Fortschritt-Berichte VDI

VDI

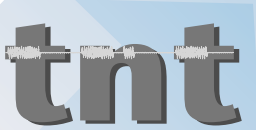
Reihe 10

Informatik/
Kommunikation

Timo von Marcard, M. Sc.,
Hemmingen

Nr. 866

Human Motion Capture with Sparse Inertial Sensors and Video



Institut für Informationsverarbeitung
www.tnt.uni-hannover.de

Human Motion Capture with Sparse Inertial Sensors and Video

Von der Fakultät für Elektrotechnik und Informatik
der Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des akademischen Grades
Doktor-Ingenieur
(abgekürzt: Dr.-Ing.)
genehmigte Dissertation

von

Timo von Marcard, M. Sc.
geboren am 31. März 1984 in Gießen, Deutschland

2019

1. Referent:	Prof. Dr.-Ing. Bodo Rosenhahn
2. Referent:	Prof. Dr.-Ing. Marcus Magnor
Vorsitzender:	Prof. Dr.-Ing. Jörn Ostermann

Tag der Promotion: 16. Oktober 2019

Fortschritt-Berichte VDI

Reihe 10

Informatik/
Kommunikation

Timo von Marcard, M. Sc.,
Hemmingen

Nr. 866

Human Motion Capture with Sparse Inertial Sensors and Video

Institut für Informationsverarbeitung
www.tnt.uni-hannover.de

von Marcard, Timo
Human Motion Capture with Sparse Inertial Sensors and Video
Fortschr.-Ber. VDI Reihe 10 Nr. 866. Düsseldorf: VDI Verlag 2019.
124 Seiten, 47 Bilder, 14 Tabellen.
ISBN 978-3-18-386610-6, ISSN 0178-9627,
€ 48,00/VDI-Mitgliederpreis € 43,20.

Keywords: Human Pose Estimation – Inertial Sensors – Video – Non-static Camera – Model-based Optimization – Sparse Sensors

This thesis explores approaches to capture human motions with a small number of sensors. In the first part of this thesis an approach is presented that reconstructs the body pose from only six inertial sensors. Instead of relying on pre-recorded motion databases, a global optimization problem is solved to maximize the consistency of measurements and model over an entire recording sequence. The second part of this thesis deals with a hybrid approach to fuse visual information from a single hand-held camera with inertial sensor data. First, a discrete optimization problem is solved to automatically associate people detections in the video with inertial sensor data. Then, a global optimization problem is formulated to combine visual and inertial information. The proposed approach enables capturing of multiple interacting people and works even if many more people are visible in the camera image. In addition, systematic inertial sensor errors can be compensated, leading to a substantial increase in accuracy.

Bibliographische Information der Deutschen Bibliothek
Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliographie; detaillierte bibliographische Daten sind im Internet unter www.dnb.de abrufbar.

Bibliographic information published by the Deutsche Bibliothek
(German National Library)
The Deutsche Bibliothek lists this publication in the Deutsche Nationalbibliographie (German National Bibliography); detailed bibliographic data is available via Internet at www.dnb.de.

Als Manuskript gedruckt. Printed in Germany.
ISSN 0178-9627
ISBN 978-3-18-386610-6

Acknowledgments

This thesis was written in the course of my activity as a scientific research assistant at the *Institut für Informationsverarbeitung* of the Leibniz University Hannover.

First of all, I would like to thank my doctoral advisor Prof. Dr.-Ing. Bodo Rosenhahn for the opportunity to work in his group and for his excellent supervision, support, and encouragement. Also, many thanks to him and Prof. Dr.-Ing. Jörn Ostermann for the great working conditions at the institute.

I am also very grateful to Dr.-Ing. Gerard Pons-Moll for many inspiring discussions and his efforts to make me grow as a researcher. His ideas and experience contributed a lot to make this thesis a success.

I thank Prof. Dr.-Ing. Marcus Magnor for being the second examiner of this thesis and Prof. Dr.-Ing. Jörn Ostermann for being the chair of the defense committee.

Special thanks go to all my colleagues at the *Institut für Informationsverarbeitung*. I had a fantastic time at the institute. Specifically, I would like to thank the administrative staff for all the support in technical and administrative matters. A special thanks also goes to my office mate and friend Dipl.-Math. Roberto Henschel for substantially improving my math skills and for making our office such a great place to waste time.

Finally, I would like to thank my family. Every single day, my wife Kathrin and my children Leni and Minna show me that there are more important things in life than work. Also, I would like to thank my parents for their unconditional support.

Contents

Acronyms

2D	*two-dimensional*
3D	*three-dimensional*
3DPW	*three-dimensional poses in the wild*
CNN	*Convolutional Neural Network*
DoF	*Degrees of Freedom*
GPS	*Global Positioning System*
IMU	*Inertial Measurement Unit*
MEMS	*Micro-Electro-Mechanical Systems*
MoCap	*Motion Capture*
MPJAE	*Mean Per Joint Angular Error*
MPJPE	*Mean Per Joint Position Error*
NN	*Neural Network*
RMS	*Root Mean Square*
SIP	*Sparse Inertial Poser*
SMPL	*Skinned Multi-Person Linear*
SO(3)	*Special Orthogonal Group of dimension three*
SE(3)	*Special Euclidean Group of dimension three*
VIP	*Video Inertial Poser*

Notation

Numbers and Arrays

a	A scalar (integer or real)
$\mathbf{a}$	A vector
$\mathbf{A}$	A matrix
$\mathbf{A}^T$	Transpose of matrix $\mathbf{A}$
$\langle \mathbf{a}, \mathbf{b} \rangle$	Scalar product of $\mathbf{a}$ and $\mathbf{b}$
$\mathbf{a} \times \mathbf{b}$	Cross product of $\mathbf{a}$ and $\mathbf{b}$
$\frac{\partial y}{\partial x}$	Partial derivative of y with respect to x
$\frac{\partial f}{\partial \mathbf{x}}$	Jacobian matrix $\mathbf{J} \in \mathbb{R}^{m \times n}$ of $f \colon \mathbb{R}^n \to \mathbb{R}^m$
$\bar{\mathbf{a}}$	Homogeneous representation of vector $\mathbf{a}$
$\tilde{\mathbf{a}}$	Estimate of vector quantity $\mathbf{a}$
$\|\|\mathbf{a}\|\|$	L^2-norm of $\mathbf{a}$
$\det(\mathbf{A})$	Determinant of $\mathbf{A}$
$\mathbf{I}_n$	Identity matrix with n rows and n columns
$\mathbf{0}_{n \times m}$	Zero matrix with n rows and m columns

Symbols

θ	A scalar angle
$\mathbf{x}$	Pose vector parametrizing a kinematic chain
δ	Gradient or perturbation of a pose vector
$\mathcal{C}$	A kinematic chain
$\mathrm{Pa}_{\mathcal{C}}(b)$	Parent joints of segment b in $\mathcal{C}$

$\mathbf{R}$	A rotation matrix $\mathbf{R} \in SO(3)$
$\mathbf{M}$	A matrix representing a rigid body motion $\mathbf{M} \in SE(3)$
$\pi(\mathbf{a})$	Projection of a 3D point $\mathbf{a}$ to homogeneous pixel coordinates
$\mathbf{K}$	Matrix of camera intrinsics
$\mathcal{N}(\mu, \Sigma)$	Gaussian distribution with mean μ and covariance Σ
$\oplus$	Perturbation-operator ($\oplus \colon G \times \mathfrak{g} \to G$)

Exponential Mapping

G	A Lie group
$\mathfrak{g}$	A Lie algebra
$\mathbf{G}_i$	Generator matrix associated to dimension i of a Lie algebra
$\widehat{\mathbf{a}}$ or $\mathbf{a}^\wedge$	Wedge-operator to construct a Lie algebra element from a coordinate vector $\mathbf{a}$
$\mathbf{A}^\vee$	Vee-operator to obtain coordinate vector from an element of a Lie algebra
exp	Matrix exponential to map from a Lie Algebra element to a Lie Group element
log	Logarithm to map from a Lie Group element to a Lie Algebra element

Sets and Graphs

$\mathbb{R}$	The set of real numbers
$\mathcal{G}$	A graph
$v \in \mathcal{V}$	A vertex v in a vertex set $\mathcal{V}$
$e \in \mathcal{E}$	An edge e in an edge set $\mathcal{E}$
c	A cost variable
$\mathcal{F}$	Feasibility set
$l \in \mathcal{L}$	A label l in a label set $\mathcal{L}$
x	A binary indicator variable
$\mathcal{H}$	An assignment hypothesis

Abstract

This dissertation explores approaches to capture human motions with a small number of sensors. Conventional methods either use a large number of static cameras, which severely limits the recording space, or a high number of body-worn inertial sensors, which is intrusive and only accurate for short time periods.

The first part of this thesis presents an approach that reconstructs the body pose from only 6 inertial sensors. Conventionally, up to 17 sensors are needed to cover all degrees of freedom of the body. Since fewer sensors inevitably lead to ambiguities, previous approaches estimate the missing information from pre-recorded motion databases. In contrast, in this work a model-based approach is proposed. More specifically, a global optimization problem is solved to maximize the consistency of measurements and model over an entire recording sequence. A key observation is that the kinematic constraints imposed by a statistical human body model constrain the search space significantly. This allows to utilize the acceleration data of inertial sensors to compensate for the missing sensor information. The performance of the method is demonstrated in challenging outdoor scenarios and accuracy is evaluated on two benchmark datasets.

The second part of this thesis deals with a hybrid approach to fuse visual information from a single hand-held camera with inertial sensor data. This approach combines the advantages of both sensor modalities. It enables capturing multiple interacting people and works even if many more people are visible in the camera image. In addition, systematic errors of the inertial sensors can be compensated, leading to a substantial increase in accuracy. In order to fuse the sensor modalities, visual information from the camera has to be associated to inertial sensor data. This is done automatically by formulating a discrete graph labeling problem. Subsequently, all sensor information of an entire tracking sequence is transformed into a global model-based optimization problem, which reconstructs body poses, camera pose and sensor errors. In several experiments accuracy is evaluated quantitatively and qualitatively. The combination of a single hand-held camera and body-worn inertial sensors enables motion capture in new complex settings. Using the approach a variety of motions are recorded, e.g. during shopping in a crowded pedestrian zone or during a bus ride. These recordings are composed into a novel dataset, which was made publicly available for research purposes.

Keywords: Human Pose Estimation, Inertial Sensors, Video, Non-static Camera, Model-based Optimization, Sparse Sensors

Kurzfassung

Diese Dissertation untersucht Ansätze zur Erfassung menschlicher Bewegungen mit wenigen Sensoren. Herkömmliche Verfahren verwenden entweder eine große Anzahl an statischen Kameras, was den Aufnahmebereich stark einschränkt, oder eine hohe Anzahl am Körper getragenen Inertialsensoren, was als unangenehm empfunden wird und nur für kurze Zeiträume präzise funktioniert.

Im ersten Teil dieser Arbeit wird ein Ansatz vorgestellt, der die Körperhaltung aus den Messdaten von nur 6 Inertialsensoren rekonstruiert. Üblicherweise werden bis zu 17 Sensoren benötigt um alle Freiheitsgrade des Körpers abzudecken. Da weniger Sensoren zwangsläufig zu Uneindeutigkeiten führen, werden in bisherigen Ansätzen die fehlenden Informationen aus zuvor aufgenommenen Bewegungsdatenbanken geschätzt. Im Gegensatz dazu wird ein modellbasierter, generativer Ansatz entwickelt. Sämtliche Messwerte eine Aufnahmesequenz werden in ein globales Optimierungsproblem überführt und die Konsistenz von Messdaten und Modell maximiert. Die modellierten kinematischen Einschränkungen des menschlichen Skelettes führen zu einer wesentlichen Eingrenzung des Suchraums und ermöglichen so die Beschleunigungsdaten der Inertialsensoren zur Kompensation der fehlenden Sensorinformationen heranzuziehen. Die Präzision des Ansatzes wird experimentell untersucht und durch Bewegungsrekonstruktionen aus anspruchsvollen Außenaufnahmen demonstriert.

Im zweiten Teil der Arbeit wird der vorhergehende Ansatz erweitert, um visuelle Informationen von einer in der Hand gehaltenen Smartphone-Kamera mit den Daten der Inertialsensoren zu fusionieren. Dieser Ansatz ermöglicht eine mobile Bewegungserfassung von mehreren interagierenden Personen und funktioniert selbst wenn im Kamerabild viele weitere Personen sichtbar sind. Zusätzlich können systematische Fehler der Inertialsensoren geschätzt werden, was zu einer erheblich präziseren Bewegungsschätzung führt. Um die verschiedenen Sensorinformation miteinander zu fusionieren, muss zunächst eine Zuordnung von Bildinformationen und Inertialsensordaten stattfinden. Diese Zuordnung wird zeitlich konsistent durch eine diskrete Optimierung mittels Graph-Labeling gelöst. Anschließend werden sämtliche Sensorinformationen einer gesamten Sequenz in ein globales Optimierungsproblem überführt und neben der Körperhaltung nun auch die relative Entfernung zur Kamera, die Kamerapose und Sensorfehler geschätzt. Die Präzision des Ansatzes wird in zahlreichen Experimenten evaluiert. Zusätzlich werden die im Rahmen der Arbeit aufgenommenen Bewegungssequenzen in Form eines neuartigen Datensatzes vorgestellt und für Forschungszwecke bereitgestellt. Die Kombination von Smartphone-Kamera und Inertialsensoren ermöglicht erstmalig eine mobile Bewegungserfassung von mehreren Personen, die auch für Alltagssituationen wie beispielsweise beim Einkaufen in einer belebten Fußgängerzone geeignet ist.

Schlagwörter: Erfassung menschlicher Bewegungen, Inertialsensoren, Video, bewegliche Kamera, modell-basierte Optimierung, wenige Sensoren

1 Introduction

As humans, we constantly perceive the movements of others with our eyes. Over the course of our lives, we have learned to interpret these observations in various ways. On the one hand, we develop an exceptional understanding of moments and forces that generate a motion, e.g. we can easily sense the physical condition of person by simply looking at his or her movements. On the other hand, we are also able to recognize moods and attitudes, e.g. if someone is happy, sad, relaxed, nervous, aggressive, etc. In fact, body language plays a fundamental role in human communication.

Consequently, there is a lot of information in our movements and this has many potential applications. Human motions are analyzed to diagnose pathological conditions or to better understand the biomechanics of our musculoskeletal system. Human motions are also used for animating virtual characters in movies or games. This enables to naturally transport moods even for characters who do not look like a human at all. Furthermore, human motion is an important input to man-machine and human-computer interaction.

A prerequisite for such applications is the ability to capture human motions and to transfer them to a numeric representation. This task is commonly called human motion capture[1] or simply *Motion Capture* (MoCap).

1.1 A Brief History

This section provides a very brief overview of the origins of human motion capture. For a more comprehensive review, we refer the reader to Rosenhahn *et al.* [1].

There has always been a great interest in capturing, visualizing and analyzing how humans move and interact with their environment. Probably the earliest art works visualizing humans in motion date back to 20000 years before present,

[1] In this work human motion capture refers to the task of reconstructing the body pose, rather than hand gestures or facial expressions.

Figure 1.1: The earliest art works showing humans in motion: cavemen drew hunting scenes at rock walls [2].

where cavemen created paintings of hunting scenes, see Figure 1.1. While artists commonly deal with the process of visualization, mankind has also always been interested in understanding and finding physical laws, that explain what we observe. A famous example who started to investigate the human body and the underlying physical principles is the polymath Leonardo DaVinci (1452-1519). His drawing *The Vitruvian Man*, depicted in Figure 1.2, can be seen as an early work to the research field of biomechanics and motion analysis.

Figure 1.2: The vitruvian man, drawn by the Italian polymath Leonardo DaVinci around 1490, shows the correlations of ideal human body proportions [3]. It demonstrates the increasing interest to understand the physical laws of nature and of the human body and its movements in particular.

In order to objectify the analysis of human motion, researchers and engineers started to develop motion measurement tools that enable to record and replay human motions. The first technical realization of a motion capture system was invented by

Muybridge (1830-1904), see Figure 1.3. He built an apparatus to capture images at high frame rates, which enabled to replay and review the motions after they have been observed.

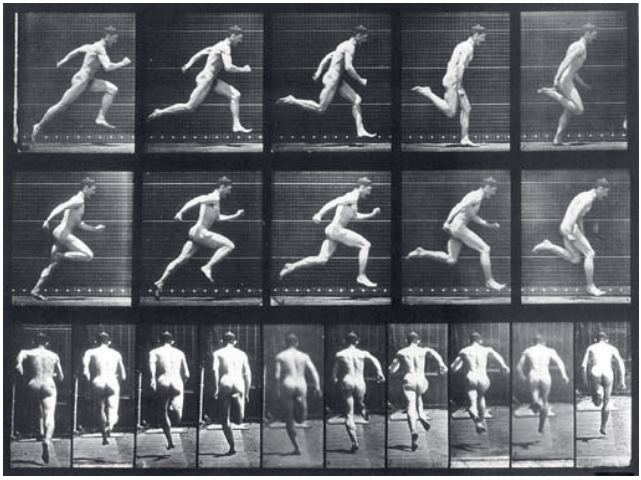


Figure 1.3: The first technical motion capture system was invented by Muybridge (1830-1904). His system was capable to take image at high frame rates as illustrated in the shown sequence of a running man [4].

From then on, various motion capture technologies were developed and nowadays they are not limited to recording, but are also capable to automatically reconstruct a numerical representation of the body motion. These numerical representations are key to various applications.

1.2 Applications

Capturing and reconstructing the body pose has many applications.In bio-medical and bio-mechanical applications, the focus is on *analyzing* the motions to diagnose, adjust or optimize the human musculoskeletal system. For example, in medical diagnoses, motion capture systems are used to assess physical activity, diagnose pathological conditions or to monitor rehabilitation progress. In sports science, they are used to optimize athletes' movements or to find optimal equipment. MoCap systems are also utilized to evaluate ergonomic aspects of products or workstations and to remotely monitor sport or rehabilitation exercises.

In addition to analyzing purposes, motion capture systems are also widely used to *animate* virtual characters or for *human-machine interactions*. Notable movies, such as Avatar, Lord of the Rings, Pirates of the Caribbeans and many more, use motion capture technology to transfer the movements of an actor to virtual avatars. The gaming industry applies the player's motion to control the gameplay or to realistically animate virtual characters. In the field of collaborative robots, interacting persons have to be monitored to prevent accidents. Also, virtual and augmented reality applications require accurate knowledge about the human pose to control and interact with virtual objects or environments. A prerequisite for all these applications is to accurately capture human motion. However, this poses a challenging problem as described in the following.

1.3 The MoCap Problem

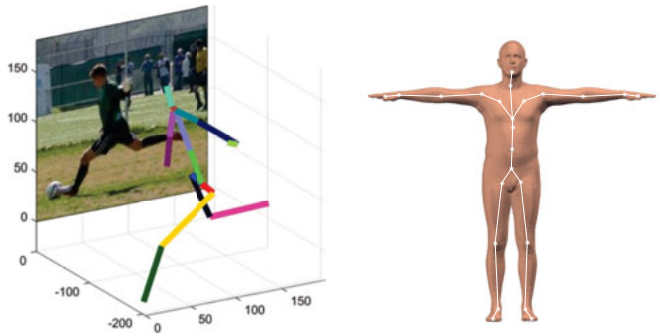
In general, human motion capture is about inferring the unknown body pose from sensor observations. In this work, we refer to this task as the *MoCap Problem*. The MoCap problem is a challenging problem for various reasons. The human musculoskeletal system of an adult is a complex compound of 206 bones connected by joints, muscles, ligaments, tendons and other connective tissue.Hence, ideally the body pose is defined in terms of the actual overall state of this system. However, modeling the static and dynamic intricacies of this complex system is almost impossible, specifically, since the musculoskeletal system is very diverse across the population due to natural variations, but also due to congenital malformations or injuries.

A further challenge arises from the fact that the skeletal structure is covered by muscle, fat, other soft tissue, and potentially clothing. Hence the actual bone states that constitute the body pose cannot be observed directly, and we have to estimate the underlying hidden states from surface observations. Of course, this could be circumvented by using x-Ray or by screwing markers into bony structures of the body. However, in this work we only consider radiation-free and non-invasive methods.

In addition to the challenges arising from modeling the human body and the hidden nature of the actual bone states, we are also faced with measurement errors in the sensor observations. All sensors have a limited precision and a method which addresses the MoCap problem should be able to alleviate these sensor errors in an adequate way. In the following we introduce common high-level strategies to approach the MoCap problem.

Human Pose Representation

In order to reduce complexity, the muscoloskeletal system is typically approximated by a simpler kinematic model. The majority of recent approaches represent the body



(a) (b)

Figure 1.4: Human pose representations: (a) Human pose is defined in terms of the 3D positions of major body joints [5]. (b) Pose is defined in terms of a skeletal model with associated joint angles. This can be used to animate a surface mesh.

pose either as a set of pre-defined *three-dimensional* (3D) key points or in terms of joint angles, see Figure 1.4.

1. The set of 3D key points usually refers to the coordinates of major joints of the human body. This representation is simple, intuitive and easy to visualize. The downside of this representation is that during inference, properties of the human skeleton, such as constant bone length and symmetry, have to be imposed explicitly. Further, not all *Degrees of Freedom* (DoF) of the human skeleton can be modeled. The state of a bone segment is determined by the 3D coordinates of the proximal and distal end. This is invariant to rotations about the bone's long axis. Hence, pronation and supination cannot be captured using this representation.

2. Joint angles refer to the set of rotational states of all major joints of the body. In particular, the joint angle(s) of a single joint describe the relative orientation between two connected bone segments. This state can be defined in terms of one, two or three numbers depending on the number of rotational DoF of the joint. This representation has the advantage that it is well suited for motion analysis and pose transfer since it is independent of anthropometric parameters such as bone length and stature. However, for visualization and in order to relate joint angles to positional sensor observations, this representation requires a body model which encodes the anthropometric properties of the body.

Usually, the total number of DoF of both pose representations lies in the range of

30-80 parameters. This is still a very high-dimensional search space. Also, not all parameter configurations lead to physically plausible poses. This accounts for joint limit violations or poses in which limbs would intersect each other.

Inference Models

Intuitively, the MoCap problem can be formalized to find the most probable pose $x \in X$ given the sensor observations $z \in Z$, which corresponds to the maximizer of the posterior probability according to

$$\arg\max_x p(x|z). \tag{1.1}$$

A probabilistic view on this problem makes sense, since both the model and measurements are associated with uncertainty. However, in this thesis we will not rely on this probabilistic view explicitly, but use it to classify approaches according to their inference model and how temporal information is processed. Almost all methods in the literature can be categorized to tackle the MoCap problem using either a generative or discriminative inference model. Generative approaches are based on a mathematical model relating sensor observations to the underlying unknown states x. Based on this model, the posterior distribution is typically modeled in terms of a fitness function f_g

$$f_g \colon X \times Z \to \mathbb{R}, \tag{1.2}$$

which provides a measure of how well the model under pose parameters x matches the sensor observations z. The most probable pose is then determined by finding the maximizer of f using some kind of optimization algorithm.

Approaches using a discriminative inference model are typically based on a direct mapping f_d from observations to pose parameters:

$$f_d \colon Z \to X. \tag{1.3}$$

Here, the idea is to learn an implicit representation of (1.1) using a large dataset of observations with corresponding poses.

Generative approaches have the advantages that domain knowledge can be modeled explicitly and inference is interpretable. Also, typically the number of model parameters is rather small and they generalize to arbitrary motions. The disadvantages are that building an adequate model is usually involved and often inference is slow.

Discriminative approaches on the other hand learn a direct mapping from inputs to the desired output. Typically, these approaches are easy to implement and inference is fast. However, incorporating domain knowledge is rather difficult and the performance heavily depends on the amount and quality of training data. A problem frequently encountered with discriminative approaches is that they do not generalize well to unseen poses or motions that have not been in the training data.

Temporal Processing

Measurements of real sensors always have an associated uncertainty. In order to better deal with these uncertainties, it makes sense to consider a larger temporal window of measurements. Also, instead of inferring the pose from measurements at a single time instance, we are commonly interested in reconstructing the time-varying pose $x(t)$ for $t = 1, ..., T$, where t refers to sample time and T is the total number of frames. In the following we will use the short-hand notation $x_{1:T}$ to refer to a time sequence.

Consequently, instead of maximizing the posterior as in (1.1), it usually makes sense to maximize the posterior of the time-varying pose $x_{1:T}$, given all measurements at corresponding time steps $z_{1:T}$:

$$\arg\max_{x_{1:T}} p(x_{1:T}|z_{1:T}). \tag{1.4}$$

Since this incorporates all available information of a complete recording sequence, we refer to this as a *global optimization formulation*. However, in order to maximize (1.4) one has to wait until all measurements are available.

An alternative formulation to this is filtering, which estimates the pose x_t at current time t given all measurements $z_{1:t}$ up to and including t. The filtering problem can be expressed in terms of the posterior according to

$$\arg\max_{x_t} p(x_t|z_{1:t}), \tag{1.5}$$

which has to be solved at each successive time step. This enables online processing. The downside is future measurements cannot be taken into account, making it less accurate than global formulations. A large variety of intermediate methods exist, which consider fixed window lengths of measurements and states, or use this information to predict future states.

Capturing and reconstructing the body pose is a very challenging task that can be solved in various ways. A perfect solution to the MoCap problem would be accurate, non-intrusive, portable, inexpensive, easy to operate and capable to capture multiple interacting people in natural environments. However, such a perfect solution does not exist yet, which is illustrated in the following section.

1.4 State of the Art

In this section, we provide a brief overview of the state of the art relevant to this work. Human motion capture has been actively researched for decades, hence an extensive survey is out of the scope of this thesis. For a more comprehensive review of the literature on this topic we refer the interested reader to survey papers [6, 7, 8].

In the following the state of the art is structured according to the sensor modality used. This comprises vision-based, inertial sensor-based and hybrid approaches. In the last section, we briefly mention magnetic and mechanical systems, which are rarely used in practice.

1.4.1 Vision-based

Vision-based approaches use cameras to capture human motions. In general, the process of mapping the 3D scene onto a *two-dimensional* (2D) sensor surface creates depth-ambiguities and occlusions, which are resolved in different ways.

Multiple Cameras

Multi-camera approaches recover the depth information by using a set of cameras to triangulate the 3D position of the human body. This requires to track and associate points on the body across multiple views. In order to simplify this process, the majority of commercial systems utilizes special markers attached to anatomical landmarks on the body. These markers are usually easy to detect in the images, and once their 3D position is triangulated, the skeletal state can be reconstructed by fitting a skeletal model to marker positions. Marker-based optical motion capture systems, such as Vicon [9] or Qualisys [10], are well established and they are generally considered the gold-standard in motion capture technology.

