no. 1, pp. 11–26, 2013.

[345] H. Ghasemzadeh and R. Jafari, "Coordination analysis of human movements with body sensor networks: A signal processing model to evaluate baseball swings", *IEEE Sensors Journal*, vol. 11, no. 3, pp. 603–610, 2010.

[346] D. S. Alexiadis and P. Daras, "Quaternionic signal processing techniques for automatic evaluation of dance performances from mocap data", *IEEE Transactions on Multimedia*, vol. 16, no. 5, pp. 1391–1406, 2014.

[347] D. Y. Kwon and M. Gross, "Combining body sensors and visual sensors for motion training", in *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology*, 2005, pp. 94–101.

[348] C. Velardo and J.-L. Dugelay, "Real time extraction of body soft biometric from 3d videos", in *Proceedings of the 19th ACM international conference on Multimedia*, 2011, pp. 781–782.

[349] P. Elias, J. Sedmidubsky, and P. Zezula, "Motion images: An effective representation of motion capture data for similarity search", in *International Conference on Similarity Search and Applications*, Springer, 2015, pp. 250–255.

[350] J. P. Varkey, D. Pompili, and T. A. Walls, "Human motion recognition using a wireless sensor-based wearable system", *Personal and Ubiquitous Computing*, vol. 16, no. 7, pp. 897–910, 2012.

[351] V. Andersson, R. Dutra, and R. Araújo, "Anthropometric and human gait identification using skeleton data from kinect sensor", in *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, 2014, pp. 60–61.

[352] E. M. Tapia, S. S. Intille, W. Haskell, K. Larson, J. Wright, A. King, and R. Friedman, "Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor", in *2007 11th IEEE international symposium on wearable computers*, IEEE, 2007, pp. 37–40.

[353] F. Lv and R. Nevatia, "Recognition and segmentation of 3-d human action using hmm and multi-class adaboost", in *European conference on computer vision*, Springer, 2006, pp. 359–372.

[354] T. H. Thi, J. Zhang, L. Cheng, L. Wang, and S. Satoh, "Human action recognition and localization in video using structured learning of local space-time features", in *2010 7th IEEE International Conference on Advanced Video and Signal Based Surveillance*, IEEE, 2010, pp. 204–211.

[355] A. Sundaresan, A. RoyChowdhury, and R. Chellappa, "A hidden markov model based framework for recognition of humans from gait sequences", in *Proceedings 2003 International Conference on Image Processing (Cat. No. 03CH37429)*, IEEE, vol. 2, 2003, pp. II–93.

[356] D. Zhang, Y. Wang, and B. Bhanu, "Age classification base on gait using hmm", in *2010 20th International Conference on Pattern Recognition*, IEEE, 2010, pp. 3834–3837.

[357] Y. Liang, W. Lu, W. Liang, and Y. Wang, "Action recognition using local joints structure and histograms of 3d joints", in *2014 Tenth International Conference on Computational Intelligence and Security*, IEEE, 2014, pp. 185–188.

[358] L. Miranda, T. Vieira, D. Martinez, T. Lewiner, A. W. Vieira, and M. F. Campos, "Real-time gesture recognition from depth data through key poses learning and decision forests", in *2012 25th SIBGRAPI Conference on Graphics, Patterns and Images*, IEEE, 2012, pp. 268–275.

[359] H. Kataoka, K. Hashimoto, and Y. Aoki, "Feature integration with random forests for real-time human activity recognition", in *Seventh International Conference on Machine Vision (ICMV 2014)*, SPIE, vol. 9445, 2015, pp. 23–27.

[360] M. Müller, *Information retrieval for music and motion*. Springer, 2007, vol. 2.

[361] J. P. Park, K. H. Lee, and J. Lee, "Finding syntactic structures from human motion data", in *Computer Graphics Forum*, Wiley Online Library, vol. 30, 2011, pp. 2183–2193.