However, marker-based systems have drawbacks in terms of applicability. Typically 30-50 markers have to be attached to the body to capture all degrees of freedom of the skeletal structure. This generates long setup times and the subject has to be careful not to wipe off any markers during the recording. Also, subjects cannot wear regular clothing as the markers have to be attached to the skin or a tight fitting capture suit.

In the computer vision community, a great number of approaches have been published to perform motion capture without the need of markers. Instead, these approaches automatically create and associate points on the body from image features, such as silhouettes or edges. Alternatively, optical-flow or statistical models of person appearance have been used in this context. For a comprehensive overview of marker-less motion capture approaches we refer to survey papers [6, 7, 11]. Research in this area has also developed into commercial systems. The Captury [12] and Simi [13] provide systems that perform marker-less motion capture with multiple standard RGB-cameras. Marker-less approaches allow to wear regular apparel, but are usually not as accurate as marker-based approaches.

A common challenge to all camera-based motion capture systems are occlusions. In order to triangulate a point on the body, a free line of sight has to be available from at least two cameras at the same time. The articulated structure of the human body

quickly generates occlusions and points on non-facing surfaces have to be captured from other views. Consequently, this requires a large number of cameras. Despite the installation and calibration effort, this also limits the observation space to a rather small volume.

Depth-Sensors

The field of human pose estimation has experienced significant advances with the availability of the inexpensive depth sensor Kinect. A depth sensor significantly simplifies the problem since many depth ambiguities can be resolved. In the influential paper of Shotton *et al.* [14] the pose estimation problem is turned into a body part classification problem, where a pixel with known depth is associated to points on the human body. Taylor *et al.* [15] and Pons-Moll *et al.* [16] extended this approach to directly regress correspondences to a body model to improve prediction accuracy. Several other approaches have been published to tackle the problem of pose and shape estimation from depth sensors. Chen *et al.* [17] provide a survey on pose estimation using depth images. Although depth sensors are very appealing for applications such as gaming, they do not work very well outdoors, and the recording volume is limited. Furthermore, for both video and depth data, orientation ambiguities are still an issue.

Monocular Pose Estimation

Monocular pose estimation approaches aim to reconstruct the 3D body pose from a single 2D image. Typically, in a first step the pixel coordinates of major landmarks of the human body are detected. Then, the missing depth information is compensated by lifting the joint positions to 3D using learning-based methods or geometric reasoning. This direction is researched very actively [18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30]. Monocular approaches are very appealing since they only require a single camera, work outdoors and do not need any further equipment. However, they are still much less accurate than multi-camera systems or depth sensors. A main reason for the limited accuracy is that these approaches require large training data sets with a high variability in poses, appearance and environments, which are currently not available. Also, generalization to unseen poses and situations is still an open research question.

1.4.2 IMU-based

Inertial sensors are based on the principle of inertia to measure linear acceleration and rate of turn. Commonly, a set of three orthogonal accelerometers and a set of three orthogonal gyroscopes are built into an *Inertial Measurement Unit* (IMU) to sense motion in all spatial directions. By combining the sensor measurements,

it is possible to track the IMU orientation in space. Often, an IMU also contains a three-axis magnetometer measuring the local magnetic field vector to stabilize the orientation estimates. The body pose can be tracked with inertial sensors by attaching an IMU to every bone segment of interest.

In the early work of Roetenberg *et al.* [31] they use 17 IMUs and fuse the sensor measurements using a Kalman Filter. By achieving stable orientation measurements 17 IMUs completely define the pose of the subject. In the seminal work of Vlasic *et al.* [32] a custom made system is proposed. It consists of 18 sensor boards, each equipped with an IMU and acoustic distance sensors, to compensate for typical drift in the orientation estimates. For a comprehensive review of IMU-based motion capture approaches we refer to López-Nava and Munoz-Melendez [8]. Meanwhile, there also exist commercial solutions for full-body motion capture using IMUs, e.g. from Xsens [33], Shimmer [34] or Notch [35].

The advantage of IMU-based motion capture is that the sensors are body-worn and do not require any external equipment. Consequently, the recordings are not limited to a specific location or volume, and users can wear regular apparel. The main drawbacks of IMU-based motion capture are that magnetic disturbances can deteriorate the accuracy of orientation estimates, and absolute position in space cannot be tracked. In addition, in order to perform full-body motion capture, up to 17 IMUs are required, which is cumbersome to setup and quite intrusive to wear.

In this work, a method is presented that addresses the latter limitation and reconstructs the body pose from a reduced set of inertial sensors. Similar attempts have been presented in other works. Slyper and Hodgins [36] and Tautges *et al.* [37] reconstruct human pose from 5 accelerometers by retrieving pre-recorded poses with similar accelerations from a database. Acceleration data is, however, very noisy and the space of possible accelerations is huge which makes learning a very difficult task. Liu *et al.* [38] use 6 IMUs to regress the full pose using online local models to query a database. Schwarz *et al.* [39] directly regress full pose using only four IMUs with Gaussian Process regression. Both methods report very good results when the test motions are present in the database.

Although pre-recorded human motion greatly constrains the problem, methods that heavily rely on pre-recorded data are limited; in particular, capturing arbitrary activities is difficult if it is missing in the databases.

1.4.3 Hybrid Approaches

Accurate vision-based methods require a high number of cameras to resolve the ambiguities created by mapping the 3D scene onto the 2D sensor surface. Such systems are limited to a static and rather small recording volume. Inertial sensor-based systems do not suffer from these limitations since the sensors are body-worn and no external equipment is required. However, the position of a person in space

cannot be tracked with IMUs and orientation estimates are only accurate for short time periods. In general, the characteristics of camera-based and IMU-based motion capture are complementary.

Several works exploit this fact and combine both sensor modalities to hybrid approaches. Pons-Moll *et al.* [40] propose a setup with 4-8 static cameras and 5 IMUs. A local optimization scheme is applied to fit a body model to IMU orientations and person silhouettes obtained from the videos. The same setup is used in another work [41], where a particle-based optimization scheme samples from a manifold of poses which are consistent with IMU orientations. Trumble *et al.* [42] propose a CNN-based approach to fuse information from 8 camera views and IMU data to directly regress the body pose. Malleson *et al.* [43] combine IMUs with 2D poses detected in two or more static cameras. Sparse optical markers and a sparse set of IMUs are combined by Andrews *et al.* [44] to reconstruct the body pose. Since these approaches all use (multiple) static cameras, recordings are restricted to a fixed recording volume. A simple recording of movements in natural environments is not possible with such systems. Other works combine depth data with IMUs [45, 46]. However, IMUs are only used to query similar poses in a database and depth data is used to obtain the full pose.

1.4.4 Other Sensor Modalities

Mechanical systems

Mechanical motion capture systems utilize rigid or flexible body-worn goniometers to measure the relative angle between body segments [47]. In order to perform full-body motion capture, the general idea is to construct an articulated exoskeleton which is driven by the wearer. The pose is then simply defined in terms of the exoskeleton's joint states. Mechanical systems do not require external equipment, are invariant to occlusions and provide a very direct way to measure posture. However, such systems have to be carefully adapted to each user and the alignment of linkages is difficult, especially for joints with multiple degrees of freedom. Also, wearing an exoskeleton is quite intrusive and hampers the user's motions.

Magnetic systems

Magnetic motion capture systems use sensors attached to the body to measure the magnetic field generated by a transmitter source [48]. Based on the sensed magnetic field vector and strength, it is possible to compute sensor orientation and position in space. Since magnetic fields are unaffected by the human body, such systems have the advantage of being invariant to occlusions. However, ferro-magnetic materials in the capture volume can disturb the measurements, and the maximal distance

from transmitter source to sensors is limited as the magnetic field strength decreases rapidly as the distance increases.

1.5 Contributions and Outline

This thesis presents two novel methods to solve the MoCap problem. In contrast to the state of the art, the proposed methods recover the body pose from very sparse sensor sets and without making any assumptions on the motions to be captured. This significantly improves practicability and enables motion capture in everyday environments.

In order to cope with the sparsity of measurements, the proposed methods apply a global optimization formulation to maximize the consistency between a generative body model and measurements of an entire recording sequence. In particular, the modeled cost functions consist of a sum of quadratic error terms which are minimized using the Levenberg-Marquard method. Interestingly, such a setting has already been proposed in other areas such as SLAM [49] or bundle adjustment [50]. In contrast to these methods, however, in this work we do not reconstruct the static 3D point coordinates and the temporally varying rigid body motion of a single camera. Rather, the rigid body motions of all individual body segments are optimized over all frames of a recording sequence. This approach forms the foundation for the methods developed in this work, which are briefly summarized in the following.

Sparse Inertial Poser

Standard IMU-based human motion capture motion capture systems require 10-17 sensors to capture the full-body pose [33, 34, 35]. This is tedious to setup and wearing such a high number of sensors is intrusive. Existing approaches, which work with a smaller number of 5-6 inertial sensors, apply discriminative methods and reconstruct the full pose from previously recorded motion databases [36, 37, 38, 39]. However, this only works to a limited extent and does not generalize to unseen movements.

In the first part of this thesis we present the *Sparse Inertial Poser* (SIP), which is a generative method to recover the full 3D human pose from only 6 IMUs attached to wrists, lower legs, waist and head. This is a minimally intrusive solution to capture human activities with IMUs. However, orientation at the extremities and waist only provides a weak constraint on the body pose, and incorporation of acceleration data is usually affected by drift. To solve this difficult problem, we exploit a statistical body model and formulate a global optimization problem considering all sensor information of a recording sequence. In particular, we design an objective function that enforces the coherency between body model orientation and acceleration estimates against

IMU recordings. In contrast to the previous methods, the approach works for arbitrary movements and does not require pre-recorded motion databases. In several experiments we show that SIP, while simple, is very powerful and can recover all poses of a sequence as a result of a single optimization.

Video Inertial Poser

In the second part of this thesis, we present the *Video Inertial Poser* (VIP) which combines visual information from a hand-held camera with body-worn IMUs. In contrast to previous works, VIP improves IMU-based motion capture using sparse visual information, rather than extending a static camera setup with sparse IMU inputs [40, 51, 41, 42, 43]. This adds a minor additional recording effort and the system remains portable. With VIP, we extend ideas from SIP to jointly estimate the body pose of multiple people by using 6-17 IMUs attached at the body limbs and estimate their relative position and heading drift from the visual cues of the camera. Specifically, a novel graph-based association method is proposed to automatically associate IMU data with 2D image observations. This facilitates to fuse visual and inertial cues by defining an objective function and to jointly optimize for the 3D poses of the full sequence, the per-sensor heading errors, the camera pose and translation. This approach enables accurate 3D human motion capture in challenging natural scenes. We demonstrate applicability of VIP by recording *three-dimensional poses in the wild (3DPW)*: a dataset consisting of hand-held video with accurate 3D human pose and shape in natural environments.

List of Publications

During the course of this dissertation, the following peer-reviewed publications have been published at major computer vision and computer graphic conferences and journals. The first three publications deal with human motion capture and form the basis of this thesis. The fourth publication, which is not part of this thesis, deals with combining video and inertial sensors for the task of multi-people tracking and re-identification.

[51] **Timo von Marcard**, Gerard Pons-Moll, and Bodo Rosenhahn. Human Pose Estimation from Video and IMUs. *In: Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2016.

In this work, we present an approach to fuse video with sparse orientation data obtained from inertial sensors to improve and stabilize full-body human motion capture. Even though video data is a strong cue for motion analysis, tracking artifacts occur frequently due to ambiguities in the images, rapid motions, occlusions or noise. As a complementary data source, inertial sensors

allow for accurate estimation of limb orientations even under fast motions. However, accurate position information cannot be obtained in continuous operation. Therefore, we propose a hybrid tracker that combines video with a small number of inertial units to compensate for the drawbacks of each sensor type: on the one hand, we obtain drift-free and accurate position information from video data and, on the other hand, we obtain accurate limb orientations and good performance under fast motions from inertial sensors. In several experiments we demonstrate the increased performance and stability of our human motion tracker.

[52] **Timo von Marcard**, Bodo Rosenhahn, Michael J. Black, and Gerard Pons-Moll. Sparse Inertial Poser: Automatic 3D Human Pose Estimation from Sparse IMUs. *In: Computer Graphics Forum 36(2), Proceedings of the 38th Annual Conference of the European Association for Computer Graphics (Eurographics)*, 2017.[2]

We address the problem of making human motion capture in the wild more practical by using a small set of inertial sensors attached to the body. Since the problem is heavily under-constrained, previous methods either use a large number of sensors, which is intrusive, or they require additional video input. We take a different approach and constrain the problem by: (i) making use of a realistic statistical body model that includes anthropometric constraints and (ii) using a joint optimization framework to fit the model to orientation and acceleration measurements over multiple frames. The resulting tracker Sparse Inertial Poser (SIP) enables motion capture using only 6 sensors (attached to the wrists, lower legs, back and head) and works for arbitrary human motions. Experiments on the recently released TNT15 dataset show that, using the same number of sensors, SIP achieves higher accuracy than the dataset baseline without using any video data. We further demonstrate the effectiveness of SIP on newly recorded challenging motions in outdoor scenarios such as climbing or jumping over a wall.

[53] **Timo von Marcard**, Roberto Henschel, Michael J. Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering Accurate 3D Human Pose in The Wild Using IMUs and a Moving Camera. *In: European Conference on Computer Vision (ECCV)*, 2018.

In this work, we propose a method that combines a single hand-held camera and a set of Inertial Measurement Units (IMUs) attached at the body limbs to estimate accurate 3D poses in the wild. This poses many new challenges: the moving camera, heading drift, cluttered background, occlusions and many people visible in the video. We associate 2D pose detections in each image to the corresponding IMU-equipped persons by solving a novel graph based optimization problem that forces 3D to 2D coherency within a frame and

[2]Received the Günter-Enderle Award for the Best Paper at Eurographics 2017

across long range frames. Given associations, we jointly optimize the pose of a statistical body model, the camera pose and heading drift using a continuous optimization framework. We validated our method on the TotalCapture dataset, which provides video and IMU synchronized with ground-truth. We obtain an accuracy of 26mm, which makes it accurate enough to serve as a benchmark for image-based 3D pose estimation in the wild. Using our method, we recorded 3D Poses in the Wild (3DPW), a new dataset consisting of more than 51,000 frames with accurate 3D pose in challenging sequences, including walking in the city, going up-stairs, having coffee or taking the bus. We make the reconstructed 3D poses, video, IMU and 3D models available for research purposes at http://virtualhumans.mpi-inf.mpg.de/3DPW.

[54] Roberto Henschel, **Timo von Marcard**, and Bodo Rosenhahn. Simultaneous Identification and Tracking of Multiple People using Video and IMUs. *In: Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2019.

Most modern approaches for multiple people tracking rely on human appearance to exploit similarity between person detections. In this work we propose an alternative tracking method that does not depend on visual appearance and is still capable to deal with very dynamic motions and long-term occlusions. We make this feasible by: (i) incorporating additional information from body-worn inertial sensors, (ii) designing a neural network to relate person detections to orientation measurements and (iii) formulating a graph labeling problem to obtain a tracking solution that is globally consistent with the video and inertial recordings. We evaluate our approach on several challenging tracking sequences and achieve a very high IDF1 score of 91.2%. We outperform appearance-based baselines in scenarios where appearance is less informative and are on-par in situations with discriminative people appearance.

In addition to the peer-reviewed publications, two datasets have been recorded and published for research purposes:

[55] **Timo von Marcard**, Gerard Pons-Moll, and Bodo Rosenhahn. TNT15 - Multimodal Motion Capture Dataset. http://www.tnt.uni-hannover.de/project/TNT15/, 2016.

The TNT15 dataset consists of synchronized data streams from 8 RGB-cameras and 10 IMUs. In contrast to existing datasets it has been recorded in a normal office room environment and the high number of 10 IMUs can be used for new tracking approaches or improved evaluation purposes.

[56] **Timo von Marcard**, Roberto Henschel, Michael J. Black, Bodo Rosenhahn, and Gerard Pons-Moll. 3DPW - 3D Poses in the Wild Dataset. http://virtualhumans.mpi-inf.mpg.de/3DPW, 2018.

The*3D Poses in the Wild dataset* (3DPW) is the first dataset in the wild with accurate 3D poses for evaluation. While other datasets outdoors exist, they are all restricted to a small recording volume. 3DPW is the first one that includes video footage taken from a moving phone camera.

Structure of the Thesis

The thesis is structured as follows. A graphical overview is shown in Figure 1.5.

Chapter 1. Introduction: Introduces the problem statement of this work and summarizes the main contributions.

Chapter 2. Fundamentals: Provides the basic mathematical tools relevant to this work. The chapter covers rigid body motions, modeling human motion and non-linear least-squares optimization. In addition, the basic working principle of IMUs are introduced and the final section is dedicated to datasets and metrics used to evaluate the presented methods.

Chapter 3. Sparse Inertial Poser: This chapter presents a method to reconstruct the full-body pose from only a sparse set of IMUs attached to wrists, lower legs, head and waist. A global objective function is designed that enforces coherency between the body model orientation and acceleration estimates against the IMU recordings for an entire recording sequence. In several quantitative and qualitative experiments it is shown that motion capture with a reduced set of IMUs is feasible.

Chapter 4. Video Inertial Poser: This chapter presents a method for multi-person motion capture using inertial sensors and a single hand-held camera. In order to combine visual and inertial cues, 2D body poses detected in the images are automatically associated to 3D body poses obtained from IMU data using a graph labeling formulation. Then the sensor modalities are fused by minimizing an objective function to reconstruct body poses, camera pose and sensor errors. In several experiments it is shown that the approach is very accurate and enables motion capture in challenging natural environments.

Chapter 5. Conclusions: Summary of contributions, results and limitations of the proposed methods. In addition, interesting directions for future work are given.

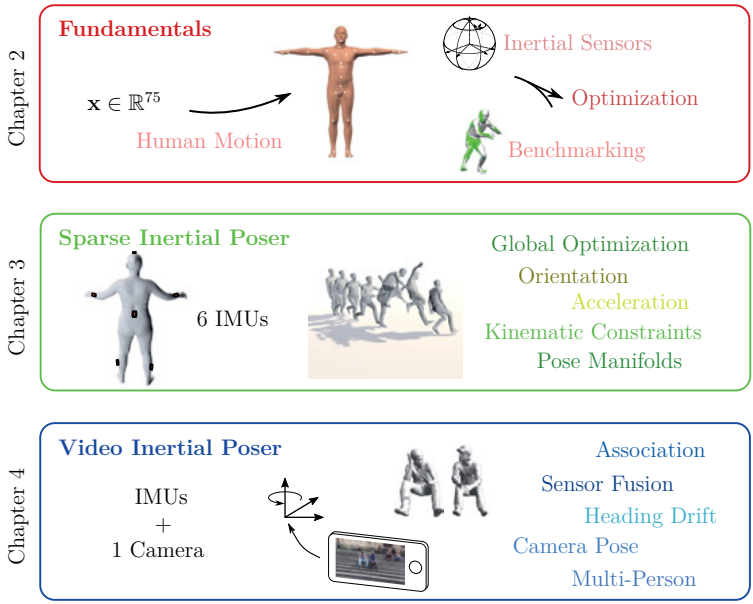


Figure 1.5: Thesis overview.

2 Fundamentals

This thesis deals with human motion in three-dimensional euclidean space. In particular, we are interested in modeling the time-dependent map $g(t) : \mathbb{R}^3 \to \mathbb{R}^3$, which describes how any point of the human body moves over time t. However, due to the complex and non-rigid structure of the human body, modeling all intricacies is almost impossible, see Section 1.3.

A common simplification is to approximate the map g by modeling the human body as a concatenation of rigid body segments. This has several advantages. First, the motion of all points belonging to a rigid body segment can be described in terms of a single mapping - a rigid body motion. Second, the articulated structure of this model enables to compute the rigid body motion of each segment in terms of joint angles, which describe the relative orientation between adjacent body segments. This parametrization is well suited to define the pose of a person in a uniform way, which is independent of anthropometric properties. Further, the rigidity assumption enables to directly relate sensory observations of sparse points on the body surface to the underlying skeletal state. The downside to this is that soft tissue motions are completely disregarded introducing a systematic source of error.

In the following, this chapter introduces the mathematical tools and other important fundamentals, which are of particular importance for this thesis. Section 2.1 deals with describing and parametrizing 3D motion of a single rigid body in space. In Section 2.2, this is extended to kinematic chains and to a full model of the human body. In order to reconstruct joint angles from sensor measurements, the methods presented in this thesis apply non-linear least-squares optimization. This topic is covered in Section 2.3. Section 2.4 deals with inertial sensors. More specifically, it covers details on measured and derived quantities and briefly describes potential sources of errors. Finally, in Section 2.5 benchmarks and accuracy metrics are introduced which are used for evaluating the proposed methods.

2.1 Rigid Body Motion

A rigid body motion $g\colon \mathbb{R}^3 \to \mathbb{R}^3$ is a mapping [57], which has the special properties that it

1. preserves distances between points:

$$||g(p) - g(q)|| = ||p - q||, \forall p, q \in \mathbb{R}^3$$

   and

2. preserves the cross product (or "orientation") between vectors:

$$g(v \times w) = g(v) \times g(w), \forall v, w \in \mathbb{R}^3.$$

The first property ensures that an object or body is not deformed under the transformation. The second property prevents internal reflections, which are physically not realizable for rigid objects. The sets of transformations satisfying these properties are denoted *Special Orthogonal Group of dimension three* (SO(3)) and *Special Euclidean Group of dimension three* (SE(3)), which are briefly introduced in the next section. For a more comprehensive introduction, we refer the interested reader to Murray *et al.* [57].

2.1.1 SO(3) and SE(3): Rigid Body Transformations

A rigid body motion corresponds to an affine transformation between two Cartesian coordinate frames, which is illustrated in Figure 2.1. We can use such a transformation to describe how points on the body move with respect to a global reference coordinate frame. Let $\mathbf{p}^b \in \mathbb{R}^3$ be a point defined a the body-fixed frame F^b and $\mathbf{p}^a \in \mathbb{R}^3$ be the same point with respect to reference frame F^a. A rigid body motion $g^a_b\colon \mathbb{R}^3 \to \mathbb{R}^3$ maps the point coordinates from F^b to F^a according to

$$\mathbf{p}^a = \mathbf{R}^a_b \mathbf{p}^b + \mathbf{t}^a, \tag{2.1}$$

where $\mathbf{R}^a_b \in SO(3)$ describes the relative rotation between F^a and F^b and $t^a \in \mathbb{R}^3$ refers to the relative translation between the origins of both coordinate systems, defined in frame F^a. The rotation is an element of $SO(3)$, which refers to the group of three-dimensional rotation matrices or special orthogonal group of dimension three:

$$\mathrm{SO}(3) := \left\{ \mathbf{R} \in \mathbb{R}^{3\times 3}\colon \mathbf{R}^T\mathbf{R} = \mathbf{I}, \det(\mathbf{R}) = +1 \right\}. \tag{2.2}$$

The special structure of a rotation matrix $\mathbf{R}$ ensures that the properties of a rigid body transformation are not violated: due to $\mathbf{R}^T\mathbf{R} = \mathbf{I}$ rotation matrices must have orthonormal columns, which guarantees that the euclidean distance between

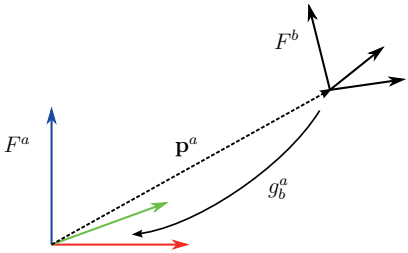


Figure 2.1: The rigid body motion g_b^a corresponds to an affine transformation of a Cartesian coordinate frame F^b to another Cartesian coordinate frame F^a.

mapped points does not change. Additionally, the constraint $\det(\mathbf{R}) = +1$ prohibits reflections, which have a negative unit determinant.

A rigid body transformation has two interpretations which are useful to keep in mind for this work. First, if the reference frame F^a represents a fixed world coordinate frame, then g_b^a describes the actual configuration of the rigid body in space. Second, if frame F^a represents the body frame, but at another time in the past, then the rigid body transformation refers to the relative net motion the body has experienced during the specified time interval. Hence, a rigid body transformation can represent both, pose *and* motion.

A rigid body transformation of the form Eq. (2.1) can be defined in terms of a *linear* matrix operation according to

$$\bar{\mathbf{p}}^a = \begin{bmatrix} \mathbf{R}_b^a & \mathbf{t}^a \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} \bar{\mathbf{p}}^b =: \mathbf{M}_b^a \bar{\mathbf{p}}^b, \tag{2.3}$$

where $\bar{\mathbf{p}} = (x, y, z, 1)^T$ denotes the homogeneous representation of a point $\mathbf{p} = (x, y, z)^T$ and $\mathbf{M} \in \mathbb{R}^{4\times 4}$ is a matrix containing the relative rotation $\mathbf{R}$ and translation $\mathbf{t}$. The set of 4×4 matrices representing rigid body motions forms a group under matrix multiplication and is denoted Special Euclidean Group SE(3):

$$SE(3) = \left\{ \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} : \mathbf{R} \in SO(3), \mathbf{t} \in \mathbb{R}^3 \right\}. \tag{2.4}$$

Using this representation, we can concatenate rigid body motions in terms of matrix multiplications

$$\mathbf{M}_a^c = \mathbf{M}_b^c \mathbf{M}_a^b \tag{2.5}$$

and invert them by

$$\mathbf{M}_b^a = (\mathbf{M}_a^b)^{-1} := \begin{bmatrix} \mathbf{R}^T & -\mathbf{R}^T\mathbf{t} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}. \tag{2.6}$$

The matrix representation of SE(3) is convenient for applying and concatenating rigid body motions as this can be done by simple matrix operations, see Eq. (2.3) and Eq. (2.5). However, matrices are not the best representation for other operations such as differentiation and optimization. This is due to the over-parametrization of the underlying group: matrices in SE(3) have 12 parameters, while a rigid body motion only has six DoF: three in rotation and three in translation.

The following section introduces the concepts of Lie groups and exponential coordinates, which provide a minimal parametrization of rigid body motions. At the same time, this representation naturally relates to infinitesimal transformations which makes it particularly easy to differentiate. There exist alternative representations for rigid body motions, such as quaternions or Euler angles which are not covered in this work. We refer the interested reader to Murray *et al.* [57] for more details.

2.1.2 Exponential Coordinates

Both SO(3) and SE(3) are Lie groups with an associated Lie algebra. Elements of a Lie group G form a smooth manifold, where the group operations multiplication and inversion are differentiable. Associated to each Lie group is a Lie algebra $\mathfrak{g}$, which corresponds to a tangent space at the identity element. The tangent space is a vector space generated by differentiating the identity element with respect to each DoF of the corresponding group. The basis elements of a k-dimensional tangent space are called generators $\{\mathbf{G}_1, \ldots, \mathbf{G}_k\}$. Every element of the tangent space $\mathbf{A} \in \mathfrak{g}$ can be represented as a linear combination of the generators $\mathbf{G}_i$ and a vector of coefficients $\mathbf{c} \in \mathbb{R}^k$ according to

$$\mathbf{A} = \sum_{i=1}^{k} c_i \mathbf{G}_i. \tag{2.7}$$

Throughout the thesis we will use the wedge-operator ($\cdot^\wedge \colon \mathbb{R}^k \to \mathfrak{g}$) to construct a Lie algebra element from a coordinate vector and the vee-operator ($\cdot^\vee \colon \mathfrak{g} \to \mathbb{R}^k$) to obtain a coordinate vector from an element of the Lie algebra. To improve readability, we will also use the hat-operator $\widehat{\cdot}$ as a replacement for $\cdot^\wedge$.

The exponential map converts any element from the Lie algebra exactly to an element of the respective Lie group. For matrix Lie groups, such as SO(3) and SE(3), the exponential map corresponds to matrix exponentiation. Conversely, the exponential map can be inverted using the matrix logarithm. Together with the Adjoint representation, the exponential map is well suited for parametrizing and differentiating rigid body motions, as described in the following.

Exponential Map on SO(3)

The exponential map on SO(3) is surjective, thus any rotation matrix $\mathbf{R} \in SO(3)$ can be constructed from a matrix exponential of the form

$$\mathbf{R} = \exp(\widehat{\omega}) = \mathbf{I} + \widehat{\omega} + \frac{1}{2!}(\widehat{\omega})^2 + \frac{1}{3!}(\widehat{\omega})^3 + \dots, \tag{2.8}$$

where $\widehat{\omega}$ is a skew-symmetric matrix

$$\widehat{\omega} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}. \tag{2.9}$$

The set of 3×3 skew-symmetric matrices corresponds to the Lie algebra of SO(3):

$$\mathfrak{so}(3) = \{\mathbf{S} \in \mathbb{R}^{3\times 3} : \mathbf{S}^T = -\mathbf{S}\}. \tag{2.10}$$

The generators of $\mathfrak{so}(3)$ are given by

$$\mathbf{G}_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}, \mathbf{G}_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}, \mathbf{G}_3 = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \tag{2.11}$$

and we can rewrite $\widehat{\omega}$ in terms of a linear combination of the generators according to

$$\widehat{\omega} = \omega_x \mathbf{G}_1 + \omega_y \mathbf{G}_2 + \omega_z \mathbf{G}_3. \tag{2.12}$$

The coordinate vector $\omega = \begin{bmatrix} \omega_x & \omega_y & \omega_z \end{bmatrix}^T$ contains the exponential coordinates of a rotation.

According to Euler's rotation theorem [57], any rotational displacement in 3D can be represented by a rotation of an angle θ about a fixed axis of unit length $\mathbf{e} \in \mathbb{R}^3$, see Figure 2.2. The pair $\{\mathbf{e}, \theta\}$ is denoted the axis-angle parameters of a rotation and is closely related to the exponential coordinates: The angle of rotation corresponds to $\theta = ||\omega||$ and the axis is given by $\mathbf{e} = \frac{\omega}{||\omega||}$.

The matrix exponential in Eq. (2.8) can be computed analytically using the Rodriguez Formula:

$$\exp(\widehat{\omega}) = I + \frac{\widehat{\omega}}{||\omega||}\sin(||\omega||) + \left(\frac{\widehat{\omega}}{||\omega||}\right)^2 (1 - \cos(||\omega||)). \tag{2.13}$$

Exponential Map on SE(3)

The exponential map on SE(3) is surjective. Any rigid motion $\mathbf{M} \in SE(3)$ can be written in exponential form

$$\mathbf{M} = \exp(\widehat{\xi}) = \mathbf{I} + \widehat{\xi} + \frac{(\widehat{\xi})^2}{2!} + \frac{(\widehat{\xi})^3}{3!} + \dots, \tag{2.14}$$

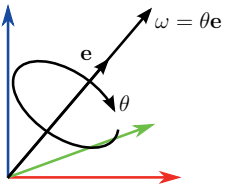


Figure 2.2: Any element in SO(3) can be represented as a rotation of an angle θ about a fixed unit axis $\mathbf{e} \in \mathbb{R}^3$. The corresponding exponential coordinates ω represent the same orientation in terms of the product of θ and $\mathbf{e}$.

where $\widehat{\xi} \in \mathbb{R}^{4\times4}$ is a matrix of the form

$$\widehat{\xi} = \begin{bmatrix} 0 & -\omega_z & \omega_y & v_x \\ \omega_z & 0 & -\omega_x & v_y \\ -\omega_y & \omega_x & 0 & v_z \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{2.15}$$

The set of matrices of the form $\widehat{\xi}$ correspond to the Lie algebra of SE(3):

$$\mathfrak{se}(3) = \{ \begin{bmatrix} \widehat{\omega} & \mathbf{v} \\ \mathbf{0}_{1\times3} & 0 \end{bmatrix} : \widehat{\omega} \in \mathfrak{so}(3), \mathbf{v} \in \mathbb{R}^3 \}. \tag{2.16}$$

The generators of SE(3), which correspond to the infinitesimal translations and rotations are defined as

$$\begin{gathered} \mathbf{G}_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \mathbf{G}_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \mathbf{G}_3 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \\ \mathbf{G}_4 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \mathbf{G}_5 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \mathbf{G}_6 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \end{gathered} \tag{2.17}$$

An element of $\widehat{\xi} \in \mathfrak{se}(3)$ is denoted a twist and the six independent parameters $\xi \in \mathbb{R}^6$ are called exponential coordinates or twist coordinates. They are composed

of the rotational parameters $\omega \in \mathbb{R}^3$ and a vector $\mathbf{v} \in \mathbb{R}^3$, which encodes the location of the axis of rotation *and* the amount of translation along that axis.

Similar to the exponential map on $SO(3)$, there exists an analytic solution to the exponential map on $SE(3)$. For a pure translational motion, i.e. if $||\omega|| = 0$, the analytic solution is simply

$$\exp(\widehat{\xi}) = \begin{bmatrix} \mathbf{I}_{3\times 3} & \mathbf{v} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}. \tag{2.18}$$

Otherwise, the analytic solution of Eq. (2.14) is given by

$$\exp(\widehat{\xi}\theta) = \begin{bmatrix} \exp(\widehat{\omega}\theta) & (\mathbf{I} - \exp(\widehat{\omega}\theta))(\omega \times \mathbf{v}) + \omega\omega^T\mathbf{v}\theta \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}, \tag{2.19}$$

where a twist is represented as $\widehat{\xi}\theta \in \mathfrak{se}(3)$, such that $||\omega|| = 1$ by appropriate scaling using $\theta \in \mathbb{R}$.

Logarithm

The exponential map defines a surjective map $g\colon \mathfrak{g} \to G$ from a Lie algebra $\mathfrak{g}$ to its corresponding Lie group G. The inverse operation $\log\colon G \to \mathfrak{g}$ is denoted log function or logarithm on G. Note that, the exponential map for SO(3) and SE(3) is not injective. A rotation about a unit rotation axis ω with angle $\theta = 2\pi k$ for any integer k produces the same rotation matrix. Hence, we restrict the log function to $\theta = [0, \pi]$, where negative rotation angles can be constructed by adapting the sign of ω.

In order to compute the twist coordinates $\xi \in \mathbb{R}^6$ from a rigid body motion $\{\mathbf{R}, \mathbf{t}\} \in SE(3)$ we have to consider two cases [57]. If $\mathbf{R} = \mathbf{I}$, the rotational parameters are zero and the twist is given by

$$\xi = \begin{bmatrix} 0 & 0 & 0 & \mathbf{t}^T \end{bmatrix}^T. \tag{2.20}$$

If $\mathbf{R} \neq \mathbf{I}$, the rotation angle θ and unit rotation axis ω are given by

$$\theta = \cos^{-1}\left(\frac{\mathrm{tr}(\mathbf{R}) - 1}{2}\right), \qquad \omega = \frac{1}{2\sin\theta}\begin{bmatrix} \mathbf{R}_{32} - \mathbf{R}_{23} \\ \mathbf{R}_{13} - \mathbf{R}_{31} \\ \mathbf{R}_{21} - \mathbf{R}_{12} \end{bmatrix}, \tag{2.21}$$

where $\mathrm{tr}(\cdot)$ refers to the trace of the matrix. From Eq. (2.19) it follows that $\mathbf{v}$ equals

$$\mathbf{v} = \mathbf{A}^{-1}\mathbf{t}, \tag{2.22}$$

with

$$\mathbf{A} = (\mathbf{I} - \exp(\theta\widehat{\omega}))\,\widehat{\omega} + \omega\omega^T\theta. \tag{2.23}$$

Adjoint

A property of Lie groups is that the tangent space has the same structure at all group elements. Specifically, we can linearly transform a tangent vector at one element to any other group element using the Adjoint transformation. Given the twist coordinates $\xi^b \in \mathbb{R}^6$ representing a rigid body motion in coordinate frame F^b, we can use the Adjoint to express the twist coordinates with respect to another coordinate frame F^a according to

$$\xi^a = Adj_{\mathbf{M}} \cdot \xi^b, \tag{2.24}$$

where $\mathbf{M} = (\mathbf{R}, \mathbf{t}) \in SE(3)$ is the configuration of F^b with respect to F^a and $Adj_{\mathbf{M}}$ is the Adjoint transformation associated to $\mathbf{M}$. The Adjoint $Adj_{\mathbf{M}} \in \mathbb{R}^{6\times 6}$ is defined as

$$Adj_{\mathbf{M}} = \begin{bmatrix} \mathbf{R} & [\mathbf{t}]_\times \mathbf{R} \\ \mathbf{0}_{3\times 3} & \mathbf{R} \end{bmatrix}. \tag{2.25}$$

The notation $[\mathbf{t}]_\times$ refers to the screw-symmetric matrix generated from $\mathbf{t}$, implementing the cross product in matrix form. Equivalently, the twist action $\widehat{\xi} \in \mathfrak{se}(3)$ is transformed according to

$$\widehat{\xi}^a = \mathbf{M} \cdot \widehat{\xi}^b \cdot \mathbf{M}^{-1}. \tag{2.26}$$

2.1.3 Differentiation

Using the exponential map formulation we can parametrize a rigid body motion $g\colon \mathbb{R}^6 \to SE(3)$ using exponential coordinates:

$$g(\xi) = \exp(\widehat{\xi}) = \exp(\sum_{i=1}^{6} \xi_i \cdot \mathbf{G}_i). \tag{2.27}$$

During optimization we are commonly interested in differentiating rigid body motions with respect to the transformation parameters ξ_i. If the differentiation is evaluated at identity, i.e. $\xi = \mathbf{0}$, this simply equates to the generators:

$$\left.\frac{\partial g(\xi)}{\partial \xi_i}\right|_{\xi=\mathbf{0}} = \mathbf{G}_i, \text{ for } i = 1, \ldots, 6. \tag{2.28}$$

Due to the Adjoint property, we can always compute the derivative at identity and transform the resulting tangent vector to any group element, where we actually want to compute the derivative. This makes differentiation particularly simple. Note that Eq. (2.28) only holds if the derivative is computed at identity.

2.2 Human Motion Modeling

In order to model human motion, we simplify and idealize the human body to an articulated structure of rigid bone segments, which are connected through joints. This defines a kinematic chain, which models the configuration space of the human skeleton. The state of this skeletal model - the body pose - is defined by a set of joint angles describing the relative orientation between each pair of bones and six additional parameters referring to global rotation and translation.

In order to accurately relate sensor observations to skeletal motions, we have to consider anthropometric properties of the person. We use the statistically learned body model SMPL to obtain person specific body models by fitting the model to body scans. Further, a central topic of this thesis is to optimize the body pose with respect to various objectives. We briefly introduce all these topics in the following.

2.2.1 Kinematic Chains

Consider two rigid segments connected by a rotational joint as shown in Figure 2.3. Attached to the left ends of each segment is a coordinate frame F^a and F^b, respectively. The rotational joint is located at the origin of F^b and has three rotational DoF. The map from frame F^b to F^a constitutes a rigid body motion

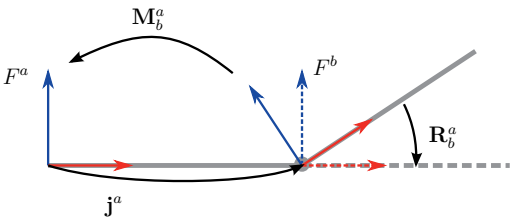


Figure 2.3: A kinematic chain with two segments connected by a rotational joint. The resulting rigid body motion $\mathbf{M}_b^a$ mapping from frame F^b to F^a depends on the joint rotation $\mathbf{R}_b^a$ and relative joint position $\mathbf{j}^a$.

$$\mathbf{M}_b^a = \begin{bmatrix} \mathbf{R}_b^a & \mathbf{j}^a \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}, \tag{2.29}$$

where $\mathbf{R}_b^a \in SO(3)$ accounts for the joint rotation, i.e. the rotational difference between F^a and F^b, and $\mathbf{j}^a \in \mathbb{R}^3$ is the joint location expressed in frame F^a. Since the joint location does not change with respect to frame F^a, we can parametrize

$\mathbf{M}_b^a$ in terms of the exponential coordinates $\omega \in \mathbb{R}^3$ of the joint rotation:

$$\mathbf{M}_b^a(\omega) = \left[\begin{array}{c|c} \exp(\widehat{\omega}) & \mathbf{j}^a \\ \hline \vec{0} & 1 \end{array}\right], \tag{2.30}$$

where we consider the joint location to be a known and constant model parameter. This setup can be extended to a chain of multiple rigid segments, illustrated in Figure 2.4. The chain comprises an ordered set (a, b, c, d) of segments, connected by an enumerated set $(1, 2, 3)$ of rotational joints. In order to map from frame F^d to

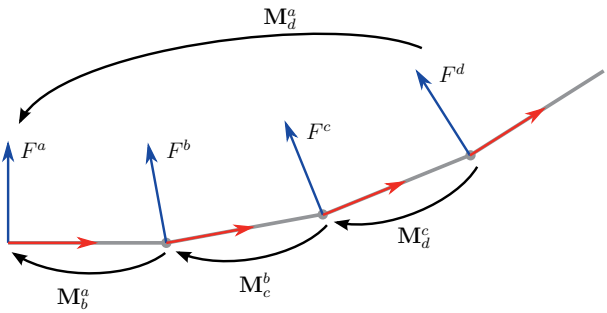


Figure 2.4: A kinematic chain with multiple rigid segments connected by rotational joints. The resulting rigid body motion from the last segment to the first segment $\mathbf{M}_d^a(\theta)$ is a composition of the rigid body motions between the individual segments. Since the segment lengths are fixed, $\mathbf{M}_d^a(\Theta)$ only depends on the angular states of all joints Θ.

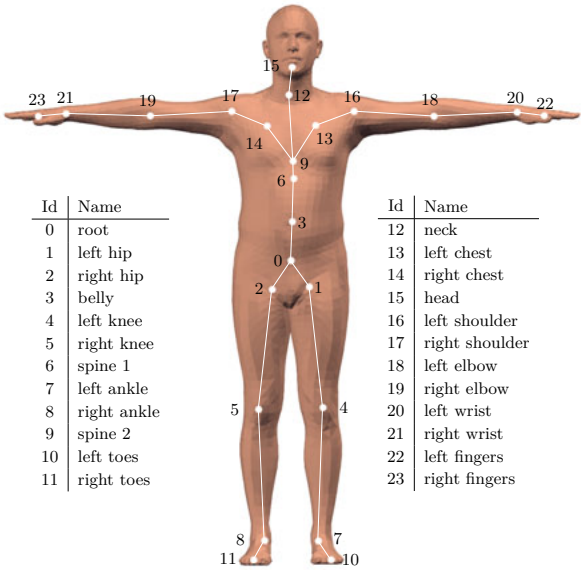
frame F^a along the chain we simply concatenate the rigid body motions

$$\mathbf{M}_d^a(\Theta) = \mathbf{M}_b^a(\omega_1) \cdot \mathbf{M}_c^b(\omega_2) \cdot \mathbf{M}_d^c(\omega_3). \tag{2.31}$$

The parameter vector $\Theta = \begin{bmatrix}\omega_1^T \, \omega_2^T \, \omega_3^T\end{bmatrix}^T \in \mathbb{R}^n$, with $n = 9$ in this case, contains the stacked exponential coordinates of each joint. In general, the map $\mathbf{M}_d^a(\Theta) : \mathbb{R}^n \to SE(3)$ from a parameter vector of pose parameters to the resulting rigid body motion is denoted the *forward kinematic map*.

2.2.2 Pose Parametrization

In order to model human articulation, we define a root joint that determines the overall orientation and position of the body. Starting from the root joint, a kinematic chain $\mathcal{C}$ is constructed which ends in the distal extremities and head, see Figure 2.5.



Id	Name
0	root
1	left hip
2	right hip
3	belly
4	left knee
5	right knee
6	spine 1
7	left ankle
8	right ankle
9	spine 2
10	left toes
11	right toes

Id	Name
12	neck
13	left chest
14	right chest
15	head
16	left shoulder
17	right shoulder
18	left elbow
19	right elbow
20	left wrist
21	right wrist
22	left fingers
23	right fingers

Figure 2.5: Skeletal structure of the SMPL body model. The kinematic chain comprises 24 joints.

Within the kinematic chain, each joint has a single parent joint, but may have multiple child joints.

In this work we model the human body with a kinematic chain $\mathcal{C}$ consisting of rigid bone segments linked by $N_j = 24$ joints. Each joint is modeled as a ball joint with three rotational DoF. At first, this seems counter-intuitive since the human skeleton comprises several joints with a limited range of motion. However, the restriction to joints with fewer DoF is not realistic. Even the knee joint, which is typically modeled as a hinge joint, permits slight internal and external rotations [58]. Secondly, the range of motion of a joint can be limited in terms of soft constraints during the pose reconstruction process, which in our observations has proven to be advantageous in contrast to reducing the DoFs.

While most joints in the body model correspond to anatomical joints, this is not true for the shoulder belt and spine. Anatomically, the shoulder belt is composed of two bones, clavicle and scapula, together with muscular connections and allows a great freedom of movement of the shoulder with respect to the rib cage [58]. This

flexibility is approximated by inserting a ball joint in the chest region, between spine and shoulder joint. For modeling the spine, five joints are considered. This is an approximation to reduce the number of model parameters. In reality, the adult human spine consists of 24 articulated vertebrae.

Finally, we define the pose of the kinematic chain $\mathcal{C}$ in terms of a pose vector $\mathbf{x} \in \mathbb{R}^d$ with $d = 3 \times 24 + 3 = 75$ parameters. This accounts to the exponential coordinates of the $N_j = 24$ ball joints and three additional parameters for global translation. To map from a body pose to rigid body motions, we define a Cartesian coordinate system to the proximal end of each bone. The rigid motion $\mathbf{M}_b^g(\mathbf{x})\colon \mathbb{R}^d \to \mathrm{SE}(3)$ of a bone F^b with respect to a global reference coordinate frame F^g depends on the states of parent joints in the kinematic chain and can be computed by the forward kinematic map:

$$\mathbf{M}_b^g(\mathbf{x}) = \left(\prod_{j \in \mathrm{Pa}_{\mathcal{C}}(b)} \left[\begin{array}{c|c} \exp(\widehat{\omega}_j) & \mathbf{j}_j \\ \hline \mathbf{0}_{1\times 3} & 1 \end{array} \right] \right) = \left(\prod_{j \in \mathrm{Pa}_{\mathcal{C}}(b)} \exp\left(\widehat{\xi}_j\right) \right), \tag{2.32}$$

where $\mathrm{Pa}_{\mathcal{C}}(b) \subseteq \{0, \cdots, N_j - 1\}$ is an ordered set of parent joints, $\omega_j \in \mathbb{R}^3$ are the exponential coordinates of the joint rotation, $\mathbf{j}_j \in \mathbb{R}^3$ is the joint location expressed in the corresponding parent frame and $\widehat{\xi}_j \in \mathfrak{se}(3)$ is the twist action of joint j. Since we assume non-variable bone lengths, the joint locations $\mathbf{j}_j$ are constant model parameters and only have to be determined to fit the model to the anthropometric properties of a person. The only exception is the root joint position $\mathbf{j}_1$ which is also a variable and accounts for global translations of the overall model.

The kinematic model described in the previous paragraphs is adapted from the SMPL body model, which is briefly introduced in the following.

2.2.3 SMPL Body Model

The Skinned Multi-person Linear (SMPL) model [59] is a body model that uses a template mesh $\mathbf{T}$ with $V = 6890$ vertices and a template skeleton, such as described in the previous section. The actual vertex positions of SMPL are adapted according to identity-dependent shape parameters and the skeleton pose.

The parameters of the model as well as a regressor from vertices to joint locations are learned from body scans. Specifically, the joint locations $\mathbf{Q} = [\mathbf{j}_1^T \dots \mathbf{j}_n^T]^T$ are predicted as a function of the surface mesh:

$$\mathbf{Q} = \mathcal{J}\mathbf{T}', \tag{2.33}$$

where $\mathcal{J}$ is a sparse regression matrix and $\mathbf{T}'$ refers to the adapted template mesh, that has been fitted to the subject shape using the identity-dependent shape parameters. This is visualized in Figure 2.6.

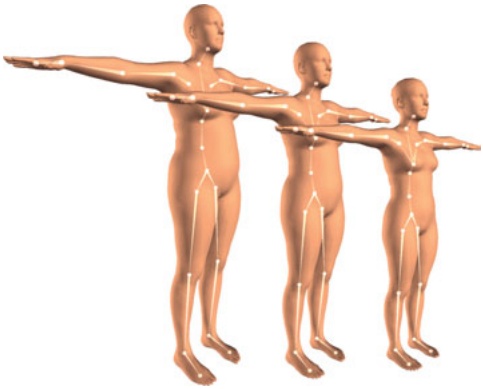


Figure 2.6: The joint positions of the skeleton in SMPL are predicted as a function of the surface.

The original intent of SMPL is to generate more realistic surface deformations, especially in regions of strong articulation. In this work, we use SMPL to primarily obtain accurate, person-specific joint positions. Hence, we skip a lot of details of SMPL and refer to the original paper [59] for more details.

2.2.4 Pose Differentiation

In this work, we will often take the derivative of rigid body motions with respect to the model parameters $\mathbf{x}$. In particular, we are interested how small parameter variations affect a rigid body motion $\mathbf{M}_b^g(\mathbf{x})$ to optimize the pose towards a certain criterion. Instead of using additive perturbations in parameter space, we define a rotational perturbation of joint j according to

$$\mathbf{R}(\omega_j \oplus \delta_j) := \exp(\widehat{\delta_j}) \cdot \mathbf{R}(\omega_j) = \exp\left(\sum_{i=1}^{3} \delta_{j,i} \mathbf{G}_i\right) \cdot \mathbf{R}(\omega_j), \tag{2.34}$$

where $\mathbf{R} \in SO(3)$ and $\omega_j, \delta_j \in \mathbb{R}^3$ are the pose and perturbation parameters associated to joint j, respectively. As we will see in the following, this enables to take derivatives of rigid body motions at identity which is particularly simple, see Section 2.1.3.

Using the notation of rotational perturbations we can rewrite the forward kinematic map (Eq. (2.32)) to

$$\mathbf{M}_b^{g'}(\mathbf{x} \oplus \delta) = \left(\prod_{j \in I(i)} \left[\begin{array}{c|c} \exp(\widehat{\delta_j}) \cdot \exp(\widehat{\omega}_j) & \mathbf{j}_j \\ \hline \mathbf{0}_{1\times 3} & 1 \end{array} \right] \right), \tag{2.35}$$

where the prime of coordinate frame g' reflects the perturbation in the resulting rigid body motion. If we only consider a scalar perturbation $\delta_{j,i}$ of joint j, we can use the Adjoint transformation to factorize Eq. (2.35) into two terms:

$$\mathbf{M}_b^{g'}(\mathbf{x} \oplus \delta_{j,i}) = \mathbf{M}_g^{g'}(\delta_{j,i}) \cdot \mathbf{M}_b^g(\mathbf{x}). \tag{2.36}$$

The right term corresponds to the original rigid body motion associated with $\mathbf{x}$ and the left term depends only on the perturbation $\delta_{j,i}$ according to

$$\mathbf{M}_{g'}^g(\delta_{j,i}) = \mathbf{M}_{p,j} \cdot \exp\left(\delta_{j,i}\mathbf{G}_i\right) \cdot (\mathbf{M}_{p,j})^{-1}, \tag{2.37}$$

where $\mathbf{M}_{p,j}$ corresponds to the motion associated with the parent joints in the chain including the translational offset $\mathbf{j}_j$ of the associated joint of j.