[362] J. Valcik, J. Sedmidubsky, and P. Zezula, "Improving kinect-skeleton estimation", in *International Conference on Advanced Concepts for Intelligent Vision Systems*, Springer, 2015, pp. 575–587.

[363] ——, "Assessing similarity models for human-motion retrieval applications", *Computer Animation and Virtual Worlds*, vol. 27, no. 5, pp. 484–500, 2016.

[364] E. Keogh, "Efficiently finding arbitrarily scaled patterns in massive time series databases", in *European Conference on Principles of Data Mining and Knowledge Discovery*, Springer, 2003, pp. 253–265.

[365] D. J. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series.", in *KDD workshop*, Seattle, WA, USA: vol. 10, 1994, pp. 359–370.

[366] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks", in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4489–4497.

[367] L. Sun, K. Jia, D.-Y. Yeung, and B. E. Shi, "Human action recognition using factorized spatio-temporal convolutional networks", in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4597–4605.

[368] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 2625–2634.

[369] Z. Li, K. Gavrilyuk, E. Gavves, M. Jain, and C. G. Snoek, "Videolstm convolves, attends and flows for action recognition", *Computer Vision and Image Understanding*, vol. 166, pp. 41–50, 2018.

[370] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos", *Advances in neural information processing systems*, vol. 27, 2014.

[371] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1933–1941.

[372] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt, "Spatio-temporal convolutional sparse auto-encoder for sequence classification.", in *BMVC*, Citeseer, 2012, pp. 1–12.

[373] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks", in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2014, pp. 1725–1732.

[374] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition", *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 221–231, 2012.

[375] G. Varol, I. Laptev, and C. Schmid, "Long-term temporal convolutions for action recognition", *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 6, pp. 1510–1517, 2017.

[376] P. Molchanov, S. Gupta, K. Kim, and J. Kautz, "Hand gesture recognition with 3d convolutional neural networks", in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2015, pp. 1–7.

[377] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4694–4702.

[378] G. Gkioxari and J. Malik, "Finding action tubes", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 759–768.

[379] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid, "Deepflow: Large displacement optical flow with deep matching", in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 1385–1392.

[380] Y. Poleg, A. Ephrat, S. Peleg, and C. Arora, "Compact cnn for indexing egocentric videos", in *2016 IEEE winter conference on applications of computer vision (WACV)*, IEEE, 2016, pp. 1–9.

[381] G. Chéron, I. Laptev, and C. Schmid, "P-cnn: Pose-based cnn features for action recognition", in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3218–3226.

[382] L. Pigou, S. Dieleman, P.-J. Kindermans, and B. Schrauwen, "Sign language recognition using convolutional neural networks", in *European Conference on Computer Vision*, Springer, 2014, pp. 572–578.

[383] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville, "Describing videos by exploiting temporal structure", in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4507–4515.

[384] L. Pigou, A. Van Den Oord, S. Dieleman, M. Van Herreweghe, and J. Dambre, "Beyond temporal pooling: Recurrence and temporal convolutions for gesture recognition in video", *International Journal of Computer Vision*, vol. 126, no. 2, pp. 430–439, 2018.

[385] S. Sharma, R. Kiros, and R. Salakhutdinov, "Action recognition using visual attention", *arXiv preprint arXiv:1511.04119*, 2015.

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

[387] A. Paiement, L. Tao, S. Hannuna, M. Camplani, D. Damen, and M. Mirmehdi, "Online quality assessment of human movement from skeleton data", in *British Machine Vision Conference*, BMVA press, 2014, pp. 153–166.

[388] L. Tao, A. Paiement, D. Damen, M. Mirmehdi, S. Hannuna, M. Camplani, T. Burghardt, and I. Craddock, "A comparative study of pose representation and dynamics modelling for online motion quality assessment", *Computer vision and image understanding*, vol. 148, pp. 136–152, 2016.

[389] A. Elkholy, M. Hussein, W. Gomaa, D. Damen, and E. Saba, "Efficient and robust skeleton-based quality assessment and abnormality detection in human action performance", *IEEE journal of biomedical and health informatics*, 2019.