Instead of differentiating a rigid body motion with respect to a pose parameter in $\mathbf{x}$, we can now differentiate with respect to a perturbation $\delta_{j,i}$ and evaluate the resulting expression at identity, i.e. $\delta = \mathbf{0}$:

$$\left.\frac{\partial \mathbf{M}_b^{g'}(\mathbf{x} \oplus \delta)}{\partial \delta_{j,i}}\right|_{\delta=\mathbf{0}} = \mathbf{M}_{p,j} \cdot \mathbf{G}_i \cdot (\mathbf{M}_{p,j})^{-1} \cdot \mathbf{M}_b^g(\mathbf{x}). \tag{2.38}$$

Using Eq. (2.26), this can be simplified to

$$\left.\frac{\partial \mathbf{M}_b^{g'}(\mathbf{x} \oplus \delta)}{\partial \delta_{j,i}}\right|_{\delta=\mathbf{0}} = \widehat{\xi}'_{j,i} \cdot \mathbf{M}_b^g(\mathbf{x}). \tag{2.39}$$

where the coordinates of the twist $\widehat{\xi}'_{j,i}$ correspond to

$$\xi'_{j,i} = Adj_{\mathbf{M}_{p,j}} \cdot \mathbf{G}_i^{\vee}. \tag{2.40}$$

2.3 Non-Linear Least-Squares Optimization

The methods presented in this thesis reconstruct the body pose of a human body model from sensor measurements. This task can be formulated as finding the minimizer of an objective function $f : \mathbb{R}^n \to \mathbb{R}$, which measures the discrepancy between the model under model parameters $\mathbf{x} \in \mathbb{R}^n$ and the observed sensor information. In fact, all objective functions developed in this work have a non-linear least-squares form:

$$f(\mathbf{x}) = \frac{1}{2}\sum_{i=1}^{m}(r_i(\mathbf{x}))^2, \tag{2.41}$$

where $r_i : \mathbb{R}^n \to \mathbb{R}$ are smooth, non-linear functions representing the deviations from data point i to the predicted values provided by the model. Since a r_i describes the

residual error between model and observations, we simply refer to it as a residual or a residual function. By stacking all residuals into a single column vector

$$\mathbf{r} = \begin{bmatrix} r_1 & r_2 & \dots & r_m \end{bmatrix}^T \tag{2.42}$$

the objective function is typically written in vector form according to

$$f(\mathbf{x}) = \frac{1}{2}||\mathbf{r}(\mathbf{x})||^2 = \frac{1}{2}\mathbf{r}(\mathbf{x})^T\mathbf{r}(\mathbf{x}). \tag{2.43}$$

2.3.1 Gauss-Newton Algorithm

The Gauss-Newton algorithm [60] is an iterative method to solve non-linear least-squares problems. In each iteration, the original residual function $\mathbf{r}$ is approximated by a first-order Taylor expansion according to

$$\mathbf{r}(\mathbf{x}_i + \delta) \approx \mathbf{r}(\mathbf{x}_i) + \mathbf{J}_r\delta, \tag{2.44}$$

where $\mathbf{x}_i$ are the current parameter values at iteration i, $\delta \in \mathbb{R}^n$ is a parameter perturbation and $\mathbf{J}_r \in \mathbb{R}^{m\times n}$ is the Jacobian of $\mathbf{r}$ evaluated at $\mathbf{x}_i$:

$$\mathbf{J}_r = \left.\frac{\partial \mathbf{r}(\mathbf{x}_i + \delta)}{\partial \delta}\right|_{\delta=0}. \tag{2.45}$$

By inserting Eq. (2.44) in f we obtain an approximated objective:

$$f(\mathbf{x}_i + \delta) = \frac{1}{2}||\mathbf{r}(\mathbf{x}_i + \delta)||^2 \tag{2.46}$$

$$\approx \frac{1}{2}||\mathbf{r}(\mathbf{x}_i) + \mathbf{J}_r\delta||^2 \tag{2.47}$$

$$= \frac{1}{2}(\mathbf{r}(\mathbf{x}_i)^T\mathbf{r}(\mathbf{x}_i) + 2\delta^T\mathbf{J}_r^T\mathbf{r}(\mathbf{x}_i) + \delta^T\mathbf{J}_r^T\mathbf{J}_r\delta). \tag{2.48}$$

This approximation is quadratic in δ and setting the derivative of f to zero gives

$$\frac{df}{d\delta} = \mathbf{J}_r^T\mathbf{r}(\mathbf{x}_i) + \mathbf{J}_r^T\mathbf{J}_r\delta = 0. \tag{2.49}$$

Solving for δ leads to the optimal parameter perturbation, which minimizes the approximated objective:

$$\delta = -(\mathbf{J}_r^T\mathbf{J}_r)^{-1}\mathbf{J}_r^T\mathbf{r}(\mathbf{x}_i), \tag{2.50}$$

Finally, the parameter vector can be updated providing the starting point for the next iteration:

$$\mathbf{x}_{i+1} \leftarrow \mathbf{x}_i + \delta. \tag{2.51}$$

By repeating this process, the Gauss-Newton algorithm iterates to a local minimum of the original objective. Close to the minimum or in general if the objective is locally well approximated by the quadratic form in Eq. (2.48), it offers quadratic convergence rates. However, depending on the shape of the objective and for poorly initialized parameters the Gauss-Newton method might become unstable. Hence, convergence is not guaranteed in all cases.

2.3.2 Levenberg-Marquardt Algorithm

The Levenberg algorithm [60] is an extension to the Gauss-Newton method. Both methods have similar local convergence properties, but Levenberg proposed a trust-region strategy to avoid divergence. His algorithm incorporates an additive damping term $\lambda\mathbf{I}$ in the solution of the optimal update step of Eq. (2.50) according to

$$\delta = -\left(\mathbf{J}_r^T\mathbf{J}_r + \lambda\mathbf{I}\right)^{-1}\mathbf{J}_r^T\mathbf{r}(\mathbf{x}). \tag{2.52}$$

The non-negative parameter λ is used to control the influence of the damping term. For very small λ, the update is identical to Gauss-Newton. If λ is very large, the update step is dominated by the additive term such that

$$\delta \approx -\frac{1}{\lambda}\mathbf{J}_r^T\mathbf{r}(\mathbf{x}). \tag{2.53}$$

This corresponds to an update step of the scaled gradient descent method, which is guaranteed to converge to a local minimum if the step-size is sufficiently small. Levenberg proposed to continuously adjust λ and to accept an update step only if the new objective value is smaller than the old one. In this case, the quadratic approximation seems to be reasonable and λ is decreased. This corresponds to less damping, which can be seen as expanding the trust-region of the approximation. In contrast, if an update step does not lead to a decreasing objective value, the step is discarded and the trust-region is shrunken by adapting λ to a larger value.

The identity matrix in the damping term of the Levenberg algorithm scales each parameter dimension equally. This circumstance is improved in the Levenberg-Marquardt algorithm, which instead uses a diagonal matrix with diagonal elements of $\mathbf{J}_r^T\mathbf{J}_r$ [61]. Hence, the update step is computed according to

$$\delta = -\left(\mathbf{J}_r^T\mathbf{J}_r + \lambda\mathrm{diag}(\mathbf{J}_r^T\mathbf{J}_r)\right)^{-1}\mathbf{J}_r^T\mathbf{r}(\mathbf{x}). \tag{2.54}$$

This has the effect, that for large λ convergence is increased in directions with small gradients. Even though the damping strategy is different, both algorithms are commonly denoted Levenberg-Marquardt algorithm in the literature. Therefore, it is usually the update formula which indicates the variant of the algorithm.

The best strategy to initialize, increase and decrease λ depends on the starting point and the properties of the objective function. A rule of thumb, which generally shows good performance in terms of convergence time, is to use $\lambda_0 = 10$ for initialization and to use $\lambda_{i+1} = 10\lambda_i$ for increasing and $\lambda_{i+1} = 0.1\lambda_i$ for decreasing the damping parameter. These parameter settings have also been used for all experiments in this thesis.

2.3.3 Optimization on SO(3) and SE(3)

So far, the algorithms described for solving non-linear least-squares problems presume parameter values $\mathbf{x} \in \mathbb{R}^n$, which are elements of a vector space. This justified to

perturb the parameter values using plain vector addition, as in Eq. (2.44) and Eq. (2.51). However, this operation can not be applied if $\mathbf{x}$ represents an element of a non-euclidean Lie group G, such as SO(3) or SE(3). These groups are not closed under vector addition and it would require additional effort to back project onto the manifold. We adopt the notation from Eade [61] and apply a more elegant solution by defining the

$$\oplus : G \times \mathfrak{g} \to G \tag{2.55}$$

operator, which perturbs a group element by an increment defined in terms of the corresponding Lie algebra $\mathfrak{g}$. In this thesis we use a left-multiplicative formulation and define the parameter perturbation as

$$\mathbf{x} \oplus \widehat{\delta} = \exp(\widehat{\delta}) \cdot \mathbf{x}, \tag{2.56}$$

where $\mathbf{x} \in G$ and $\widehat{\delta} \in \mathfrak{g}$. In order to solve a non-linear least-squares problem, where $\mathbf{x} \in G$, only slight modifications to the Gauss-Newton and Levenberg-Marquardt algorithm are required. The Taylor expansion of the residuals remains almost identical, only the perturbation operation has to be replaced:

$$\mathbf{r}(\mathbf{x}_i \oplus \widehat{\delta}) \approx \mathbf{r}(\mathbf{x}_i) + \mathbf{J}_\mathbf{r} \cdot \delta, \tag{2.57}$$

where the Jacobian $\mathbf{J}_\mathbf{r}$ is again evaluated at $\delta = \mathbf{0}$:

$$\mathbf{J}_\mathbf{r} = \left. \frac{\partial \mathbf{r}(\mathbf{x} \oplus \widehat{\delta})}{\partial \delta} \right|_{\delta = \mathbf{0}}. \tag{2.58}$$

Correspondingly, the parameter update is then given by

$$x_{n+1} \leftarrow x_n \oplus \widehat{\delta}. \tag{2.59}$$

This formulation has several advantages over methods that ignore the manifold structure of the parameter space. The perturbation operator ensures to stay on the manifold during all steps of the algorithm. Also, the exponential map is always linearized around zero, which is particularly easy to compute.

In Section 2.2.4, we have already defined a slightly different perturbation operator $\oplus$ to model rotational perturbations of joints in a human body model. In contrast to Eq. (2.56), the parameter $\mathbf{x}$ and perturbation δ are exponential coordinates representing rotations. Hence, throughout this thesis we use a modified version of Eq. (2.59) to update the model parameters:

$$\omega_{n+1} \leftarrow \log(\mathbf{R}(\omega_n \oplus \delta))^\vee, \tag{2.60}$$

where $\omega \in \mathbb{R}^3$ are the exponential coordinates associated to a joint rotation $\mathbf{R} \in SO(3)$.

An alternative solution to optimize on SO(3) and SE(3) is to directly differentiate the Rodriguez formula defined in Eq. (2.13) with respect to the exponential coordinates

representing the rigid body motion. This way, the parameter space is a vector space and the Rodriguez formula provides an analytic map to the (non-euclidean) manifold structure of SO(3) and SE(3). Using this approach, updating parameters is particularly easy (plain vector addition) but computing Jacobians is more involved due to the analytic form of the Rodriguez formula.

2.4 Inertial Measurement Units

An IMU is a device containing inertial sensors. Commonly this comprises a three-axis accelerometer and a three-axis gyroscope. These sensors measure linear acceleration and rate-of-turn based on inertia, hence the naming inertial sensors. Often, IMUs also contain a three-axis magnetometer measuring the magnetic field strength and direction.

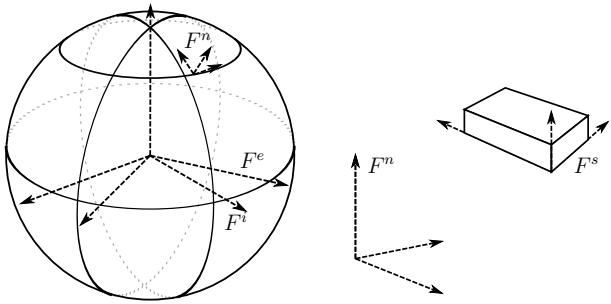
In practice, IMUs are used to track position and orientation. Originally, they were mechanical devices used to maneuver aircraft and spacecraft. With the advent of miniature IMUs based on *Micro-Electro-Mechanical Systems* (MEMS) technology, the areas of applications have multiplied: Nowadays IMUs can be found in cars, smartphones, smartwatches, fitness trackers, gamepads, VR headsets and many more. The miniaturization of IMUs has also facilitated to track the motion of individual body limbs, i.e. to perform human motion capture.

2.4.1 Coordinate Frames

Inertial sensors measure linear acceleration and angular velocity of the sensor unit with respect to a stationary reference coordinate frame. In order to describe the formation of the measured signals, we adopt the notation of Kok *et al.* [62] and define the following coordinate frames, wich also depticted in Figure 2.7:

- Sensor frame F^s is a body-fixed frame of the moving IMU.
- Navigation frame F^n is a reference frame fixed on the earth surface.
- Earth frame F^e has the origin in the center of the earth and and rotates with the earth at a rate of approximately $7.29 \cdot 10^{-5} \frac{rad}{s}$.
- Inertial frame F^i is a stationary frame, which serves as a global reference. The coordinate axes of F^i are aligned with respect to the stars and the origin coincides with the origin of the earth frame.

The gyroscope measures angular velocity ${}_{is}\omega^s$ of the sensor frame with respect to the inertial frame (indicated by the left subscript), expressed in the sensor frame (indicated by the right superscript). This quantity is a composition of the earth's



(a) (b)

Figure 2.7: IMU and reference coordinate frames. Figure adapted from Kok *et al.* [62].

angular velocity ${}_{ie}\omega^n$, transformed to the sensor frame by the rotation matrix $\mathbf{R}_n^s \in SO(3)$, and the angular velocity ${}_{ns}\omega^s$ of the sensor frame with respect to the navigation frame:

$$ {}_{is}\omega^s = \mathbf{R}_n^s \cdot {}_{ie}\omega^n + {}_{ns}\omega^s. \tag{2.61}$$

The accelerometer measures the specific force $\mathbf{a}^s$ of the sensor unit. It is a composition of the linear acceleration $\check{\mathbf{a}}^n$ due to motion and gravity $\mathbf{g}^n$:

$$ \mathbf{a}^s = \mathbf{R}_n^s \cdot (\check{\mathbf{a}}^n + \mathbf{g}^n). \tag{2.62}$$

For a stationary navigation frame, the linear acceleration is given by

$$ \check{\mathbf{a}}^n = \mathbf{a}^n + 2 \cdot {}_{ie}\omega^n \times \mathbf{v}^n + {}_{ie}\omega^n \times {}_{ie}\omega^n \times \mathbf{p}^n, \tag{2.63}$$

where $\mathbf{a}^n$ denotes the acceleration of the sensor unit with respect to the navigation frame. The terms $2 \cdot {}_{ie}\omega^n \times \mathbf{v}^n$ and ${}_{ie}\omega^n \times {}_{ie}\omega^n \times \mathbf{p}^n$ correspond to Coriolis and centrifugal acceleration, respectively. They depend on the earth rate as well as the sensor velocity $\mathbf{v}^n$ with respect to the navigation frame and sensor position $\mathbf{p}^n$.

For the task of human motion capture, the magnitude of centrifugal and Coriolis acceleration is usually smaller than $3.39 \cdot 10^{-2} m/s^2$ [62]. Also, the effect of earth rotation of approximately $7.29 \cdot 10^{-5} \frac{rad}{s}$ on the measured angular velocity is rather small compared to the rates due to motion. Hence, we disregard the effects of earth rotation in Eq. (2.61) and Eq. (2.63) by setting ${}_{ie}\omega^n = \mathbf{0}$. This facilitates to re-position the global stationary frame F^i to the ground of the recording scenery and coordinate axes aligned to gravity and a user-defined heading. The unconsidered signal portions stemming from earth rotation are simply considered measurement noise. In the following, we assume that all angular velocities refer to the rate of turn

of the sensor frame with respect to the new global stationary frame F^i and drop corresponding left subscripts to simplify notation.

2.4.2 Measurement Models

Similar to all physical sensors, inertial sensor measurements suffer from errors. These errors might stem physical properties of the measurement principle, manufacturing tolerances, temperature effects, aging, measurement noise, etc. We refer to Kok *et al.* [62] for a detailed description and discussion on these errors.

A common measurement model for inertial sensors relates the output of the sensor $\mathbf{y} \in \mathbb{R}^3$ and the quantity to measure $\tilde{\mathbf{y}} \in \mathbb{R}^3$ according to

$$\mathbf{y} = \begin{bmatrix} 1+s_x & m_{xy} & m_{xz} \\ m_{yx} & 1+s_y & m_{yz} \\ m_{zx} & m_{zy} & 1+s_z \end{bmatrix} \tilde{\mathbf{y}} + \mathbf{b} + \mathbf{v}, \tag{2.64}$$

where s are scale-factor variations and m represent axis misalignments caused by non-orthogonal sensor axes. The subscripts refer to the x-,y- and z-axis, respectively. The vector $\mathbf{b} \in \mathbb{R}^3$ denotes a sensor bias and $\mathbf{v} \in \mathbb{R}^3$ models sensor noise. Scale-factors and misalignments are usually considered deterministic and can be calibrated from the manufacturer.

The sensor bias is a slowly time-varying quantity, which is commonly modeled as a random walk process or treated constant and calibrated at the beginning of the recording. The sensor noise is usually modeled with a zero-mean Gaussian distribution $\mathbf{v} \sim \mathcal{N}(\mathbf{0}, \Sigma)$ with a diagonal covariance matrix Σ.

We briefly apply the measurement model of Eq. (2.64) to accelerometer, gyroscope and magnetometer signals in the following. We drop scale-factors and misalignments, since they are usually pre-calibrated and not important at this point.

The accelerometer output $\mathbf{y}_a \in \mathbb{R}^3$ is modeled as a composition of the specific force $\mathbf{a}^s$, sensor bias $\mathbf{b}_a^s$ and noise $\mathbf{v}_a^s$ according to

$$\mathbf{y}_a = \mathbf{R}_i^s \cdot (\mathbf{a}^i + \mathbf{g}^i) + \mathbf{b}_a^s + \mathbf{v}_a^s, \tag{2.65}$$

where we replaced $\mathbf{a}^s$ with its definition in Eq. (2.62).

The gyroscpoe output $\mathbf{y}_\omega \in \mathbb{R}^3$ is considered to be corrupted by a slowly time-varying bias $\mathbf{b}_\omega^s$ and noise $\mathbf{v}_\omega^s$:

$$\mathbf{y}_\omega = \omega^s + \mathbf{b}_\omega^s + \mathbf{v}_\omega^s. \tag{2.66}$$

For the magnetometer we are only interested in the direction $\mathbf{m}^i \in \mathbb{R}^3$ of the local magnetic field. This could be the earth magnetic field or a magnetic field due to magnetic material in the vicinity of the recording site. We model the measured magnetic field $\mathbf{y}_m$ as

$$\mathbf{y}_m = \mathbf{R}_i^s \cdot \mathbf{m}^i + \mathbf{v}_m^s, \tag{2.67}$$

where $\mathbf{R}_i^s \in SO(3)$ maps from navigation to sensor frame and $\mathbf{v}_m^s \in \mathbb{R}^3$ represents sensor noise. The underlying assumption is that the local magnetic field is constant. However, this assumption is rarely correct and the resulting uncertainty is commonly considered additional measurement noise.

2.4.3 Orientation Estimation

Inertial sensors provide motion information that can be used to track the sensor orientation $\mathbf{R}_s^i$ with respect to the stationary navigation frame F^i. If the initial orientation is known, this can be done by simply integrating gyroscope measurements $\mathbf{y}_\omega$. However, this only works in theory since the rate-of-turn measurements are corrupted by noise and bias drift, see Eq. (2.66). Straight integration of $\mathbf{y}_\omega$ also integrates measurement errors and this leads to an accumulating error in orientation estimates over time.

To compensate this, acceleration and magnetometer measurements can be incorporated since they also contain information about $\mathbf{R}_s^i$, see Eq. (2.65) and Eq. (2.67). Typically, this is realized in form of a state observer.

There exist various ways to implement such a state observer. We roughly sketch a common approach in the following. The state $\mathbf{x}_t \in \mathbb{R}^d$ at time t represents the orientation $\mathbf{R}_s^i$ in terms of a d-dimensional parametrization (usually exponential coordinates or quaternions). The subsequent state at time $t+1$ can be predicted by integrating the gyroscope measurements by a non-linear function f:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{y}_\omega) \tag{2.68}$$

Without taking into account further information this quickly becomes inaccurate due to the integration of gyroscope errors. Under the assumption that on average the linear acceleration a^i in Eq. (2.65) is zero, we can estimate the acceleration reading by

$$\tilde{\mathbf{y}}_{a,t} = h(\mathbf{x}(t)) \cdot \mathbf{g}^i, \tag{2.69}$$

where $h\colon \mathbb{R}^d \to SO(3)$ recovers the rotation matrix associated to $\mathbf{x}(t)$. Similarly, the magnetometer measurements can be estimated by

$$\tilde{\mathbf{y}}_{m,t} = h(\mathbf{x}(t)) \cdot \mathbf{m}^i. \tag{2.70}$$

In the preceding equations, we omitted sensor biases and considered them as noise. However, they can be augmented to the state vector and considered explicitly in the corresponding equations.

Equations (2.68), (2.69) and (2.70) constitute a state space model. In such a model, the state $\mathbf{x}$ is frequently corrected to reduce the deviation between estimated and measured observations. This is usually implemented in terms of an Extended Kalman Filter, which models measurement and modeling errors as Gaussians or in form of a

complementary filter that estimates orientations from sensor observations and uses a weighted average of observations and the current state. Refer to Kok *et al.* [62] for more details.

In practice, orientation accuracy for MEMS IMUs is usually $< 1^\circ$ for axes, which can be stabilized using gravity [63]. This comprises the angles with respect to the ground plane. Unfortunately, heading direction, which has to be stabilized using the magnetometer, is not as accurate. Actually, due to magnetic disturbances and local variations in the magnetic field the heading angle error is unbounded and accumulates with time.

2.4.4 Calibration

Inertial sensor measurements are taken with respect to static global inertial frame F^i. Initially, this frame is computed internally in a sensor unit: the y-axis is the negative direction of gravity measured by the accelerometer and the x-axis is the horizontal direction of the magnetic field measured by the magnetometer. Finally, the z-axis is defined by the cross-product of x- and y-axis.

As a result, each sensor defines its own reference frame $F^{i'}$. If multiple sensors are used, or if the IMU measurements are related to other sensor modalities, the reference frame needs to be unified. Since gravity determines attitude, the individual reference frames differ in a one parametric planar rotation about the vertical axis. This can be calibrated by aligning the IMUs to a common direction and correcting the IMU orientations $\mathbf{R}_s^{i'}$ by the inverse orientation $\mathbf{R}_{i'}^{i}(\gamma)$ about the Y-axis associated with respective heading angle γ_0 at $t = 0$:

$$\mathbf{R}_s^i(t) = \mathbf{R}_{i'}^i(-\gamma_0) \cdot \mathbf{R}_s^{i'}(t). \tag{2.71}$$

2.5 Benchmarking

2.5.1 Datasets

Marker-less human motion capture has been a very active research area for decades and there exist various benchmarks for evaluating video-based approaches, e.g. HumanEva [64], Human3.6M [65], CMU [66], TUM kitchen dataset [67]. Unfortunately, these datasets lack inertial data.

In the early phase of this thesis, the only exception was the MPI08 dataset [68] published by Pons-Moll *et al.* [40] and Baak *et al.* [69]. This dataset provides inertial data of five IMUs along with video data recorded in a green screen environment. Since five IMUs are insufficient for evaluating IMU-based approaches we recorded a new dataset, called TNT15, and published it in 2016 [51]. In 2018, the TotalCapture

dataset [42] was published. In addition to IMU data and multi-view video this dataset also contains ground-truth poses from a marker-based motion capture system.

In this work we use the TNT15 and TotalCapture dataset for evaluating accuracy and investigating tracking parameters. In the following, we provide details on both datasets and introduce accuracy metrics used for the experiments. We conclude this section with a brief discussion on ground-truth poses. The gold-standard to human motion capture are marker-based systems, but as we will see these systems also have an associated uncertainty.

TNT15

The TNT15 dataset consists of synchronized data streams from 8 RGB-cameras and 10 IMUs. Four actors perform five activities: walking, running on the spot, rotating arms, jumping and punching. In total, the TNT15 dataset contains more than 4:30 minutes of video and IMU data, which amounts to almost 13 thousand frames at a frame rate of 50 Hz.

For each actor, high resolution 3D laser scans are available. The dataset also contains rigged surface meshes, created by first fitting a template mesh and manually placing a skeletal model. Then, the mesh vertices are registered to the skeleton using the approach of Baran and Popović [70].

Inertial data is recorded with the wire-less MTw Development Kit by XSens [63]. In total, 10 IMUs are attached to shanks, thighs, lower arms, upper arms, sternum and waist. The actual sensor placement is depicted in Figure 2.8.

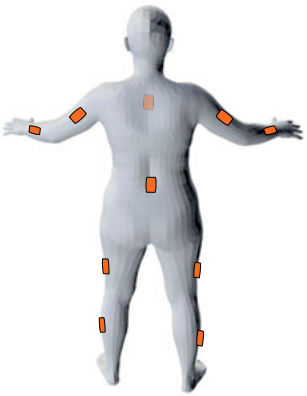


Figure 2.8: IMU placement of TNT15 dataset. IMUs are located at shanks, thighs, lower arms, upper arms, sternum and waist.

The cameras are arranged along the walls of the recording site and are calibrated to a common coordinate system. A standard pinhole camera model is applied and the calibration comprises the internal and external camera parameters as well as radial distortion coefficients [71].

To synchronize the cameras with the IMU measurements, the actors were asked to perform a foot stamp at the beginning and end of every sequence. This motion is very prominent in the camera images and IMU acceleration data and is used to manually align the data streams.

TotalCapture

The TotalCapture dataset consists of 5 subjects performing several activities such as walking, acting, range of motions and freestyle motions. Each activity is repeated three times. In order to evaluate learning-based approaches, the authors of the dataset recommend a partitioning into train and test sets. In this work, we follow this recommendation and evaluate only on the test set, which contains the activities walking 2, freestyle 3 and acting 3 for all five subjects, respectively.

The dataset is recorded using eight calibrated, static RGB-cameras and 13 IMUs attached to head, sternum, waist, upper arms, lower arms, upper legs, lower legs and feet. The exact sensor locations are illustrated in Figure 2.9.

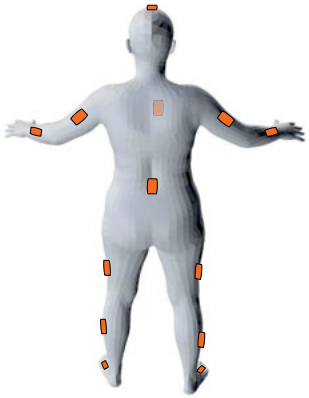


Figure 2.9: IMU placement of TotalCapture dataset. IMUs are located at feet, shanks, thighs, lower arms, upper arms, sternum, waist and head.

Ground-truth poses are obtained using a marker-baser motion capture system. All data is synchronized and captured at a frame-rate of 60Hz. The ground-truth poses

are provided in terms of joint positions, which do not contain information about pronation and supination angles, i.e. rotations about the bone's long axis. To obtain ground-truth poses with full DoFs, we fit the SMPL model to the raw ground-truth markers using a method similar to Loper *et al.* [72].

Video and IMU data are calibrated to the same coordinate system and synchronized in a similar fashion as for the TNT15 dataset.

2.5.2 Accuracy Metrics

We evaluate the accuracy of pose estimates using two error metrics: *Mean Per Joint Position Error* (MPJPE) and *Mean Per Joint Angular Error* (MPJAE). The MPJPE evaluates the accuracy of estimated joint positions with respect to the ground-truth joint positions in terms of the euclidean distance in $\mathbb{R}^3$. In contrast, the MPJAE evaluates the joint angle error in terms of the geodesic orientation distance between estimated and ground-truth joint orientations. Hence, it is a metric referring to SO(3).

The mean position error is a common metric in video-based human motion tracking benchmarks (e.g. HumanEva [64], Human3.6M [65]) and is partially complementary to the mean orientation error. Even if the joint locations are perfect, a rotation about a bone's axis does not alter the position error. This is only visible in the orientation error. On the other hand, a rather small orientation error might have a strong influence on the overall pose. For example a small orientation error in the shoulder might lead to a large positional error of the wrist due to the articulated structure. At the same time, a small orientation error about the lower arms bone axis might not be as critical. Hence, in order to evaluate tracking performance we have to consider both error metrics.

Mean Per Joint Position Error

We define the MPJPE $d_{pos}\colon \mathbb{R}^{3\times N_T\times N_J} \rightarrow \mathbb{R}$ as the average euclidean distance between the ground-truth joint positions $\mathbf{p} \in \mathbb{R}^3$ and estimated joint position $\tilde{\mathbf{p}} \in \mathbb{R}^3$ according to

$$d_{pos} = \frac{1}{N_T N_j} \sum_{t=1}^{N_T} \sum_{j=1}^{N_j} ||\mathbf{p}_j(t) - \tilde{\mathbf{p}}_j(t)||, \tag{2.72}$$

where N_j refers to the number of considered joints and N_T is the number of frames of a particular recording sequence.

Joint positions refer to an absolute position in a spatial reference coordinate system. For IMU-based motion capture this is not meaningful as global position is not trackable. A standard practice, which is also often practioned in standard benchmarks like Human3.6M [65], is to align the estimated joint positions to ground

truth joint positions before computing the MPJPE. This alignment, commonly denoted procrustes alignment, comprises a global rigid body motion and makes the MPJPE independent of global position and orientation.

Mean Per Joint Angular Error

We define the MPJAE $d_{ori}\colon \mathbb{R}^{3\times N_T\times N_j} \to \mathbb{R}$ as the average geodesic distance between the ground-truth and estimated joint rotations according to

$$d_{ori} = \frac{1}{N_T N_j} \sum_{t=1}^{N_T} \sum_{j=1}^{N_j} ||\omega_j(t)||, \tag{2.73}$$

where $\omega_j \in \mathbb{R}^3$ corresponds to the exponential coordinates of the relative orientation between ground-truth joint rotation $\mathbf{R}_j \in SO(3)$ and estimated joint rotation $\tilde{\mathbf{R}}_j \in SO(3)$ of joint j at time t:

$$\omega_j(t) = \log\left(\tilde{\mathbf{R}}_j(t) \cdot (\mathbf{R}_j(t))^{-1}\right)^{\vee}. \tag{2.74}$$

Again, N_j refers to the number of considered joints and N_T is the number of frames of a particular recording sequence. The name geodesic distance refers to the fact that $||\omega||$ with $\hat{\omega} \in \mathfrak{so}(3)$ corresponds to the relative rotation angle θ. Hence this measure represents an angular distance. An alternative metric is to compute the Frobenius norm of the rotation matrix differences: $d_{chordal} = ||\tilde{\mathbf{R}}_j(t) - \mathbf{R}_j(t)||_F$. This corresponds to a chordal distance, but is not as intuitive as the angular distance used in this work. Refer to [73] for more information about metrics on SO(3).

2.5.3 Ground-Truth Poses

Both in practical applications and in research, poses obtained with marker-based motion capture systems are usually considered gold standard. These systems infer the skeletal pose from surface-mounted markers on the human body. However, this introduces errors in two ways.

First, every physical sensor has an associated measurement uncertainty. For fully visible and moving markers, commercial marker-based systems achieve a *Root Mean Square* (RMS) marker position error lower than $2.0mm$ [74, 75]. Positional errors could be further decreased by incorporating more views. Unfortunately, the number of cameras is limited and even worse, only a subset of them has usually a free line of sight to a specific marker.

Second, the human body is a complex system composed of bones, muscles and various forms of soft tissue, see Section 1.3. Therefore, inferring the underlying skeletal state from surface-mounted markers introduces errors. In addition, accuracy

depends on proper and rigid marker positioning, which is rarely perfect in practical applications.

Even though marker-based systems suffer from the aforementioned errors, the advantages, overall accuracy and practicability justify their use as a gold standard to evaluate other motion capture approaches. However, it is important to keep the limited precision in mind, when considering this *ground-truth*.

3 Sparse Inertial Poser[1]

Figure 3.1: Illustration of the tracking performance using only 6 IMUs attached wrists, lower legs, back and head. The animation in the bottom row shows the output of the proposed method for a jumping sequence. The images in the top row are only shown for reference and are not part of the method.

This chapter presents a method to improve practicability of IMU-based human motion capture. In particular, the method enables to recover the full-body pose from only a small set of inertial sensors attached to the body. Since the problem is heavily under-constrained, previous methods rely on motion databases learning a mapping from the lower dimensional input to the full body pose. However, this makes strong assumptions about the motions to be captured and does not generalize to unseen

[1]This chapter contains previously published images, text and results [52].

movements. We take a different approach and constrain the problem by: (i) making use of a realistic statistical body model that includes anthropometric constraints and (ii) using a joint optimization framework to fit the model to orientation and acceleration measurements over multiple frames. The resulting tracker Sparse Inertial Poser (SIP) enables motion capture using only 6 sensors (attached to the wrists, lower legs, back and head) and works for arbitrary human motions, which is illustrated in Figure 3.1.

3.1 Introduction

The recording of human motion contributed substantially to the fields of biomechanics and computer animation. Typically, the recordings are made using commercial marker-based systems [9, 13], and numerous recordings of human performances are now available for research purposes [66, 76, 77]. The recording of human motion is also important for psychology and medicine, where biomechanical analysis can be used to assess physical activity and diagnose pathological conditions and monitor post-operative mobility of patients.

Unfortunately, marker-based systems are intrusive and restrict motions to controlled laboratory spaces. Therefore, activities such as skiing, biking or simple daily activities like having coffee with friends cannot be recorded with such systems. The research community in the field of computer vision has seen significant progress in the estimation of 3D human pose from images, but this typically involves multi-camera calibrated systems, which again limit applicability. Also, methods for estimating 3D human pose from single images have been proposed [78]. However, these methods are still far less accurate than motion capture systems. A common limitation of all camera-based systems is that they are prone to occlusions and during continuous operation, a free line of sight have to be available throughout the whole recording.

Systems based on IMUs do not suffer from such limitations; they can track the human pose without cameras which make them more suitable for outdoor recordings, scenarios with occlusions, baggy clothing or where tracking with a dedicated camera is simply not possible. However, inertial measurement systems such as Xsens BioMech [33] are quite intrusive, requiring 17 sensors worn on the body or attached to a suit. This is one of the reasons that large amounts of data have not been recorded yet. Hence, a less intrusive solution that can capture people through occlusions is needed.

In this chapter, we present the Sparse Inertial Poser (SIP), a method to recover the full 3D human pose from only 6 IMUs attached to the end-effectors wrists, lower legs, head and to the torso. This is a minimally intrusive solution to capture human activities with inertial sensors. Furthermore, many consumer products already have IMUs integrated, e.g. smartphones, fitness straps and smartwatches, Google glasses, and Oculus rift. A 6-sensor system could easily be worn with a hat or glasses, two

wrist bands, a belt, and shoe or ankle sensors. However, recovering human pose from only 6 IMUs is a very difficult task.

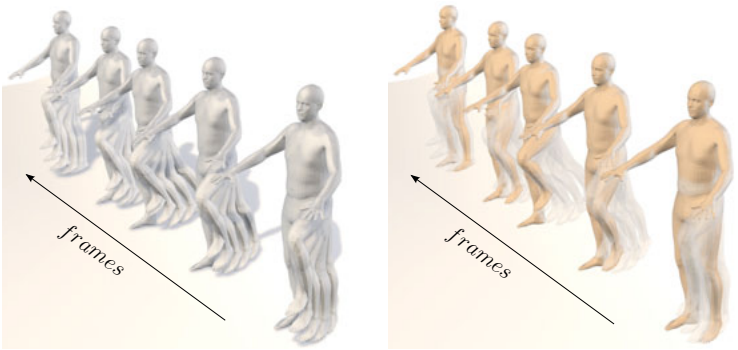
In state-of-the-art IMU-based human motion capture, IMUs are attached to each major limb segment of the human body. With theses systems, the body pose is typically reconstructed by computing the relative orientations of adjacent body segments. Hence, these solutions primarily utilize sensor orientation to capture the DoF of respective joints. In our setup, we do not have IMUs at adjacent body segments, hence we have to compensate for this missing information.

Probably the most obvious solution is to double integrate IMU accelerations to reconstruct sensor positions in space. Then, given the orientation *and* position of end-effectors and torso, the states of intermediate bone segments can be derived from the articulated structure of the human body. However, reconstructing sensor position from acceleration signals is only accurate for very short time intervals. This is due to the double integration of acceleration signals, where sensing errors (see Section 2.4.2) lead to a rapid deterioration of position estimates. This effect is usually denoted drift. In order to derive stable position information from acceleration data, an alternative is to consider other sensor modalities (e.g. *Global Positioning System* (GPS) or video) or to detect salient events where the sensor position is known (e.g. floor contact).

Figure 3.2: Sparse IMU orientations give only weak constraints on the full pose. Multiple knee and hip joint configurations fit well the IMU orientation of the lower left leg.

We take a different approach and solve this problem using a more global perspective. A key insight is that the sparse orientation measurements together with an accurate

body model constrain the set of admissible poses. In particular, orientation measurements determine the alignment of body end-effectors and waist. This information alone is not helpful, since the pose of the intermediate body parts could be arbitrary. However, not all poses are physically realizable. Hence, a body model that accurately models anatomical restrictions such as joint limits and range of motion, which we refer to as anthropometric constraints, reduces the set of theoretically possible poses to a smaller set of feasible poses. In fact, the pose parameters corresponding to feasible poses collapse to a lower-dimensional manifold. An exemplary pose manifolds is visualized in Figure 3.2, where several hip and knee joint configurations lead to identical lower leg orientations.



(a) (b)

Figure 3.3: (a) Illustration of pose manifolds for five subsequent frames. In each frame the pose manifold is obtained by fitting the IMU orientation of the lower left leg. (b) Illustration of the pose trajectory (shown in orange), that lies within the frame-wise pose manifolds and is also consistent with the acceleration data. The joint optimization over multiple frames helps to disambiguate the poses obtained with sparse orientation inputs. At the same time the frame-wise pose manifolds provide sufficient constraints to incorporate acceleration data, which would produce severe drift otherwise.

Looking at a single frame, exact pose inference is not possible. By looking at a sequence of frames however, it becomes obvious that the set of admissible poses is further constrained if we only consider smooth motions. While this is still very ambiguous, we found that it provides sufficient constraints to prevent drift if we incorporate acceleration data. Hence, the key idea of the Sparse Inertial Poser is to find a pose trajectory, which lies within the frame-wise pose manifolds and at the same time is consistent with the acceleration measurements. This is illustrated in Figure 3.3.

In summary, SIP makes the challenging problem of human pose estimation from sparse IMU data feasible by:

- Making use of a realistic body model that incorporates anthropomorphic constraints (with a skeletal rig).
- A joint optimization framework that fits the poses of a body model to the orientation and acceleration measurements over multiple frames.

Altogether SIP is the first method that is able to estimate the 3D human pose from only 6 IMUs without relying on databases of MoCap or learning methods that make strong assumptions about the recorded motion.

3.2 Model

3.2.1 Body Model

We use the SMPL body model to reconstruct the body pose from only a sparse set of IMU measurements. In this section we briefly review the basic equations to model human motion using a kinematic chain $\mathcal{C}$. Refer to Section 2.2.3 for more details.

The skeletal model of SMPL consists of rigid bone segments linked by $N_j = 24$ joints. Each joint has three rotational DoF, parametrized with exponential coordinates $\omega \in \mathbb{R}^3$. The full body pose $\mathbf{x} \in \mathbb{R}^{75}$ is defined in terms of a vector containing the stacked exponential coordinates of each joint and the three parameters for global translation.

The rigid body motion $\mathbf{M}_b^g \colon \mathbb{R}^{75} \to SE(3)$ of a bone b with respect to a global coordinate frame g depends on the pose $\mathbf{x}$ in terms of of the forward kinematic map:

$$\mathbf{M}_b^g(\mathbf{x}) = \left(\prod_{j \in \mathrm{Pa}_{\mathcal{C}}(b)} \left[\begin{array}{c|c} \exp(\omega_j^\times) & \mathbf{j} \\ \hline \vec{0} & 1 \end{array} \right] \right) = \left(\prod_{j \in \mathrm{Pa}_{\mathcal{C}}(b)} \exp\left(\xi_j^\times\right) \right), \tag{3.1}$$

where $\mathrm{Pa}_{\mathcal{C}}(b) \subseteq \{0, \cdots, N_j - 1\}$ is an ordered set of parent joints, $\omega_j \in \mathbb{R}^3$ are the exponential coordinates of the joint rotation and $\mathbf{j} \in \mathbb{R}^3$ is the corresponding joint location. Since the joint locations are individual to each person, we fit the SMPL surface mesh to each person to be tracked.

3.2.2 IMU Placement

The Sparse Inertial Poser is capable of recovering human motion from only 6 IMUs strapped to the lower legs, the lower arms, waist and head, see Figure 3.4. We

found that this sensor configuration constrains a large number of pose parameters and produces good quantitative and qualitative results. An alternative sensor configuration would be to move the lower-leg and lower-arm IMUs to the end-effectors, i.e. feet and hands. Theoretically, this would enclose the full set of major limb joint parameters of the human body. However, we found that this adds too much uncertainty along the kinematic chain structure and results in worse performance than the proposed sensor placement.

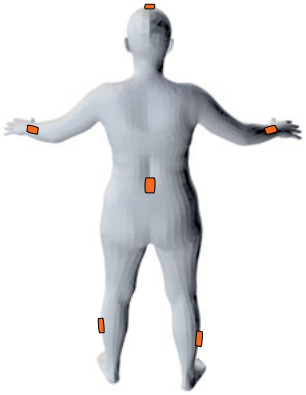


Figure 3.4: Sensor placement for the Sparse Inertial Poser. IMUs are attached to head, lower legs, wrists and back.

3.2.3 Coordinate Systems

In order to relate IMU measuremts to the body model we introduce several coordinate systems depicted in Figure 3.5. The body model is defined in the global tracking coordinate system F^g and each bone segment of the body has a local coordinate system F^b. The rigid body motion $\mathbf{M}_b^g \in SE(3)$ defines the mapping from bone to tracking coordinate system. Equivalently, $\mathbf{M}_s^i \in SE(3)$ defines the mapping from the local IMU sensor coordinate system F^s to a global inertial coordinate system F^i. Both global coordinate systems F^g and F^i are static and related by the constant mapping $\mathbf{M}_i^g \in SE(3)$. In the following we will assume $\mathbf{M}_i^g$ is known and express all IMU readings in the global tracking frame F^g using the transformation rule

$$\mathbf{M}_s^g(t) = \mathbf{M}_i^g \cdot \mathbf{M}_s^i(t). \tag{3.2}$$

Our aim is to find a pose trajectory such that the motion of a limb is consistent with IMU acceleration and orientation attached to it. Thus we need to know the

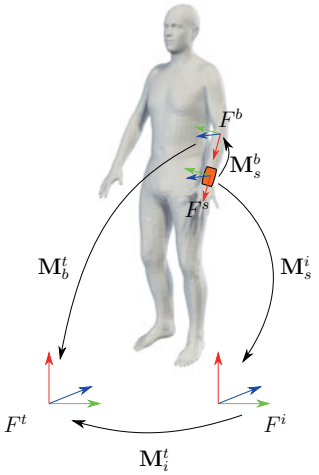


Figure 3.5: Several coordinate frames are required to relate IMU measurements to the body model: Global tracking coordinate frame F^g, inertial coordinate frame F^i, bone coordinate frame F^b and sensor coordinate frame F^s.

offset between IMU and its corresponding bone coordinate system $\mathbf{M}_s^b(t) \in SE(3)$. We assume that it is constant as the sensors are tightly attached to the limbs and compute it at the first frame of the tracking sequence according to

$$\mathbf{M}_s^b = \mathbf{M}_g^b(0) \cdot \mathbf{M}_s^g(0). \tag{3.3}$$

3.3 Method

Recovering full pose from only $N_s = 6$ IMUs, strapped at lower arms, lower legs, head and waist, is highly ambiguous. Orientation data only constrains the full pose to lie on a lower dimensional manifold. Acceleration measurements are noisy and naive double integration to obtain position leads to unbounded quadratic drift. Hence, looking at a single frame the problem is ill-posed. However, looking at the full sequence, and using anthropometric constraints from a body model, makes the problem much more constrained, see Figure 3.3. This motivates us to formulate the following multi-frame objective function:

$$\mathbf{x}_{1:T}^* = \arg\min_{\mathbf{x}_{1:T}} E_{\text{motion}}(\mathbf{x}_{1:T}, \mathbf{R}_{1:T}, \mathbf{a}_{1:T}), \tag{3.4}$$

where $\mathbf{x}_{1:T} \in \mathbb{R}^{75T}$ is a vector consisting of stacked model poses for each time step $t = 1 \dots T$. $\mathbf{R}_{1:T}$ are the sensor orientations $\mathbf{R}_t \in SO(3)$ and $\mathbf{a}_{1:T}$ are the sensor acceleration measurements respectively. We define E_{motion} as

$$\begin{aligned} E_{\text{motion}}(\mathbf{x}_{1:T}, \mathbf{R}_{1:T}, \mathbf{a}_{1:T}) = & w_{\text{ori}} \cdot E_{ori}(\mathbf{x}_{1:T}, \mathbf{R}_{1:T}) \\ & + w_{\text{acc}} \cdot E_{\text{acc}}(\mathbf{x}_{1:T}, \mathbf{a}_{1:T}) \\ & + w_{\text{anthro}} \cdot E_{\text{anthro}}(\mathbf{x}_{1:T}), \end{aligned} \tag{3.5}$$

where E_{ori}, E_{acc} and E_{anthro} are energies related to orientation, acceleration and anthropometric consistency. The weights of Eq. (3.5) balance the individual energy terms. In the following, we detail each of the objective terms.

3.3.1 The Orientation Term

The sensor orientations $\mathbf{R}_s^g(t) \in SO(3)$ are related to the bone orientations by a constant rotational offset $\mathbf{R}_s^b$. Hence, we define the estimated sensor orientation $\tilde{\mathbf{R}}_s^g(\mathbf{x}_t)$ at pose $\mathbf{x}_t$ as

$$\tilde{\mathbf{R}}_s^g(\mathbf{x}_t) = \mathbf{R}_b^g(\mathbf{x}_t) \cdot \mathbf{R}_s^b, \tag{3.6}$$

where $\mathbf{R}_b^g(\mathbf{x}_t)$ is the rotational part of the forward kinematics map defined in Eq. (3.1). The *orientation error* $\mathbf{e}_{ori} \in \mathbb{R}^3$ are the exponential coordinates of the rotational offset between estimated and measured sensor orientation:

$$\mathbf{e}_{ori}(\mathbf{x}_t) = \log\left(\tilde{\mathbf{R}}_s^g(\mathbf{x}_t) \cdot (\mathbf{R}_s^g(t))^{-1}\right)^{\vee}, \tag{3.7}$$

where the $\vee$-operator recovers the coordinates of the skew-symmetric matrix obtained from the log-operation. We define the orientation consistency E_{ori} across the sequence as

$$E_{ori} = \frac{1}{TN_s} \sum_{t=1}^{T} \sum_{n=1}^{N_s} ||\mathbf{e}_{ori,n}(t)||^2, \tag{3.8}$$

which is the sum of squared L2-norm of orientation errors over all T frames t and all N_s sensors. Actually, the squared L2-norm of $\mathbf{e}_{ori}$ corresponds to the geodesic distance between $\tilde{\mathbf{R}}_s^g(\mathbf{x}_t)$ and $\mathbf{R}_s^g(t)$, see Section 2.5.2.

3.3.2 The Acceleration Term

IMU acceleration measurements $\mathbf{a}^s \in \mathbb{R}^3$ are provided in the sensor coordinate system F^s shown in Figure 3.5. To obtain the corresponding sensor acceleration $\mathbf{a}^g$ in global tracking frame coordinates F^g we have to transform $\mathbf{a}^s$ by the current sensor orientation $\mathbf{R}_s^g(t)$ and subtract gravity $\mathbf{g}^g$:

$$\mathbf{a}^g(t) = \mathbf{R}_s^g(t) \cdot \mathbf{a}^s(t) - \mathbf{g}^g. \tag{3.9}$$

We aim to recover a sequence of poses such that the actual sensor acceleration matches the corresponding vertex acceleration of the body model. The corresponding vertex is manually selected. Since the model has the same topology across subjects this operation is done only once. The vertex acceleration $\tilde{\mathbf{a}}^g(t)$ is approximated by numerical differentiation

$$\tilde{\mathbf{a}}^g(t) = \frac{\mathbf{p}^g(t-1) - 2 \cdot \mathbf{p}^g(t) + \mathbf{p}^g(t+1)}{dt^2}, \tag{3.10}$$

where $\mathbf{p}^g(t)$ is the vertex position at time instance t and dt is the sampling time. The vertex position is related to the model pose $\mathbf{x}$ by the forward kinematic map defined in Eq. (2.32) and is given by

$$\bar{\mathbf{p}}^g(\mathbf{x}) = \mathbf{M}_b^g(\mathbf{x})\bar{\mathbf{p}}^b(0), \tag{3.11}$$

where $\bar{\mathbf{p}}$ indicates homogeneous coordinates. Hence, we define the acceleration error as the difference of estimated and measured acceleration

$$\mathbf{e}_{\text{acc}}(t) = \tilde{\mathbf{a}}^g(\mathbf{x}_{t-1}, \mathbf{x}_t, \mathbf{x}_{t+1}) - \mathbf{a}^g(t). \tag{3.12}$$

Adding up the acceleration error for all T frames and N_s sensors defines the motion acceleration consistency E_{acc}:

$$E_{\text{acc}} = \frac{1}{TN_s} \sum_{t=1}^{T} \sum_{n=1}^{N_s} ||\mathbf{e}_{\text{acc},n}(t)||^2. \tag{3.13}$$

3.3.3 The Anthropometric Term

In order to constrain the skeletal joint states to human-like poses we use a multivariate Gaussian distribution of model poses with a mean pose $\mu_{\mathbf{x}}$ and covariance matrix $\Sigma_{\mathbf{x}}$ learned from scan registrations of SMPL. While this encodes anthropometric constraints it is not motion specific as it is learned from a variety of static poses. Note that this is much less restrictive than learning based or database retrieval based approaches. We use the Mahalanobis distance that measures the likelihood of a pose $\mathbf{x}$ given the distribution $\mathcal{N}(\mu_{\mathbf{x}}, \Sigma_{\mathbf{x}})$:

$$d_{\text{mahal}}(t) = \sqrt{(\mathbf{x}_t - \mu_{\mathbf{x}})^T \, \Sigma_{\mathbf{x}}^{-1} \, (\mathbf{x}_t - \mu_{\mathbf{x}})}. \tag{3.14}$$

Additionally, we explicitly model joint limits by an error term which produces repulsive forces if a joint limit is violated. We define the joint limit error $\mathbf{e}_{\text{limit}}$ as

$$\mathbf{e}_{\text{limit}}(t) = \min(\mathbf{x}_t - \mathbf{l}_{\text{lower}}, \mathbf{0}) + \max(\mathbf{x}_t - \mathbf{l}_{\text{upper}}, \mathbf{0}) \tag{3.15}$$

where $\mathbf{l}_{lower}$ and $\mathbf{l}_{upper}$ are lower and upper joint limit parameters. Altogether, the anthropometric energy term E_{antro} is a weighted combination of terms

$$E_{\text{anthro}} = w_{\text{mahal}} \frac{1}{T} \sum_{t=1}^{T} d_{\text{mahal}}(t)^2 + w_{\text{limit}} \frac{1}{T} \sum_{t=1}^{T} ||\mathbf{e}_{\text{limit}}(t)||^2 \tag{3.16}$$

where the weighting factors w_{mahal} and w_{limit} balance the influence of the pose prior term and the joint limits term.

3.3.4 Energy Minimization

E_{motion} is a highly non-linear function and generally difficult to optimize. However, the exponential map formulation enables to analytically compute gradients and since E_{motion} is composed of a sum of squared residual terms we can use the Levenberg-Marquardt algorithm introduced in Section 2.3.2.

In order to compute an update-step for the Levenberg-Marquardt algorithm, we have to linearize the residual terms $\mathbf{e} \in \mathbb{R}^d$ around the current pose estimate $\mathbf{x} \in \mathbb{R}^{75}$:

$$\mathbf{e}(\mathbf{x} \oplus \delta) \approx \mathbf{e}(\mathbf{x}) + \mathbf{J}(\mathbf{x}) \cdot \delta, \tag{3.17}$$

where $\mathbf{J}(\mathbf{x}) : \mathbb{R}^{75} \to \mathbb{R}^d$ is the Jacobian matrix mapping a pose increment $\delta \in \mathbb{R}^{75}$ to an increment of the residual. The operator $\oplus$ refers to a parameter perturbation with respect to the manifold structure of rigid body motions, see Section 2.2.4. In the following we show how to linearize the respective residual terms associated to orientation, acceleration and anthropometric consistency.

The orientation residual defined in Eq. (3.7) can be rewritten in terms of an incremental change of the pose parameters δ according to

$$\mathbf{e}_{ori}(\mathbf{x} \oplus \delta) = \log\left(\hat{\mathbf{R}}_s^g(\mathbf{x} \oplus \delta) \cdot (\mathbf{R}_s^g)^{-1}\right)^\vee, \tag{3.18}$$

where $\hat{\mathbf{R}}_s^g(\mathbf{x} \oplus \delta)$ is the rotational part of the forward kinematic map defined in Eq. (2.35). For a single pose parameter i that has an effect on $\hat{\mathbf{R}}_s^g$, we can use the Adjoint to shift the rotation associated to the perturbation δ_i all the way to the left:

$$\mathbf{e}_{ori}(\mathbf{x} \oplus \delta_i) = \log\left(\exp(\widehat{\omega_i}) \cdot \tilde{\mathbf{R}}_s^g(\mathbf{x}) \cdot (\mathbf{R}_s^g)^{-1}\right)^\vee. \tag{3.19}$$

where

$$\omega_i = Adj_{\mathbf{R}_i} \cdot (\mathbf{G}_i)^\vee \cdot \delta_i. \tag{3.20}$$

Here, the rotation $\mathbf{R}_i$ refers to the accumulated joint rotations of parental joints of i and $\mathbf{G}_i$ is the corresponding Generator matrix. Instead of taking the derivative with respect to δ_i explicitly, we use the first-order approximation of the logarithm [79]:

$$\log\left(\exp(\widehat{\omega}_a)\exp(\widehat{\omega}_b)\right) \approx \widehat{\omega}_a + \widehat{\omega}_b, \tag{3.21}$$

where $\widehat{\omega}_a, \widehat{\omega}_b \in \mathfrak{so}(3)$ to obtain

$$\mathbf{e}_{ori}(\mathbf{x} \oplus \delta_i) \approx \mathbf{e}_{ori}(\mathbf{x}) + Adj_{\mathbf{R}_i}(\mathbf{G}_i)^\vee \cdot \delta_i = \mathbf{e}_{ori}(\mathbf{x}) + \mathbf{J}_{\text{ori},i}(\mathbf{x}) \cdot \delta_i. \tag{3.22}$$

From this we can already read off the Jacobian $\mathbf{J}_{\text{ori},i}(\mathbf{x}) \in \mathbb{R}^3$ for δ_i which is given by

$$\mathbf{J}_{\text{ori},i}(\mathbf{x}) := Adj_{\mathbf{R}_i} \cdot (\mathbf{G}_i)^\vee. \tag{3.23}$$

The full Jacobian $\mathbf{J}_{\text{ori}}(\mathbf{x}) \in \mathbb{R}^{3\times 75}$ of Eq. (3.18) is obtained by simply appending the $\mathbf{J}_{\text{ori},i}(\mathbf{x})$ for all perturbation parameters i. If the orientation residual does not

depend on a particular perturbation parameter, then the corresponding row in the Jacobian is simply a zero vector. Note that the approximation in Eq. (3.21) is only reasonably accurate if one exponent is close to identity.