[390] A. Vakanski, H.-p. Jun, D. Paul, and R. Baker, "A data set of human body movements for physical rehabilitation exercises", *Data*, vol. 3, no. 1, p. 2, 2018.

[391] Y. Liao, A. Vakanski, and M. Xian, "A deep learning framework for assessment of quality of rehabilitation exercises", *arXiv preprint arXiv:1901.10435*, 2019.

[392] H. Pirsiavash, C. Vondrick, and A. Torralba, "Assessing the quality of actions", in *European Conference on Computer Vision*, Springer, 2014, pp. 556–571.

[393] P. Parmar and B. Morris, "Action quality assessment across multiple actions", in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, 2019, pp. 1468–1476.

[394] V. Venkataraman, I. Vlachos, and P. K. Turaga, "Dynamical regularity for action analysis.", in *BMVC*, 2015, pp. 67–1.

[395] P. Parmar and B. Tran Morris, "Learning to score olympic events", in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 20–28.

[396] C. Xu, Y. Fu, B. Zhang, Z. Chen, Y.-G. Jiang, and X. Xue, "Learning to score figure skating sport videos", *IEEE Transactions on Circuits and Systems for Video Technology*, 2019.

[397] X. Xiang, Y. Tian, A. Reiter, G. D. Hager, and T. D. Tran, "S3d: Stacking segmental p3d for action quality assessment", in *2018 25th IEEE International Conference on Image Processing (ICIP)*, IEEE, 2018, pp. 928–932.

[398] Y. Li, X. Chai, and X. Chen, "End-to-end learning for action quality assessment", in *Pacific Rim Conference on Multimedia*, Springer, 2018, pp. 125–134.

[399] ——, "Scoringnet: Learning key fragment for action quality assessment with ranking loss in skilled sports", in *Asian Conference on Computer Vision*, Springer, 2018, pp. 149–164.

[400] Y. Gao, S. S. Vedula, C. E. Reiley, N. Ahmidi, B. Varadarajan, H. C. Lin, L. Tao, L. Zappella, B. Béjar, D. D. Yuh, *et al.*, "Jhu-isi gesture and skill assessment working set (jigsaws): A surgical activity dataset for human motion modeling", in *MICCAI Workshop: M2CAI*, vol. 3, 2014, p. 3.

[401] N. Ahmidi, L. Tao, S. Sefati, Y. Gao, C. Lea, B. B. Haro, L. Zappella, S. Khudanpur, R. Vidal, and G. D. Hager, "A dataset and benchmarks for segmentation and recognition of gestures in robotic surgery", *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 9, pp. 2025–2041, 2017.

[402] Z. Wang and A. M. Fey, "Deep learning with convolutional neural network for objective skill evaluation in robot-assisted surgery", *International journal of computer assisted radiology and surgery*, vol. 13, no. 12, pp. 1959–1970, 2018.

[403] I. Funke, S. T. Mees, J. Weitz, and S. Speidel, "Video-based surgical skill assessment using 3d convolutional neural networks", *International journal of computer assisted radiology and surgery*, vol. 14, no. 7, pp. 1217–1225, 2019.

[404] H. Doughty, D. Damen, and W. Mayol-Cuevas, "Who's better? who's best? pairwise deep ranking for skill determination", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6057–6066.

[405] Z. Li, Y. Huang, M. Cai, and Y. Sato, "Manipulation-skill assessment from videos with spatial attention network", *arXiv preprint arXiv:1901.02579*, 2019.

[406] M. J. Fard, S. Ameri, R Darin Ellis, R. B. Chinnam, A. K. Pandya, and M. D. Klein, "Automated robot-assisted surgical skill evaluation: Predictive analytics approach", *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 14, no. 1, e1850, 2018.