In order to linearize the acceleration residual of Eq. (3.12), we express the estimated sensor position (Eq. (3.11)) at a single time instance in terms of an incremental change in the pose vector δ according to

$$\bar{\mathbf{p}}(\mathbf{x} \oplus \delta) = \mathbf{M}_b^g(\mathbf{x} \oplus \delta) \cdot \bar{\mathbf{p}}^b(\mathbf{0}). \tag{3.24}$$

Using Eq. (2.39), the derivative of this expression with respect to a single pose parameter δ_i equates to

$$\left.\frac{\bar{\mathbf{p}}(\mathbf{x} \oplus \delta)}{\partial \delta_i}\right|_{\delta=\mathbf{0}} = \widehat{\xi}_i \cdot \mathbf{M}_b^g(\mathbf{x}) \cdot \bar{\mathbf{p}}(\mathbf{0}) =: \mathbf{J}_{p,i}(\mathbf{x}), \tag{3.25}$$

where

$$\xi_i = Adj_{\mathbf{M}_i} \cdot \mathbf{G}_i^{\vee}, \tag{3.26}$$

and $\mathbf{M}_i$ corresponds to the motion associated with the parent joints of i, see Section 2.2.4. Similar to the orientation residual, we obtain the full Jacobian $\mathbf{J}_p(\mathbf{x}) \in \mathbb{R}^{3\times 75}$ by simply appending the $\mathbf{J}_{p,i}(\mathbf{x})$ for all perturbation parameters i. If the point position does not depend on a perturbation parameter i, then $\mathbf{J}_{p,i}(\mathbf{x})$ is simply a zero vector.

By combining the position estimates of three successive time steps we get the linearized acceleration error according to

$$\mathbf{e}_{acc}(t, \delta) \approx \mathbf{e}_{acc}(t) + \begin{bmatrix} \mathbf{J}_p(\mathbf{x}_{t-1}) & -2\mathbf{J}_p(\mathbf{x}_t) & \mathbf{J}_p(\mathbf{x}_{t+1}) \end{bmatrix} \begin{bmatrix} \delta_{t-1} \\ \delta_t \\ \delta_{t+1} \end{bmatrix}. \tag{3.27}$$

The residual terms related to anthropomorphic consistency defined in Eq. (3.14) and Eq. (3.15) are already linear in the pose $\mathbf{x}$. For the Mahalanobis prior we compute the Cholesky factorization of the inverse covariance matrix

$$\Sigma_{\mathbf{x}}^{-1} := \mathbf{L}^T\mathbf{L}, \tag{3.28}$$

and rewrite the squared Mahalanobis distance as

$$d_{\text{mahal}}^2 = (\mathbf{x} - \mu_{\mathbf{x}})^T \cdot \mathbf{L}^T \cdot \mathbf{L} \cdot (\mathbf{x} - \mu_{\mathbf{x}}) = \mathbf{e}_{\text{mahal}}^T \cdot \mathbf{e}_{\text{mahal}}. \tag{3.29}$$

Then it becomes obvious that $\mathbf{e}_{\text{mahal}} : \mathbf{x} \mapsto L(\mathbf{x} - \mu_{\mathbf{x}})$ is a linear mapping with $\mathbf{J}_{\text{mahal}} := \mathbf{L}$.

In order to compute a descent update step to minimize E_{motion}, we can now simply stack the linearized residual terms for all frames. For orientation and anthropometric

terms this leads to sparse equations with the following block-diagonal structure

$$\begin{bmatrix} \ddots & & & & & \\ & \mathbf{J}_{t-1} & & & & \\ & & \mathbf{J}_t & & & \\ & & & \mathbf{J}_{t+1} & & \\ & & & & \ddots & \end{bmatrix} \begin{bmatrix} \vdots \\ \delta_{t-1} \\ \delta_t \\ \delta_{t+1} \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ \mathbf{e}(t-1) \\ \mathbf{e}(t) \\ \mathbf{e}(t+1) \\ \vdots \end{bmatrix}, \tag{3.30}$$

where $\mathbf{J}_t$ denotes the respective Jacobian of the residual term $\mathbf{e}(t)$ at time step t. Similarly, the linearized residual terms of the acceleration residuals can be combined to obtain

$$\begin{bmatrix} \ddots & \ddots & & & & \\ \ddots & -2\mathbf{J}_{t-1} & \mathbf{J}_t & & & \\ & \mathbf{J}_{t-1} & -2\mathbf{J}_t & \mathbf{J}_{t+1} & & \\ & & \mathbf{J}_t & -2\mathbf{J}_{t+1} & \ddots & \\ & & & \ddots & \ddots & \end{bmatrix} \begin{bmatrix} \vdots \\ \delta_{t-1} \\ \delta_t \\ \delta_{t+1} \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ \mathbf{e}_{\text{acc}}(t-1) \\ \mathbf{e}_{\text{acc}}(t) \\ \mathbf{e}_{\text{acc}}(t+1) \\ \vdots \end{bmatrix}. \tag{3.31}$$

By stacking the respective linearized multi-frame residual terms, we can now simply solve for the parameter updates and iterate until convergence. Iteration results for a jumping jack sequence are illustrated in Figure 3.6.

3.4 Evaluation

We evaluate SIP on two publicly available benchmark datasets and present tracking results on challenging outdoor recordings. This section is structured as follows. First, we present details on the general tracking procedure and computation times in Section 3.4.1. In Section 3.4.2 we evaluate tracking performance of SIP on the TNT15 dataset and investigate the influence of using a learned body model in contrast to a manually rigged body model. Then, in Section 3.4.3 we evaluate our tracking approach on the TotalCapture dataset and assess tracking accuracy with respect to a marker-based reference motion capture system. Finally, in Section 3.4.4 we show qualitative results on additional recordings, which were captured using a sparse set of IMUs and a hand-held smartphone camera for visualization purposes.

3.4.1 Tracker Setup

In order to reconstruct the full-body motion with our proposed SIP, we require a SMPL body model of the actor, the initial pose at the beginning of the sequence, and sensor locations on the body. Initial pose and sensor locations are required to

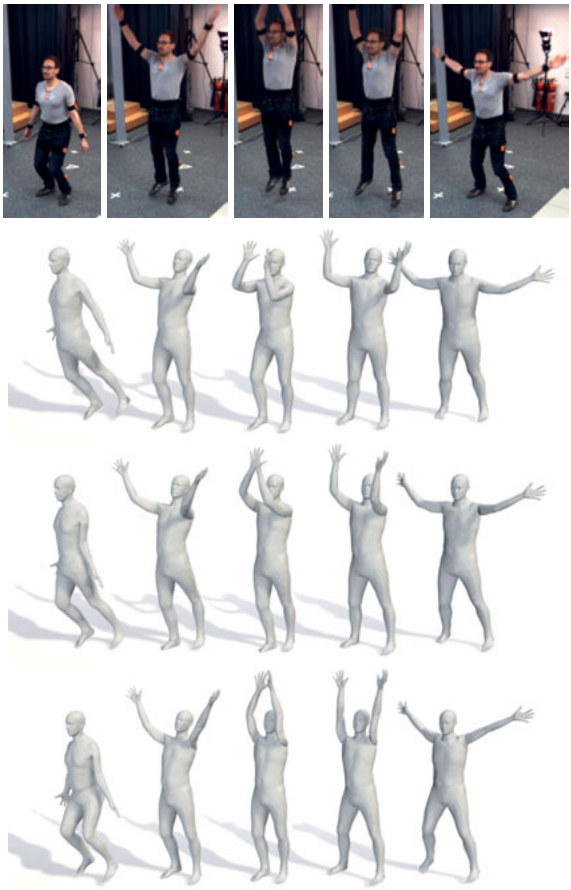


Figure 3.6: Three iterations of optimizing E_{motion} for a jumping jack sequence. First row: images of the scene, second row: pose initialization obtained by minimizing orientation and anthropometric consistency, third row: intermediate iteration, forth row: result of SIP, i.e. final pose estimates after convergence.

determine the sensor to bone offsets $\mathbf{M}_s^b$, see Section 3.2.3. Since IMUs are attached to different locations on the body, we manually selected the SMPL vertices once, and use them as sensor locations for all actors and experiments. Initial poses for the quantitative analysis were provided along the corresponding datasets. For the

outdoor recordings we simply asked the actor to pose upright with straight arms and legs at the beginning of each sequence. We obtained SMPL body models by fitting the SMPL template to laser scans or to marker locations in the case of the TotalCapture dataset, respectively. We also evaluated tracking accuracy using approximate body models estimated with the method of *bodies from words* [80]. In this case shape is estimated from only height, weight and 15 user ratings of the actor body shape.

The general tracking procedure works as follows. Starting with the initial pose we optimize the body pose for every frame sequentially using the orientation and anthropometric terms. We call this method Sparse Orientation Poser (SOP) and we use it as a baseline later. The resultant pose trajectory from SOP serves as initialization for optimizing the full cost function defined in Eq. (3.5). As can be seen in Figure 3.6, optimizing orientation and anthropometric consistency terms already recovers the pose reasonably well. This step is important, since Eq. (3.5) is highly non-linear and we apply a local, gradient-based optimization approach. After initialization, we use a standard Levenberg-Marquardt algorithm to optimize the full cost cost function and iterate until convergence.

For all experiments, we use weighting parameters $w_{ori} = 1$, $w_{acc} = 0.05$, $w_{anthro} = 1$, $w_{mahal} = 0.003$, and $w_{limits} = 0.1$, which have been determined empirically. The overall processing time for a 1000 frame sequence and 20 cost function evaluations on a quad-core Intel Core i7 3.5 GHz CPU is 7.5 minutes using single-core, non-optimized MATLAB code. For each iteration the majority of time is spent on updating the body model (14.4s) and setting up the Jacobians (3.3s), while solving the sparse equations for a Levenberg-Marquardt update step takes approximately 1.5s. Parallelization of model updates and Jacobian entries on the GPU would drastically reduce computation time.

3.4.2 Evaluation on TNT15

The TNT15 data set [51] contains recordings of four subjects performing various activities. The dataset provides inertial sensor data of 10 IMUs attached to lower legs, thighs, lower arms, upper arms, waist and chest. Refer to Section 2.5.1 for a more detailed description of the dataset.

We use the TNT15 dataset to evaluate tracking performance with respect to the ground-truth pose obtained by using all 10 IMUs. A focus is set on investigating the influence of the body model. In contrast to TotalCapture, the TNT15 dataset provides rigged body models *and* laser scans of the actors. This facilitates to fit SMPL to laser scans and use the manually rigged body models as a reference.

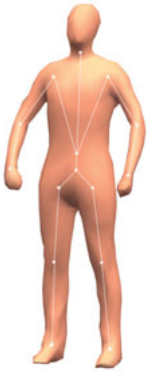


Figure 3.7: A hand-rigged body model provided along the TNT15 dataset. In contrast to SMPL, the joints are placed manually and kinematic constraints are imposed by using hinge and saddle joints.

Baseline Trackers

We compare our tracking results to two baseline methods:

- *Sparse Orientation Poser* (SOP): Minimizes orientation and anthropomorphic consistency terms but disregards acceleration.
- *SIP using an alternative body model* (SIP-M): Identical to SIP, but uses a manually rigged body model.

The estimated pose trajectory obtained by SOP is used as the initialization of our proposed SIP. The second baseline, the SIP-M, uses a body model provided along the TNT15 data set shown in Figure 3.7. It is a body model with manually placed joints and fewer pose parameters. Anatomical constraints are imposed by using hinge joints, e.g. for the knee. In total, the body model has 31 pose parameters and the manual rigging procedure is representative for models that have been used for tracking so far [81, 41, 51, 82]. In contrast, the SMPL model of SIP uses a statistical model to estimate joint positions. Every joint has 3 DoFs and anatomical constraints are imposed with the covariance of joint parameters. By comparing SIP and SIP-M we want to asses the significance of using a statistically learned body model in contrast to a typical hand-rigged one.

We also experimented with a single-frame acceleration tracker which combines the SOP approach with acceleration data using a Kalman filter. This is similar to approaches of Vlasic *et al.* [32] and Roetenberg *et al.* [31] but considers inertial data of only 6 sensors. Unfortunately, only 6 IMUs do not provide sufficient constraints

on the poses to prevent drift caused by acceleration. In all cases, the tracker got unstable and failed after a few frames.

Metrics

We use the MPJPE and MPJAE defined in Section 2.5.2 to evaluate SIP against the baseline trackers. For computing the MPJPE, we use the ground-truth positions of $N_m = 13$ virtual markers on the body model and compare them to the marker positions obtained with the estimated poses. The virtual marker positions comprise the SMPL-model joint locations of hips, knees, ankles, shoulders, elbows, wrists and neck. Unfortunately the TNT15 dataset does not provide ground-truth poses obtained with a marker-based reference motion capture system. Instead, we use the full set of 10 IMUs and generate ground-truth by adjusting the SMPL body pose to match the measured IMU orientations. Since we cannot obtain stable ground-truth global translation from IMUs alone, we set it to zero for calculating MPJPE.

For computing the MPJAE we use a slightly different approach compared to the definition in Section 2.5.2. We split the 10 IMUs into tracking and validation sets. IMUs attached to lower legs, lower arms, waist and chest are used for tracking and the other IMUs serve as validation sensors. Hence, in this section we compute the MPJAE in terms of the geodesic distance between the measured validation sensor orienations and the corresponding virtual sensor orientations obtained with the estimated poses. This gives a clear separation of signals used for tracking and validation. This separation is not entirely given for the MPJPE, as ground-truth positions are obtained using all IMUs.

Quantitative Results

Figure 3.8 shows the tracking errors for a jumping jack sequence of the TNT15 data set. This sequence contains extended arm and leg motions, also visible in Figure 3.6, as well as two foot stamps around frames 25 and 500. The SOP fails to accurately reconstruct these motions as orientation measurements of 6 IMUs are too ambiguous. This is easily illustrated for the case of a foot stamp, which can be seen in the second column of Figure 3.12. During this motion the lower leg is tilted, but without acceleration data it is impossible to infer whether the thigh was lifted at the same time. The SIP-M can resolve this ambiguity but the limited body model is not sufficiently expressive to accurately reconstruct the jumping jacks and skiing exercises. In contrast our proposed SIP shows low orientation and position errors for the whole sequence and clearly outperforms both baseline trackers. The tracking result of the jumping jack sequence is exemplary for the overall tracking performance on the TNT15 data set summarized in Table 3.1. In comparison to SOP, which uses only orientation data, SIP reduces the mean orientation error on the TNT15 data set from 19.6° to 13.3° and the mean position error decreases from

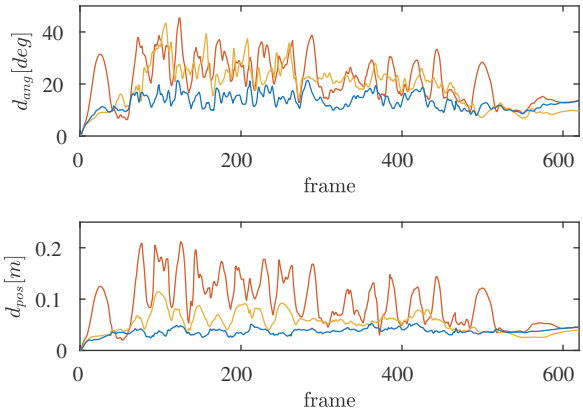


Figure 3.8: Illustration of the error metrics for a jumping jack sequence of the TNT15 data set. SIP (blue) clearly outperforms both baseline trackers SOP (red) and SIP-M (yellow). The graphs show the angular error and positional error averaged for each frame, respectively.

Table 3.1: Error metrics and standard deviations for SOP, SIP-M and SIP evaluated on TNT15.

Approach	MPJAE[°]	MPJPE[mm]
SOP	19.6 ± 17.4	72.2 ± 89.0
SIP-M	18.2 ± 15.8	55.9 ± 55.4
SIP	13.3 ± 10.1	39.1 ± 40.5

72.2 mm to 39.1 mm. In contrast, the manually rigged body model used in SIP-M achieves a MPJAE of 18.2° and a MPJPE of 55.9 mm.

It is remarkable, that SIP-M and SIP achieve a mean orientation error of 18.2° and 13.3°, respectively. In our earlier work [51] we achieved an average orientation error of 15.7°, using 5 IMUs and 8 cameras by minimizing single-frame orientation and silhouette consistency terms. SIP-M uses the same body model and is just slightly worse. Using the SMPL body model in SIP results in an even smaller orientation error. Thus, without relying on visual cues of 8 cameras we achieve competitive orientation errors by simply taking IMU accelerations into account and optimizing over all frames simultaneously.

In Figure 3.9 and Figure 3.10 the error metrics are plotted for all actors, separated by activities. Throughout all activities, SIP achieves a substantial improvement in

accuracy with respect to the baseline SOP. Since SIP does not make any assumptions about the motions to be reconstructed, this supports that SIP can generalize to arbitrary motions. For SIP-M the results are not as consistent when compared to SOP. For the majority of activities it outperforms SOP in both error metrics. However, the MPJPE of SIP-M is worse for walking and rotation arms and the MPJAE is worse for the walking sequence. Hence, the reduced DoF of the skeletal model and the manually placed joint positions prevented to successfully disambiguate the motions.

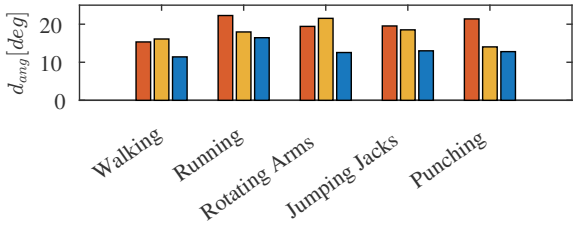


Figure 3.9: MPJAE on the TNT15 dataset. Comparison of SOP(red), SIP-M(yellow) against the proposed SIP (blue).

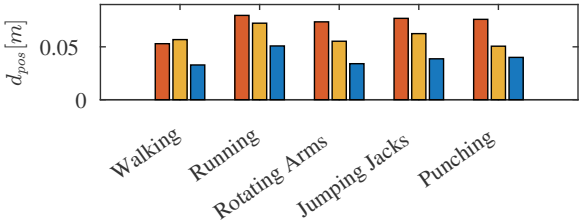


Figure 3.10: MPJPE on the TNT15 dataset. Comparison of SOP(red), SIP-M(yellow) against the proposed SIP (blue).

In Table 3.2 we summarize the tracking performance for SIP-BW, SIP-110 and SIP-120. SIP-BW is identical to SIP but uses a SMPL model estimated with the *bodies from words* approach. The tracking error difference is insignificant, which further proves applicability of SIP. Thus, we do not need the accuracy of a laser scan, making the proposed solution very easy to use. SIP-110 and SIP-120 use a scaled version of the SIP body model, where body size was increased by 10% and 20% respectively. Again, the tracking error remains comparably small demonstrating that SIP is very robust to moderate variations in body shape.

Table 3.2: Tracking errors of SIP-BW, SIP-110, SIP-120 and SIP evaluated on TNT15.

Approach	MPJAE[°]	MPJPE[mm]
SIP-BW	13.5	41.5
SIP-110	13.7	45.7
SIP-120	14.3	55.8
SIP	13.3	39.1

(a) (b) (c) (d)

Figure 3.11: Influence of the anthropometric, orientation and acceleration consistency terms. (a) image of a climbing scene (b) using only orientation without anthropometric consistency term, (c) using orientation with anthropometric consistency term , (d) our proposed SIP using anthropometric, orientation and acceleration consistency terms. The acceleration information clearly helps to disambiguate the leg poses.

Quantitative results indicate that accurate full-body motion tracking with sparse IMU data becomes feasible by incorporating acceleration data. The influence of the anthropometric, orientation and acceleration terms are also illustrated in Figure 3.11. We have also shown that for our tracking approach, the statistically learned body model SMPL leads to more accurate tracking results than using a representative manually rigged body model. Further, the SMPL model can be even created using only linguistic ratings, which obviates the need for a laser scan of the person. In Figure 3.12 we show several example frames of the tracking results obtained on the TNT15 data set.

3.4.3 Evaluation on TotalCapture

The TotalCapture dataset contains motion recordings of five subjects captured with a marker-based motion capture system and inertial sensor data of 13 IMUs attached to feet, lower legs, thighs, lower arms, upper arms, waist, sternum and head. Refer to Section 2.5.1 for a more detailed description of the dataset.

We evaluate SIP on TotalCapture to assess tracking performance of SIP with respect

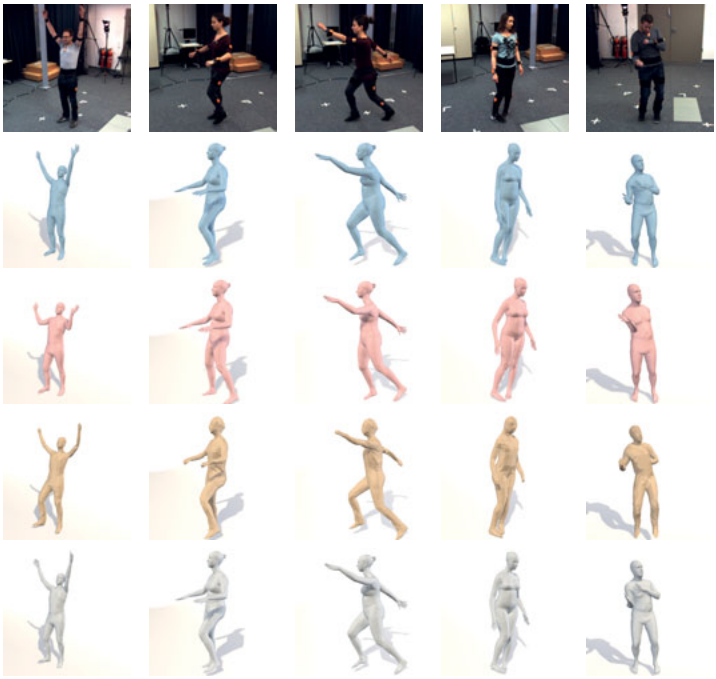


Figure 3.12: A comparison of SIP to ground truth and two baselines, the Sparse Orientation Poser (SOP), and SIP with a manually rigged body model (SIP-M). Top row: images from the TNT dataset sequences, second row: ground truth poses obtained by tracking with 10 IMUs (for reference), third row: results obtained with SOP, fourth row: results obtained with SIP-M and fifth row: results obtained with SIP. Best results are obtained with SIP. Without acceleration the pose remains ambiguous for the orientation poser (SOP) and leads to incorrect estimates, the SIP-M can disambiguate the poses by incorporating acceleration data but suffers from a limited skeletal model, which prevents the pose from appropriately fitting to the sensor data.

to the ground-truth poses obtained with a marker-based motion capture system.

Baseline Trackers

Similar to the evaluation on TNT15 in Section 3.4.2, we compare our tracking results to two baseline methods:

- *Sparse Orientation Poser* (SOP): Minimizes orientation and anthropomorphic consistency terms but disregards acceleration.
- *Inertial Tracker* (IT): Identical to SOP, but uses all $N = 13$ IMU orientations.

Again, the comparison to SOP is used to demonstrate the gain in accuracy by incorporating acceleration data. The second baseline, the IT, determines the body pose from the full set of IMU sensors. It is equivalent to the approach used to generate ground-truth poses for the TNT15 dataset. We incorporate this baseline for two reasons. First, it quantitatively shows the deviations of a marker-based motion capture systems with respect to a full-body IMU system. Second, it enables an independent evaluation of accuracy between the full sensor setup and our proposed approach using only a subset of IMUs.

Metrics

We use the MPJPE and MPJAE as defined in Section 2.5.2 to evaluate SIP against the baseline trackers. For the MPJPE we consider the joint positions of hips, knees, ankles, neck, head, shoulders, elbows and wrists. MPJAE is computed using the joint orientations of hips, knees, neck, shoulders and elbows. In order to be independent to global position and translation of the root joint, we do a procrustes alignment before computing the MPJPE.

Quantitative Results

Our tracking results are summarized in Table 3.3. By using only 6 IMUs, SIP achieves a MPJPE of 52.2 mm and a MPJAE of 14.6°. Compared to SOP, which disregards sensor acceleration, SIP is more accurate and improves the MPJPE by almost 24 mm and the MPJAE by 5.3°.

However, there is still a gap in tracking accuracy with respect to IT, which uses all 13 IMUs. While the MPJAE of SIP is only 1.4° higher, the MPJPE is approximately 21 mm worse. Hence, the proposed tracker could not resolve all ambiguities caused by using a reduced sensor set. To further investigate the reasons for this deviation, we report the tracking metrics sorted by activities in Table 3.4 and Table 3.5.

Table 3.3: Tracking errors of SOP, IT and SIP evaluated on TotalCapture.

Approach	MPJPE [mm]	MPJAE [°]
SOP	76.2 ± 61.7	19.7 ± 14.5
IT	30.5 ± 19.8	13.2 ± 8.6
SIP	52.2 ± 46.9	14.6 ± 11.5

Table 3.4: Mean Joint Position Errors (MPJPE) in [mm] of SOP, IT and SIP evaluated on TotalCapture.

Approach	Walking	Freestyle	Acting	Mean
SOP	55.3	104.1	71.3	76.2
IT	29.8	34.8	27.0	30.5
SIP	34.7	71.9	51.8	52.2

Interestingly, SIP achieves almost the same tracking metrics as IT for walking sequences. The MPJPE is only $5\,mm$ higher and the average deviations in joint angles are negligible. The main difference in tracking accuracy arises from freestyle and acting sequences. Manual inspection reveals that SIP has trouble to properly reconstruct static poses and slow motions. In these situations, measured accelerations are small and the anthropometric prior pushes the pose towards the mean pose. From this observation it can be concluded that SIP can only reconstruct rather dynamic movements. This certainly represents a limitation of the approach.

3.4.4 Qualitative Results

In order to further demonstrate the capabilities of our proposed SIP we recorded additional motions. For all recordings we have used 6 Xsens MTw IMUs [33] attached to the lower legs, wrists, head and back. The sensor placement is illustrated in Figure 3.4. Orientation and acceleration data were recorded at 60Hz and transmitted wirelessly to a laptop. Additionally, we have captured the motions with a smartphone camera to qualitatively assess the tracking accuracy.

In Figure 3.13 we show several tracking results for challenging outdoor motions, such as jumping over a wall, warming exercises, biking and climbing. For all cases, our proposed SIP approach is able to successfully track the overall motion. For most of the cases, the recovered poses are visually accurate using only 6 IMUs. Finally, in Figure 3.14 we demonstrate that SIP is capable of reconstructing the handwriting on a whiteboard. For this experiment, we attached IMUs to the lower legs, wrists, back and chest and recorded IMU data while the actor was writing “Eurographics” on a white board. The resulting wrist motion clearly resembles the hand writing.

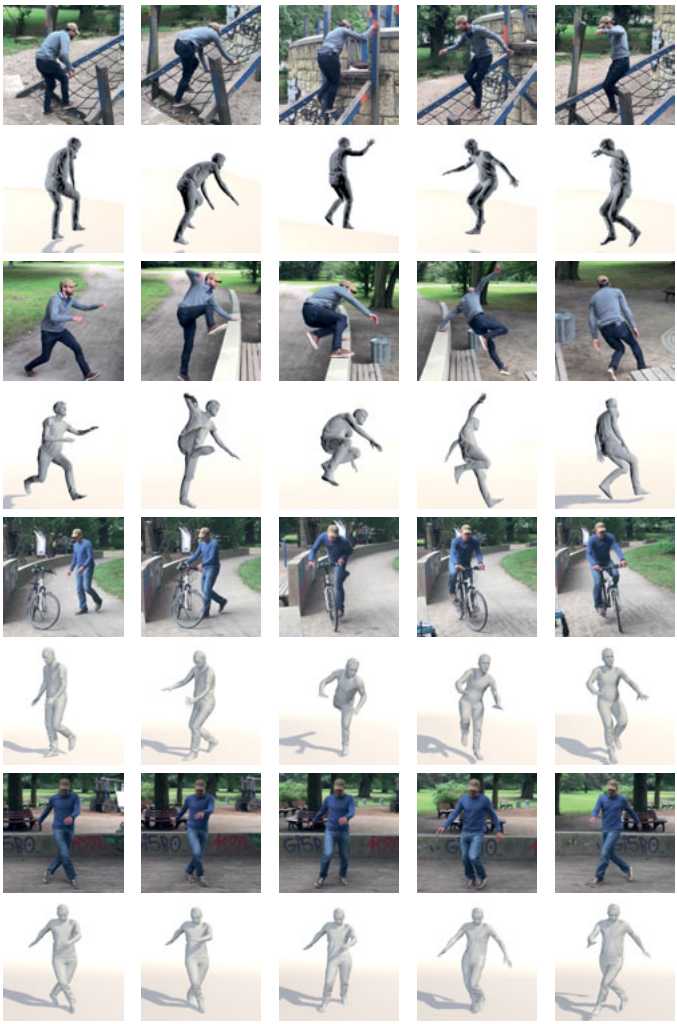


Figure 3.13: Qualitative results obtained using SIP: For most of the cases SIP successfully recovers the full human pose. This will enable to capture people performing everyday activities in a minimally intrusive way.

Table 3.5: Mean Per Joint Angular Errors (MPJAE) in [°] of SOP, IT and SIP evaluated on TotalCapture.

Approach	Walking	Freestyle	Acting	Mean
SOP	15.7	25.8	18.0	19.7
IT	11.3	16.4	12.2	13.2
SIP	11.1	19.2	14.0	14.6

Figure 3.14: Illustration of the reconstructed handwriting on a whiteboard using SIP. Left figure: image of the writing scene, middle figure: recovered pose at the end of the handwriting, right figure: recovered wrist motion projected on the whiteboard plane.

3.5 Conclusion

SIP provides a new method for estimating the pose from sparse inertial sensors. SIP makes this possible by exploiting a statistical body model and jointly optimizing pose over multiple frames to fit both orientation and acceleration data. We demonstrate that the approach works even with approximate body models obtained from a few body word ratings. Quantitative evaluation shows that SIP can reconstruct human pose accurately, with mean joint orientation errors of 13.3° and mean joint position errors of 3.9 cm. However, experiments also showed that this only holds for rather dynamic motions. In static poses, acceleration does not help do disambiguate the sparse orientation measurements, which is a limitation of this approach.

While SIP is generally able to track the full-body pose without drift, global position estimates still suffer from drift over time. This could be reduced by integrating simple physical constraints into the optimization such as center of mass preservation and ground contacts. Exploiting laws of conservation of energies is very involved whereas modeling ground contacts is comparably easier: ground contacts produce high peaks in the accelerometer signal which are easy to detect. Temporally fixing the position of body model points is straightforward to integrate in the proposed cost function and will compensate drift. However, modeling ground contacts depends on the motion to be tracked and assumes static friction [44]. Other options to compensate drift are integrating GPS measurements (e.g. from a cell carried phone on the body), or visual data, e.g. from a body mounted camera [83, 84]. The next chapter of this thesis deals with a method to compensate drift and other limitations of IMU-based motion capture with visual cues from an additional hand-held camera.

Due to the IMU placement on the body, SIP does not capture wrist and ankle joint parameters, see Section 3.2.2. While these unobserved parameters are also optimized within the anthropometric prior, one could incorporate constraints derived from the 3D world geometry. Also, instead of using static joint limits in the anthropometric term one could also incorporate pose-conditioned joint angle limits [85] to obtain physically plausible poses.

Despite the limitations, SIP provides the technology to capture human motion with as few as 6 IMUs which is much less intrusive than existing technologies. In contrast to previous work it does not rely on motion databases and generalizes to arbitrary motions. This has many potential applications in the fields of virtual reality, sports analysis, monitoring for health assessment, or recording of movement for psychological and social studies.

4 Video Inertial Poser[1]

Figure 4.1: The Video Inertial Poser (VIP) combines video obtained from a hand-held smartphone camera with data coming from body-worn IMUs. It enables to capture motions of multiple people in natural environments.

This chapter presents a method that combines measurements of IMUs attached at the body limbs and image information of a single hand-held camera to estimate accurate 3D poses in the wild. Previous methods rely on multiple static cameras, which limits the operational area and they are commonly limited to track a single person only. Instead, we present a solution that is capable to track multiple people in everyday surroundings. The visual information of the camera helps to overcome the main limitations of IMU-based motion capture: intractable root joint position, accumulating heading error and unknown sensor-to-bone alignment. However, this poses many challenges. A single hand-held camera only provides a 2D representation of the 3D world. Since we record in everyday environments, we usually have many people visible in the video, frequent occlusions and cluttered background. Also, the camera is moving which adds additional complexity as we have to simultaneously estimate camera pose. We solve this task by associating 2D pose detections in each image to the corresponding IMU-equipped persons. This is done by solving a novel graph-based optimization problem that forces 3D to 2D coherency within a frame

[1]This chapter contains previously published images, text and results [53].

and across a long range of frames. Given these associations, we jointly optimize the pose of a statistical body model, the camera pose and heading drift using a continuous optimization framework. In order to prove applicability in everyday surroundings, we recorded *3D poses in the wild* (3DPW), a new dataset consisting of more than 51.000 frames with accurate 3D pose in challenging sequences, including walking in the city, going up-stairs, having coffee or taking the bus. We validated our method on the TotalCapture dataset, which provides video and IMU synchronized with ground truth. We obtain an accuracy of $26\,mm$, which makes it accurate enough to serve as a benchmark for image-based 3D pose estimation approaches.

4.1 Introduction

The method presented in this chapter addresses two inter-related goals. First, it is capable to accurately reconstructing 3D human pose in outdoor scenes, with multiple people interacting with the environment, see Figure 4.1. Our method combines data coming from IMUs (attached at the person's limbs) with video obtained from a hand-held phone camera. This allows us to achieve the second goal, which is collecting the first dataset with accurate 3D reconstructions in the wild. Since our system works with a moving camera, we can record people in their everyday environments, for example, walking in the city, having coffee or taking the bus.

3D human pose estimation from single images and videos has been a longstanding goal in computer vision. Recently, there has been a significant progress, particularly in 2D human pose estimation [86, 87]. This progress has been possible thanks to the availability of large training datasets and benchmarks to compare research methods. While obtaining manual 2D pose annotations in the wild is fairly easy, collecting 3D pose annotations manually is almost impossible. This is probably the main reason there exist very limited datasets with accurate 3D pose in the wild. Datasets such as HumanEva [64] and Human3.6M [65] have facilitated progress in the field by providing ground truth 3D poses obtained using a marker-based motion capture system synchronized with video. These datasets, while useful and necessary, are restricted to indoor scenarios with static backgrounds, little variation in clothing and no environmental occlusions. As a result, evaluations of 3D human pose estimation methods in challenging images have been made mainly qualitatively, so far. There exist several options to record humans in natural scenes, none of which is satisfactory. Marker-based capture outdoors is limited. Depth sensors like Kinect do not work under strong illumination and can only capture objects near the camera. Using multiple cameras requires time consuming set-up and calibration [88]. Most importantly, the fixed recording volume severely limits the kind of activities that can be captured.

IMU-based systems hold promise because they are not bound to a fixed space since they are worn by the person. In practice, however, accuracy is limited by a number of

factors. Inaccuracies in the initial pose introduce sensor-to-bone misalignments. In addition, during continuous operation, IMUs suffer from heading drift, see Figure 4.2. This means, that after some time, each IMU does not measure relative to the *same* world coordinate frame. Rather, each sensor provides readings relative to *independent* coordinate frames that slowly drift away from the world frame. Furthermore, global

Figure 4.2: Illustration of IMU heading drift. The sensor heading errors have accumulated after a longer recording session and the obtained 3D pose is completely off.

position can not be accurately obtained due to positional drift, which makes it impossible to track people interactions. Moreover, IMU systems do not provide 3D pose synchronized and aligned with image data.

Therefore, we propose a new method, called VIP, that jointly estimates the pose of people in the scene by using 6 to 17 IMUs attached at the body limbs and a single hand-held moving phone camera. Even though IMUs provide much information about the pose of a person many challenges remain. First, the persons need to be detected in the video and associated with the IMU data, see Figure 4.3. Second, during continuous operation IMUs become increasingly inaccurate due to heading drift. Third, the estimated 3D poses need to align with the images of the moving camera. Furthermore, the scenes we tackle in this work include complete occlusions, multiple people, tracked persons falling out of the camera view and camera motion.

To address these difficulties, we define a novel graph-based association method, and a continuous pose optimization scheme that integrates the measurements from all frames in the sequence. To deal with noise and incomplete data, we exploit SMPL [59], which incorporates anthropometric and kinematic constraints.

Specifically, our approach has three steps: initialization, association and data fusion. During initialization, we compute initial 3D poses by fitting SMPL to the IMU orientations. The association step automatically associates the 3D poses with 2D person detections for the full sequence by solving a single binary quadratic optimization problem. Given those associations, in the data fusion step, we define

Figure 4.3: Illustration of association challenges in crowded environments. In order to combine IMU data and visual information from the camera view, 2D poses have to be associated to persons wearing IMUs. This is difficult when several people are in the scene.

an objective function and jointly optimize for the 3D poses of the full sequence, the per-sensor heading errors, the camera pose and translation. Specifically, the objective is minimized when (i) the model orientation and acceleration is close to the IMU readings and (ii) the projected 3D joints of SMPL are close to 2D CNN detections [87] in the image. To further improve results, we repeat association and joint optimization once.

With VIP we can accurately estimate 3D human poses in challenging natural scenes. To validate the accuracy of VIP, we use the Total Capture dataset [42] because it provides video synchronized with IMU data.

4.2 Model

4.2.1 Body Model

We utilize the Skinned Multi-Person Linear (SMPL) body model [59], see Section 2.2.3. We optimize the shape parameters to the person to be tracked by fitting SMPL to a 3D scan. Holding shape fixed, our aim is to recover the pose $\mathbf{x} \in \mathbb{R}^{75}$, consisting of 3 parameters for global translation and 24 relative rotations represented by exponential coordinates for each joint. We use the standard forward kinematics, defined in (2.32), to map a pose $\mathbf{x}$ to the rigid transformation $\mathbf{M}_b^g(\mathbf{x}) : \mathbb{R}^{75} \to SE(3)$ of bone F^b. The bone transformation comprises the rotation and translation $\mathbf{M}_b^g = \{\mathbf{R}_b^g, \mathbf{t}^g\}$ to map from the local bone coordinate frame F^b to

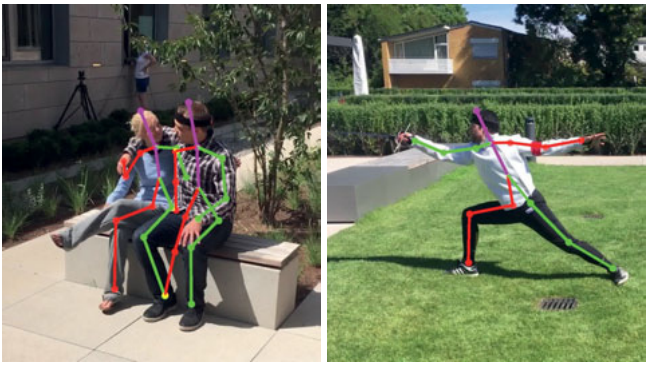


Figure 4.4: Illustrations of inaccurate 2D pose detections. Left image: current methods have trouble to correctly assign joints to persons if they are in close interaction. Right image: left and right body limbs are frequently mistaken, especially in uncommon poses.

the global SMPL frame F^g.

4.2.2 Camera Model

We apply a pinhole camera model to model the projection of a 3D point to pixel coordinates in an image [71]. In particular, a 3D point $\mathbf{p}^c := (x, y, z)^T \in \mathbb{R}^3$ defined in a camera aligned coordinate frame F^c is projected to pixel coordinates (u, v) in the image plane by

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \pi(\mathbf{K} \cdot \mathbf{x}^c) = \mathbf{K} \cdot \pi(\mathbf{p}^c), \tag{4.1}$$

where $\mathbf{K} \in \mathbb{R}^{3\times3}$ is a matrix of camera intrinsics and the operator π models the image formation process according to

$$\pi(\mathbf{p}) = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} := \frac{1}{z} \begin{bmatrix} x \\ y \\ z \end{bmatrix}. \tag{4.2}$$

The camera intrinsics $\mathbf{K}$ are defined as

$$\mathbf{K} := \begin{bmatrix} k_u f & 0 & c_u \\ 0 & k_v f & c_v \\ 0 & 0 & 1 \end{bmatrix}, \tag{4.3}$$

where

- f is the focal length of the camera,
- k_u and k_u correspond to the pixel density of the sensor, in u- and v-direction respectively, and
- c_u and c_v are the coordinates of the principal point, which is the intersection of the optical axis with the image plane.

The intrinsic camera parameters can be estimated from correspondences between known world points and corresponding image coordinates. We used a checkerboard pattern and the Mathworks MATLAB Camera Calibration Toolbox to obtain the camera intrinsics. During this process, lens distortion parameters were also estimated and for tracking images were undistorted in a pre-processing step.

In this work, 3D point coordinates (of the body model) are defined in a static global coordinate frame F^g. Hence we rewrite Eq. (4.1) according to

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \pi(\mathbf{K} \cdot \mathbf{I_c} \cdot \mathbf{M}_g^c \cdot \bar{\mathbf{p}}^g), \tag{4.4}$$

where $\mathbf{M}_g^c \in SE(3)$ corresponds to the camera pose, mapping points from the global coordinate frame to the camera coordinate frame, and $\bar{\mathbf{p}}$ is the homogeneous representation of $\mathbf{p}$. The matrix $\mathbf{I_c}$ is defined as

$$\mathbf{I_c} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{4.5}$$

and corrects for mismatching matrix dimensions. In order to avoid notational clutter we will skip this matrix in the following.

At this point, we also define the re-projection error $\mathbf{e}_{img} \in \mathbb{R}^2$ measuring the pixel difference of a 2D observation $\mathbf{p} \in \mathbb{R}^2$ in the image and the projection of a corresponding 3D point $\mathbf{q} \in \mathbb{R}^3$ under the camera model with camera pose $\mathbf{M} \in SE(3)$. For a 3D point on the body model, $\mathbf{q}$ is a function of the body pose $\mathbf{x} \in \mathbb{R}^{75}$ and we define the re-projection error of a point correspondence as a mapping $\mathbf{e}_{img} : \mathbb{R}^{75} \times SE(3) \to \mathbb{R}^2$ according to

$$e_{img}(\mathbf{x}, \mathbf{M}) = \pi(\mathbf{K} \cdot \mathbf{M}_g^c \cdot \bar{\mathbf{q}}^c(\mathbf{x})) - \bar{\mathbf{p}}. \tag{4.6}$$

4.2.3 Coordinate Frames

We introduce several coordinate systems to relate IMU measurements and visual information to the body model, depicted in Figure 4.5. Despite the camera coordinate

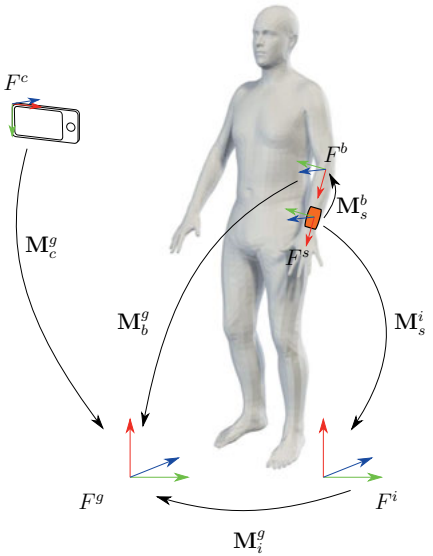


Figure 4.5: Several coordinate frames are involved to relate the body model to visual and inertial information: Global tracking frame F^g, global inertial frame F^i, bone coordinate frame F^b, IMU sensor coordinate frame F^s and camera coordinate frame F^c.

frame F^c, the coordinate frames are identical to the frames defined in Section 3.2.3 for the SIP approach. Nevertheless, we will briefly describe all frames in the following.

IMUs measure the orientation of the local coordinate frame F^s (of the sensor box) relative to a global inertial coordinate frame F^i. However, this frame F^i is different from the global reference coordinate frame F^g of SMPL, see Figure 4.5. The offset $\mathbf{M}_i^g \in SE(3)$ between these coordinate frames is typically assumed constant, and is calibrated at the beginning of a recording session. We also need to know the offset $\mathbf{R}_s^b \in SO(3)$ from the sensor to the SMPL bone where it is placed. We assume that sensors do not move relative to the bones, and hence compute $\mathbf{R}_s^b$ from the initial pose $\mathbf{x}_0$ and IMU orientations in the first frame at $t = 0$. Using the initial SMPL bone orientation $\mathbf{R}_b^g(\mathbf{x}_0)$ and the initial IMU orientation measurement $\mathbf{R}_s^i(0)$, we can compute the offset as

$$\mathbf{R}_s^b(\mathbf{x}_0) = (\mathbf{R}_b^g(\mathbf{x}_0))^{-1} \cdot \mathbf{R}_i^g \cdot \mathbf{R}_s^i(0) \tag{4.7}$$

where the raw IMU reading $\mathbf{R}_s^i(0)$ needs to be mapped to the SMPL frame first

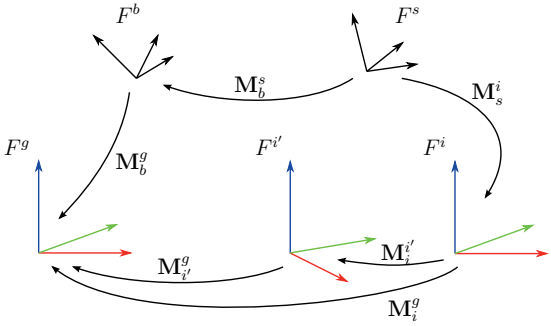


Figure 4.6: Modeling heading drift requires an additional shifted inertial frame for each sensor. The illustration shows the following coordinate frames to relate the body model to IMU data: Global tracking frame F^g, global inertial frame F^i, shifted inertial frame $F^{i'}$, bone coordinate frame F^b and IMU sensor coordinate frame F^s.

using $\mathbf{R}_i^g$.

In order to relate image cues from a hand-held camera to the body model we will use the reprojection error defined in Eq. (4.6). This requires knowledge of the camera pose, which we define as the offset $\mathbf{M}_g^c$ between a camera-fixed coordinate frame F^c and the global reference coordinate system of SMPL. Since the camera is moving, $\mathbf{M}_g^c$ is varying with time and we have to reconstruct it as part of our optimization.

4.2.4 Heading Drift

Unfortunately, the orientation measurements of the IMUs are deteriorated by magnetic disturbances, which introduce a time-varying rotational offset to $\mathbf{M}_g^i$, also commonly known as heading error or heading drift. This drift ($\mathbf{M}_i^{i'}$) shifts the original global inertial frame F^i to a disturbed inertial frame $F^{i'}$. What is even worse, the drift is different for every sensor. While most previous works ignore heading drift or treat it as noise, we model it explicitly and recover it as part of the optimization. We model it as a one-parameter rotation $\mathbf{R}(\gamma) \in SO(3)$ about the vertical axis, where γ is the rotation angle. The collection of all angles, one per IMU sensor, is denoted as Γ. Since the heading error commonly varies slowly, we assume it is constant during a single tracking sequence. Recovering heading orientation was crucial in order to be able to perform long recordings without time-consuming re-calibration.

4.2.5 Visual Cues: 2D Poses

In order to obtain pose information of humans seen in the camera view, we use an approach that regresses pixel coordinates of body landmarks from single images. We will refer to this kind of pose information as 2D poses in the following. Estimating 2D poses from images is a very active research area and recent publications show impressive performance even for very crowded scenes. A common strategy that can be found in most state-of-the-art methods is to split the task of 2D pose estimation into two stages. First, a deep CNN is commonly trained to detect dedicated joints or body parts in the image. The second stage then aims to associate these detections to individual humans.

In this work, we use the extension of the Convolutional Pose Machines framework published by Cao *et al.* [87]. For every image, the approach outputs a list of detected persons and respective 2D poses. We use the *COCO* pose parametrization which contains the image coordinates of $N_{\text{joint}} = 18$ landmark positions of hips, knees, ankles, shoulders, elbows, wrists, neck, nose, ears and eyes. Along with every 2D landmark coordinate $\mathbf{p} \in \mathbb{R}^2$, a confidence score $w \in [0, 1]$ reflects the detection uncertainty of the respective landmark. In the following, we denote a 2D pose with corresponding landmark information as a pose candidate $v_{i,t}$, where i is the i-th candidate at time t. The notion of v for a pose candidate will become more clear as each candidate will represent a vertex in a graph model later.

4.3 Method

In order to perform accurate 3D human motion capture with hand-held video and IMUs we perform three subsequent steps: initialization, pose candidate association and video-inertial fusion. Figure 4.7 provides an overview of the pipeline and we describe each step in more detail in the following.

4.3.1 Initialization

We obtain initial 3D poses by fitting the SMPL bone orientations to the measured IMU orientations. For an IMU, the *measured bone orientation* $\tilde{\mathbf{R}}_b^g$ is given by

$$\tilde{\mathbf{R}}_b^g(\mathbf{x}_0, \gamma) = \mathbf{R}_{i'}^g \cdot \mathbf{R}_i^{i'}(\gamma) \cdot \mathbf{R}_s^i \cdot \left(\mathbf{R}_s^b(\mathbf{x}_0)\right)^{-1}, \tag{4.8}$$

where $\mathbf{R}_s^b$ represents the bone to sensor offset (Eq. (4.7)), and the concatenation of $\mathbf{R}_{i'}^g$, $\mathbf{R}_i^{i'}$ and $\mathbf{R}_s^i$ describes the rotational map from sensor to global frame, see Figure 4.5 and Figure 4.6. We define the rotational discrepancy between actual bone orientation $\mathbf{R}_b^g$ and measured bone orientation $\tilde{\mathbf{R}}_b^g$ as

$$\mathbf{e}^{\text{rot}}(\mathbf{x}_t, \mathbf{x}_0, \gamma) = \log\left(\mathbf{R}_b^g(\mathbf{x}_t) \cdot \left(\tilde{\mathbf{R}}_b^g(\mathbf{x}_0, \gamma)\right)^{-1}\right)^{\vee}, \tag{4.9}$$

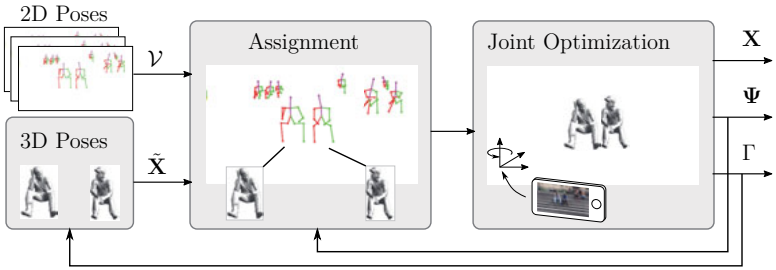


Figure 4.7: Method overview: By fitting the SMPL body model to the measured IMU orientations we obtain initial 3D poses $\tilde{\mathbf{X}}$. Given all 2D poses $\mathcal{V}$ detected in the images we search for a globally consistent assignment of 2D to 3D poses. We jointly optimize camera poses $\mathbf{\Psi}$, heading angles Γ and 3D poses $\mathbf{X}$ with respect to associated IMU and image data. In a second iteration we feed back camera poses and heading angles which provides additional information further improving the assignment and tracking results.

where the log-operation recovers the skew-symmetric matrix from the relative rotation between $\mathbf{R}_b^g$ and $\tilde{\mathbf{R}}_b^g$, and the $^\vee$-operator extracts the corresponding axis-angle parameters. We find the 3D initial poses at frame t that minimize the sum of discrepancies for all IMUs

$$\mathbf{x}_t^* = \arg\min_{\mathbf{x}} \frac{1}{N_s} \sum_{s=1}^{N_s} ||\mathbf{e}_{s,t}^{\text{rot}}(\mathbf{x}_t, \mathbf{x}_0, \gamma)||^2 + w_{\text{prior}} E_{\text{prior}}(\mathbf{x}_t), \tag{4.10}$$

where $E_{\text{prior}}(\mathbf{x})$ is a pose prior weighted by w_{prior}. $E_{\text{prior}}(\mathbf{x})$ is identical to $E_{\text{anthro}}(\mathbf{x})$ defined in Eq. (3.16), enforcing $\mathbf{x}$ to remain close to a multivariate Gaussian distribution of model poses and to stay within joint limits. During the first iteration, we have no information about the heading angles γ. To initialize them, we use the IMU placement as a proxy to know how local sensor axes are aligned with respect to the body. This gives us a rough estimate of the sensor to bone offset $\tilde{\mathbf{R}}_s^b$, which we use to compute initial heading angles by solving Eq. (4.7) for γ.