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

[408] G. Forestier, F. Petitjean, P. Senin, F. Despinoy, and P. Jannin, “Discovering discriminative and interpretable patterns for surgical motion analysis”, in *Conference on Artificial Intelligence in Medicine in Europe*, Springer, 2017, pp. 136–145.

[409] A. Zia and I. Essa, “Automated surgical skill assessment in rmis training”, *International journal of computer assisted radiology and surgery*, vol. 13, no. 5, pp. 731–739, 2018.

[410] M. Andriluka, U. Iqbal, E. Insafutdinov, L. Pishchulin, A. Milan, J. Gall, and B. Schiele, “Posetrack: A benchmark for human pose estimation and tracking”, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5167–5176.

[411] A. Kanazawa, J. Y. Zhang, P. Felsen, and J. Malik, “Learning 3d human dynamics from video”, in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5614–5623.

[412] M. Kaufmann, J. Song, C. Guo, K. Shen, T. Jiang, C. Tang, J. J. Zárate, and O. Hilliges, “Emdb: The electromagnetic database of global 3d human pose and shape in the wild”, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 14 632–14 643.

[413] `https://bit.ly/49OPGi3`.

[414] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization”, *arXiv preprint arXiv:1412.6980*, 2014.

[415] W. Zhu, X. Ma, Z. Liu, L. Liu, W. Wu, and Y. Wang, “Motionbert: A unified perspective on learning human motion representations”, in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 15 085–15 099.

[416] M. Tits, S. Laraba, E. Caulier, J. Tilmanne, and T. Dutoit, “Umons-taichi: A multimodal motion capture dataset of expertise in taijiquan gestures”, *Data in brief*, vol. 19, pp. 1214–1221, 2018.

[417] O. H. Schmitt, “A thermionic trigger”, *Journal of Scientific instruments*, vol. 15, no. 1, p. 24, 1938.

[418] M. Tits, “Expert gesture analysis through motion capture using statistical modeling and machine learning”, PhD thesis, Ph. D. Dissertation, 2018.

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

[420] C Duclos, P Desjardins, S Nadeau, A Delisle, D Gravel, B Brouwer, and H Corriveau, “Destabilizing and stabilizing forces to assess equilibrium during everyday activities”, *Journal of biomechanics*, vol. 42, no. 3, pp. 379–382, 2009.

[421] F. Multon, “Sensing human walking: Algorithms and techniques for extracting and modeling locomotion”, in *Human walking in virtual environments*, Springer, 2013, pp. 177–197.

[422] É. Caulier, *Understanding Taijiquan*. Editions Modulaires Européennes Inter-Communication SPRL, 2010, vol. 1.

[423] Y. Zhang, M. Hassan, H. Neumann, M. J. Black, and S. Tang, "Generating 3d people in scenes without people", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6194–6204.

[424] S. Li, L. Ke, K. Pratama, Y.-W. Tai, C.-K. Tang, and K.-T. Cheng, "Cascaded deep monocular 3d human pose estimation with evolutionary training data", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 6173–6183.

[425] J. Dong, W. Jiang, Q. Huang, H. Bao, and X. Zhou, "Fast and robust multi-person 3d pose estimation from multiple views", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7792–7801.

[426] C. Zimmermann, T. Welschehold, C. Dornhege, W. Burgard, and T. Brox, "3d human pose estimation in rgbd images for robotic task learning", in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2018, pp. 1986–1992.

[427] N. Ghorbani and M. J. Black, "Soma: Solving optical marker-based mocap automatically", in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 117–11 126.

[428] J. Dong, Q. Fang, W. Jiang, Y. Yang, Q. Huang, H. Bao, and X. Zhou, "Fast and robust multi-person 3d pose estimation and tracking from multiple views", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[429] Ł. Kidziński, B. Yang, J. L. Hicks, A. Rajagopal, S. L. Delp, and M. H. Schwartz, "Deep neural networks enable quantitative movement analysis using single-camera videos", *Nature communications*, vol. 11, no. 1, pp. 1–10, 2020.