In the following, we will refer to this tracking approach simply as the inertial tracker (IT), which outputs initial 3D pose candidates $\mathbf{x}_{t,l}^*$ for every tracked person l. Such initial 3D poses need to be associated with 2D detections in the video in order to effectively fuse the data – this poses a challenging assignment problem.

4.3.2 Pose Candidate Assignment

Using the CNN method of Cao *et al.* [87], we obtain 2D pose detections v, which comprise the image coordinates of $N_{\text{joints}} = 18$ landmarks along with corresponding confidence scores. In order to associate each 2D pose v to a 3D pose candidate, we create an undirected weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, c)$, with $\mathcal{V}$ comprising all detected 2D poses in a recording sequence. An assignment hypothesis, denoted as $\mathcal{H}(l, v) := (\mathbf{x}_t^l, v)$, links the 3D pose $\mathbf{x}_t^l$ of person $l \in \{1, \ldots, P\}$ to the 2D pose $v \in \mathcal{V}$ in the same frame t. We introduce indicator variables x_v^l, which take value 1 if hypothesis $\mathcal{H}(l, v)$ is selected, and 0 otherwise. The basic idea is to assign costs to each hypothesis, and select the assignments for the sequence that minimize the total costs. We cast the selection problem as a graph labeling problem by minimizing the following objective

$$\underset{x \in \mathcal{F} \cap \{0,1\}^{|\mathcal{V}|P}}{\arg\min} \sum_{\substack{v \in \mathcal{V} \\ l \in \{1,\ldots,P\}}} c_v^l x_v^l + \sum_{\substack{\{v,v'\} \in \mathcal{E} \\ l,l' \in \{1,\ldots,P\}}} c_{v,v'}^{l,l'} x_v^l x_{v'}^{l'}, \tag{4.11}$$

where the feasibility set $\mathcal{F}$ is subject to:

$$\sum_{l=1}^{P} x_v^l \leq 1 \quad \forall v \in \mathcal{V}, \tag{4.12a}$$

$$\sum_{v \in \mathcal{V}_t} x_v^l \leq 1 \quad \forall t, \forall l \in \{1, \ldots, P\}. \tag{4.12b}$$

The edge set $\mathcal{E}$ contains all pairs of 2D poses $\{v, v'\}$ that are considered for the assignment decision. Eq. (4.12a) ensures that a 2D pose v is assigned to at most 1 person, and Eq. (4.12b) ensures that each person is assigned to at most one of the 2D pose detections $v \in \mathcal{V}_t \subset \mathcal{V}$ in frame t. The objective in (4.11) consists of unary costs c_v^l measuring 2D to 3D consistency, and pairwise costs $c_{v,v'}^{l,l'}$ measuring consistency across different hypothesis. Our formulation automatically outputs a globally consistent assignment and does not require manual initialization.

Next we describe the unaries and pairwise potentials – specifically, we introduce consistency features which are mapped to the costs $c_v^l, c_{v,v'}^{l,l'}$ of the objective in (4.11) via logistic regression. Details about the training process are described in Section 4.4.1. Figure 4.8 visualizes the graph for two example frames and also illustrates the corresponding labeling solution.

Unary Costs

To measure 2D to 3D consistency of a hypothesis $\mathcal{H} := \mathcal{H}(l, v)$, we obtain a *hypothesis camera* $\mathbf{M}_{\mathcal{H}}$ by minimizing the re-projection error between 3D landmarks of $\mathbf{x}_t^l$ and the 2D detected ones v. The consistency of a hypothesis is then measured by

$$e_{\text{img}}(\mathcal{H}, \mathbf{M}_{\mathcal{H}}) = \frac{1}{N_{\text{joints}}} \sum_{k=1}^{N_{\text{joints}}} w_k \cdot ||\mathbf{e}_{\text{img,k}}(\mathcal{H}, \mathbf{M}_{\mathcal{H}})||, \tag{4.13}$$

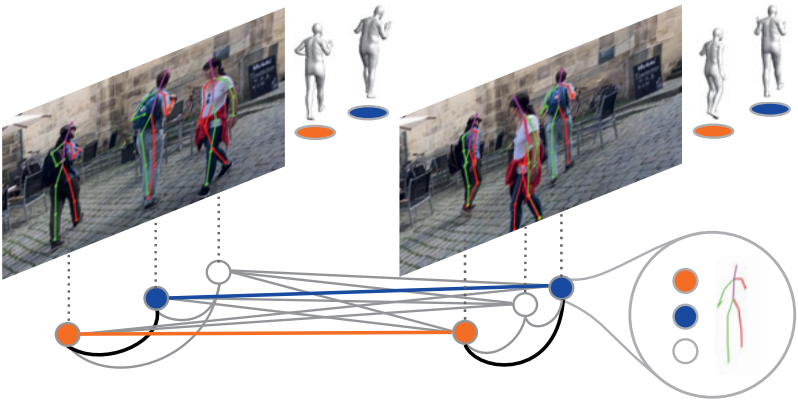


Figure 4.8: Graph labeling illustration. Every 2D pose represents a node in the graph which can be assigned to a 3D pose corresponding to person 1 or 2 (represented by colors orange and blue). The graph has intra-frame edges (shown in black) activated if two nodes are assigned in a single frame and inter-frame edges (shown in blue and orange) activated for the same person across multiple frames.

where $\mathbf{e}_{\text{img,k}}(\mathcal{H}, \mathbf{M}_{\mathcal{H}})$ is the per landmark re-projection error and w_k is the corresponding landmark confidence score. This measure depends heavily on the distance to the camera. To balance it, we scale it by the average 3D joint distance to the camera center $e_{\text{cam}}(\mathbf{M}_{\mathcal{H}})$ to obtain the feature:

$$f_{\text{un}}(\mathcal{H}) = e_{\text{img}}(\mathcal{H}, \mathbf{M}_{\mathcal{H}}) \cdot e_{\text{cam}}(\mathcal{H}, \mathbf{M}_{\mathcal{H}}). \tag{4.14}$$

Pairwise Costs

We define features to measure the consistency of two hypothesis $\mathcal{H} := (\mathbf{x}_t^l, v)$ and $\mathcal{H}' = (\mathbf{x}_{t'}^{l'}, v')$ in frames t and t'. In particular, two kinds of edges connect hypothesis: *(a) inter-frame*, and *(b) intra-frame*.

a) Inter-frame: Consider two hypothesis $\mathcal{H}, \mathcal{H}'$ corresponding to the *same person* and separated less than 30 frames. Then, the respective root joint position $\mathbf{r}(\mathbf{x}_t^l) \in \mathbb{R}^3$ and orientation $\mathbf{R}(\mathbf{x}_t^l) \in SO(3)$ in camera hypothesis $(\mathbf{M}_{\mathcal{H}})$ coordinates should not vary too much. This variation depends on the temporal distance $|t-t'|$. Consequently,

we introduce the following features

$$f_{\text{trans}}(\mathcal{H}, \mathcal{H}') = ||\mathbf{M}_{\mathcal{H}} \cdot \mathbf{r}(\mathbf{x}_t^l) - \mathbf{M}_{\mathcal{H}'} \cdot \mathbf{r}(\mathbf{x}_{t'}^{l'})||, \tag{4.15}$$

$$f_{\text{ori}}(\mathcal{H}, \mathcal{H}') = \left\| \log\left((\mathbf{R}_{\mathcal{H}} \cdot \mathbf{R}(\mathbf{x}_t^l))^{-1} (\mathbf{R}_{\mathcal{H}'} \cdot \mathbf{R}(\mathbf{x}_{t'}^{l'})) \right)^{\vee} \right\|, \tag{4.16}$$

$$f_{\text{time}}(\mathcal{H}, \mathcal{H}') = |t - t'|, \tag{4.17}$$

where f_{trans} and f_{ori} measure root joint translation and orientation consistency, and f_{time} is a feature to accommodate for temporal distance. Here, $\mathbf{R}_{\mathcal{H}}$ is the rotational part of $\mathbf{M}_{\mathcal{H}}$, and f_{rot} computes the geodesic distance between $\mathbf{R}(\mathbf{x}_t^l)$ and $\mathbf{R}(\mathbf{x}_{t'}^{l'})$, similar to Eq. (4.9).

b) Intra-frame: Now consider two hypothesis $\mathcal{H}, \mathcal{H}'$ for *different persons* in the same frame. The resulting camera hypothesis centers should be consistent. To measure coherency, we compute a meta-camera hypothesis $\mathbf{M}_{\underline{\mathcal{H}}}$ by minimizing the re-projection error of both hypothesis at the same time. Then the feature

$$f_{\text{intra}}(\mathcal{H}, \underline{\mathcal{H}}) = ||\mathbf{c}(\mathbf{x}_t^l, \mathbf{M}_{\mathcal{H}}) - \mathbf{c}(\mathbf{x}_t^l, \mathbf{M}_{\underline{\mathcal{H}}})|| \tag{4.18}$$

measures the camera center $\mathbf{c}(\mathbf{x}_t^l, \mathbf{M}_{\mathcal{H}})$ to meta-camera center $\mathbf{c}(\mathbf{x}_t^l, \mathbf{M}_{\underline{\mathcal{H}}})$ difference. Accordingly, we also use the feature $f_{\text{intra}}(\mathcal{H}', \underline{\mathcal{H}})$ for intra-frame edges.

Graph Optimization

Although the presented graph labeling problem in (4.11) is NP-Hard, it can be solved efficiently in practice [89, 90]. We use the binary LP solver Gurobi [91] by applying it to the linearized formulation of (4.11), where we replace each product $x_v^l x_{v'}^{l'}$ by a binary auxiliary variable $y_{v,v'}^{l,l'}$ and add corresponding constraints such that $x_v^l x_{v'}^{l'} = y_{v,v'}^{l,l'}$ for all $v, v' \in \mathcal{V}$, for all $l, l' \in \{1, \ldots, P\}$.

4.3.3 Video-Inertial Data Fusion

Once the assignment problem is solved we can utilize the associated 2D poses to jointly optimize model poses, camera poses and heading angles by minimizing the following energy:

$$\begin{aligned} E(\mathbf{X}, \mathbf{\Psi}, \Gamma) = \; & E_{\text{ori}}(\mathbf{X}, \Gamma, \mathbf{\Psi}) + w_{\text{acc}} E_{\text{acc}}(\mathbf{X}, \Gamma, \mathbf{\Psi}) + \\ & w_{\text{img}} E_{\text{img}}(\mathbf{X}, \mathbf{\Psi}) + w_{\text{prior}} E_{\text{prior}}(\mathbf{X}), \end{aligned} \tag{4.19}$$

where $\mathbf{X}$ is a vector containing the pose parameters for each actor and frame, Γ is the vector of IMU heading correction angles and $\mathbf{\Psi}$ contains the camera poses for each frame. $E_{\text{ori}}, E_{\text{acc}}$ and E_{img} are energy terms related to IMU orientations, IMU accelerations and image information, respectively. E_{prior} is an energy term related to pose priors. Finally, every term is weighted by a corresponding weight w.

Orientation Term

The orientation term is a sum of two parts:

$$E_{\text{ori}}(\mathbf{X}, \Gamma, \mathbf{\Psi}) = \frac{1}{N_T N_s}(E_{\text{ori,bone}}(\mathbf{X}, \Gamma) + E_{\text{ori,cam}}(\Gamma, \mathbf{\Psi})), \tag{4.20}$$

where $E_{\text{ori,bone}}$ refers to the orientation discrepancy between IMU measurements with respect to the body model and $E_{\text{ori,cam}}$ models the coherency of estimated and measured camera orientation. The sum of both terms is normalized by the total number of frames N_T in a recording sequence and the total number of IMUs N_S.

$E_{\text{ori,bone}}$ contains the bone to sensor orientation errors of Eq. (4.10), accumulated for all frames and all body-worn IMUs:

$$E_{\text{ori,bone}}(\mathbf{X}, \Gamma) = \sum_{t=0}^{N_T-1} \sum_{s=0}^{N_S-2} ||\mathbf{e}_{s,t}^{\text{rot}}(\mathbf{x}_t, \mathbf{x}_0, \gamma_s)||^2. \tag{4.21}$$

The second term $E_{\text{ori,cam}}(\mathbf{X}, \mathbf{\Psi})$ refers to the deviation of estimated camera orientation $\mathbf{R}_c^g$ and the corresponding measured IMU orientation $\tilde{\mathbf{R}}_c^g$ according to

$$E_{\text{ori,cam}}(\Gamma, \mathbf{\Psi}) = \sum_{t=0}^{N_T-1} ||\log\left(\mathbf{R}_c^g(t) \cdot \left(\tilde{\mathbf{R}}_c^g(t, \mathbf{x}_0, \gamma)\right)^{-1}\right)^{\vee}||^2. \tag{4.22}$$

Note that the estimated camera orientation $\mathbf{R}_c^g$ is given by the inverse rotational part of camera pose $\mathbf{M}_g^c$.

Acceleration Term

The acceleration term enforces consistency of the measured IMU acceleration and the acceleration of the corresponding model vertex, where the IMU is attached to. Similar to the orientation term, the acceleration term is also a sum of two parts:

$$E_{\text{acc}}(\mathbf{X}, \Gamma, \mathbf{\Psi}) = \frac{1}{N_T N_s}(E_{\text{acc,bone}}(\mathbf{X}, \Gamma) + E_{\text{acc,cam}}(\Gamma, \mathbf{\Psi})). \tag{4.23}$$

In order to compare measured accelerations with the acceleration of model points, we first have to transform the sensor acceleration $\mathbf{a}(t)^s$ at time t into world coordinates by

$$\mathbf{a}^g(t, \gamma) = \mathbf{R}_{i'}^g \cdot \mathbf{R}_i^{i'}(\gamma_s) \cdot \mathbf{R}_s^i \cdot \mathbf{a}^s(t) - \mathbf{g}^g, \tag{4.24}$$

where $\mathbf{g}^g$ is gravity in global coordinates.

For the body-worn IMUs, the corresponding SMPL vertex acceleration $\tilde{\mathbf{a}}_{s,t}$ is approximated by finite differences:

$$\tilde{\mathbf{a}}^g(\mathbf{x}_{t-1}, \mathbf{x}_t, \mathbf{x}_{t+1}) = \frac{\mathbf{p}^g(t-1) - 2 \cdot \mathbf{p}^g(t) + \mathbf{p}^g(t+1)}{dt^2}, \tag{4.25}$$

where $\mathbf{p}^g(t)$ is the vertex position in global coordinates and dt is the sampling interval. The acceleration term $E_{\text{acc,bone}}$ then contains the quadratic norm of the deviation of measured and estimated acceleration for each body-worn IMU over all frames:

$$E_{\text{acc,bone}}(\mathbf{X}, \Gamma) = \sum_{t=0}^{N_T-1} \sum_{s=0}^{N_S-2} ||\tilde{\mathbf{a}}_s^g(\mathbf{x}_{t-1}, \mathbf{x}_t, \mathbf{x}_{t+1}) - \mathbf{a}_s^g(t, \gamma_s)||^2. \tag{4.26}$$

Similarly, the acceleration $\tilde{\mathbf{c}}^g(t)$ of the camera center $\mathbf{q}^g$ with respect to the global coordinate frame F^g is approximated by finite differences:

$$\tilde{\mathbf{c}}^g(t) = \frac{\mathbf{q}^g(t-1) - 2\mathbf{q}^g(t) + \mathbf{q}^g(t+1)}{dt^2}. \tag{4.27}$$

The camera center can be computed from the camera pose $\mathbf{M}_g^c$ according to

$$\mathbf{q}^g(t) = -\left(\mathbf{R}_g^c\right)^T \cdot \mathbf{t}^c. \tag{4.28}$$

here, $\mathbf{R}_g^c$ refers to the rotational part of $\mathbf{M}_g^c$ and $\mathbf{t}^c$ is the associated translation. The camera acceleration term for the camera IMU is then defined as

$$E_{\text{acc,cam}}(\Gamma, \mathbf{\Psi}) = \sum_{t=0}^{N_T-1} ||\tilde{\mathbf{c}}^g(t) - \mathbf{a}_s^g(t, \gamma_s)||^2. \tag{4.29}$$

Image Term

The image term simply accumulates the re-projection error over all N_{joints} landmarks and all frames N_T according to

$$E_{\text{img}}(\mathbf{X}, \mathbf{\Psi}) = \frac{1}{N_T N_{\text{joints}}} \sum_{t=1}^{N_T} \sum_{i=k}^{N_{\text{joints}}} w_k ||\mathbf{e}_{\text{img},k}(\mathbf{x}_t, \mathbf{M}_t)||^2, \tag{4.30}$$

where w_k is the confidence score associated with a landmark.

Prior Term

The prior term is the same as in Eq. (4.10), now accumulated for all poses $\mathbf{X}$ and scaled by the number of poses $N_{\mathbf{X}}$.

4.3.4 Optimization

In order to solve the optimization problems related to obtaining initial 3D poses in Eq. (4.10), obtaining camera poses to minimize re-projection error and to jointly optimize all variables in Eq. (4.19), we apply the gradient-based Levenberg-Marquardt algorithm described in Section 2.3.2.

4.4 Evaluation

We evaluate our approach quantitatively on the TotalCapture dataset. This dataset provides IMU data and video synchronized with ground-truth poses obtained with a marker-based motion capture system. We use TotalCapture to evaluate tracking accuracy and perform several ablation studies to investigate the influence of tracking parameters of our proposed tracker. Additionally, we provide details of the newly recorded 3DPW dataset and demonstrate 3D pose reconstructions of VIP in challenging outdoor scenes. We also use 3DPW to evaluate the accuracy of our automatic assignment of 2D poses to 3D poses. This accounts for natural situations with multiple people visible at the same time, which is not covered in TotalCapture.

4.4.1 Tracker Setup

Inputs

In order to run our proposed VIP we require

- a SMPL body model of the actor,
- a rough initial pose at the beginning of the sequence,
- IMU sensor locations on the body,
- 2D poses obtained from the video.

We use the initial pose and sensor locations to get a rough estimate of the sensor to bone offsets $\mathbf{M}_s^b$ and initial heading offsets. Similar to the SIP approach of Chapter 3, we manually selected the SMPL vertices of approximate sensor locations and use them for all actors and experiments. For the quantitative analysis, initial pose parameters are set to zero, corresponding to a T-pose for the SMPL body model. For the outdoor recordings we simply asked the actor to pose upright with straight arms and legs at the beginning of each sequence. We obtained SMPL body models by fitting the SMPL template to laser scans or marker locations, respectively. In order to get 2D poses of all people visible in the video we use the approach of Cao *et al.* [87]. The general tracking procedure is divided into three subsequent steps, described in Section 4.3 and visualized in Figure 4.7.

Pose Assignment

In the graph $\mathcal{G}$, edges $e \in \mathcal{E}$ are created between any two nodes that are at most 30 frames apart. This corresponds to a time span of $1s$, which is a reasonable

compromise between motion plausibility and long-term assignment consistency. Motion plausibility, enforced by the pairwise costs, becomes less descriptive for longer time spans or overly constrains admissible motion velocities. In contrast, the graph should contain large temporal dependencies to gap longer periods of non-visibility of a person.

The logistic regression weights mapping from features to costs, are learned using 5 sequences from 3DPW dataset, which have been manually labeled for this purpose. Given the features $\mathbf{f}$ defined in Section 4.3.2 and learned weights α from logistic regression, we turn features into costs via $c = -\langle \mathbf{f}, \alpha \rangle$, making the optimization problem (4.11) probabilistically motivated [92].

Video-inertial Fusion

Different weighting parameters in Eq. (4.10) and Eq. (4.19) produce good results as long as they are balanced. However, rather than setting them by hand, we used Bayesian Optimization [93] in the proposed training set of TotalCapture (seen subjects). The values found are $w_{\text{acc}} = 0.2$, $w_{\text{img}} = 0.0001$ and $w_{\text{prior}} = 0.006$ and are kept fixed for all experiments. Note, that these are very few parameters and therefore, there is very little risk of over-fitting, which is also reflected in the results.

4.4.2 Evaluation on TotalCapture

We quantitatively evaluate tracking accuracy on the TotalCapture dataset. The dataset consists of 5 subjects performing several activities such as walking, acting, range of motions and freestyle motions, which are recorded using 8 calibrated, static RGB-cameras and 13 IMUs attached to head, sternum, waist, upper arms, lower arms, upper legs, lower legs and feet. Ground-truth poses are obtained using a marker-based motion capture system. For more details on the dataset, see Section 2.5.1.

To run our tracker, we only use one camera and the full set of 13 IMUs. The cameras in TotalCapture are rigidly mounted to the building and are not equipped with an IMU. Hence we assume a static camera with *unknown* pose.

Baseline Trackers

We compare our tracking results to three baseline trackers:

- Inertial Tracker (IT): Minimizes orientation and pose prior terms according to Eq. (4.10).
- Trumble [42]: A hybrid tracking approach of Trumble *et al.* published along

TotalCapture. It fuses IMU data with a probabilistic visual hull obtained from *all 8 cameras*.

- Malleson [43]: A real-time approach of Malleson *et al.* to estimate pose from IMU data and 2D poses obtained from multiple camera views.

The Inertial tracker (IT) corresponds to the single frame approach of Section 4.3.1. It uses only raw IMU orientations of all 13 sensors and is the initialization for VIP. We consider the trackers of Trumble *et al.* [42] and Malleson *et al.* [43] because they also report tracking accuracy on TotalCapture and are comparable to our approach in terms of the sensor modalities used for tracking.

Trumble uses a *Convolutional Neural Network* (CNN) to regress a volumetric probabilistic visual hull from the different camera views and a subsequent *Neural Network* (NN) of fully-connected layers to fuse the data with the pose estimate obtained from the IMU orientations. While different number of cameras have been tested in their work, we only refer to the 8 camera setup for this evaluation. The tracker of Malleson minimizes an energy function similar to Eq. (4.19). Besides orientation and acceleration consistency it also considers the re-projection error with respect to 2D joint locations detected in the images. Their method runs in real-time and uses a robust loss function to reject false joint detections. Since it does include a dedicated pose assignment strategy, it can only track a single person. Also, heading errors and sensor-to-bone misalignments are not considered in their model. The approach is evaluated for different numbers of cameras and IMUs. However, for this evaluation we only consider the setting that incorporates all 8 cameras and 13 IMUs, which showed best performance on TotalCapture.

One has to keep in mind, that the comparison of selected baselines with VIP is not completely fair. They all process the data in a frame-by-frame manner which is an advantage w.r.t. VIP, which jointly optimizes over all frames simultaneously. However, VIP uses only a single camera with unknown pose whereas the approaches of Trumble and Malleson *use 8 fully calibrated cameras*.

Metrics

We report two error metrics: Mean Per Joint Position Error (MPJPE) and Mean Per Joint Angular Error (MPJAE), as defined in Section 2.5.2. For the MPJPE we consider the SMPL joint positions of hips, knees, ankles, neck, head, shoulders, elbows and wrists. The baselines of Trumble and Malleson report the MPJPE with respect to a skeletal model obtained from the marker-based reference system. This skeletal model only comprises joint locations, hence angular errors are not presented. Even though the ground-truth models used for evaluation are different, we consider the MPJPE a valid metric for comparison as it is computed w.r.t. ground-truth poses derived from identical marker positions. For the MPJAE we consider the joint orientations of hips, knees, neck, shoulders and elbows.

Quantitative Results

Our tracking results on TotalCapture are summarized in Table 4.1, Table 4.2 and Table 4.3. VIP achieves a remarkable low average MPJPE of 26 mm and a MPJAE of only 12.1°. Over all sequences, IT achieves a MPJPE of 55 mm and a MPJAE of 16.9°. Hence, VIP decreases these errors by more than 50% and 25%, respectively. This demonstrates the usefulness of fusing image information and optimizing heading angles.

Table 4.1: Mean Per Joint Position Error (MPJPE) and Mean Per Joint Angular Error (MPJAE) obtained on the TotalCapture dataset.

Approach	MPJPE [mm]	MPJAE [°]
Trumble [42]	70.0	-
Malleson [43]	62.0	-
IT	55.0 ± 36.9	17.5 ± 9.8
VIP	26.0 ± 17.9	12.3 ± 7.9

Table 4.2: Mean Joint Position Errors (MPJPE) in [mm] for the TotalCapture dataset.

Approach	Walking	Freestyle	Acting	Mean
IT	54.5	57.3	53.2	55.0
VIP	21.9	32.2	24.2	26.0

Table 4.3: Mean Per Joint Angular Errors (MPJAE) in [°] for the TotalCapture dataset.

Approach	Walking	Freestyle	Acting	Mean
IT	15.6	20.5	16.8	17.5
VIP	10.5	15.3	11.2	12.3

In Figure 4.9 we show the MPJPE of VIP and IP for an example tracking sequence. VIP achieves a lower error during all frames of the sequence. In Figure 4.10, we also show an example frame of estimated and ground-truth poses applied to the SMPL body model. Both body models are almost perfectly aligned, demonstrating the accuracy of VIP.

VIP clearly outperforms the learning based approach of Trumble by 44 mm. This approach uses *all 8 cameras* for video-inertial data fusion. We also outperform Malleson, who report a mean MPJPE of 62 mm using 8 cameras and all 13 IMUs. Interestingly, IT, without using any visual input, also achieves a lower MPJPE

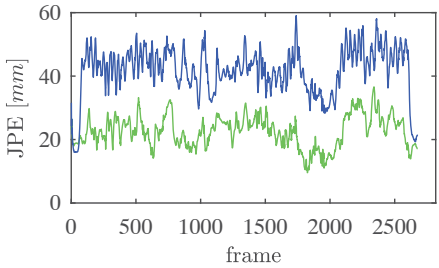
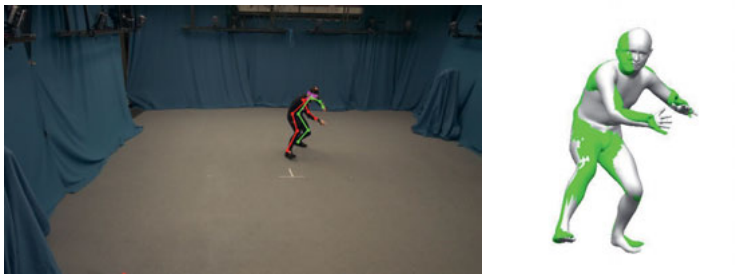


Figure 4.9: Mean Joint Position Error of the inertial tracker (blue) and our proposed method (green) for an example sequence of TotalCapture.



(a) (b)

Figure 4.10: Illustration of the tracking performance on TotalCapture. (a) camera view and detected 2D pose, (b) Overlay of ground-truth (green) and estimated pose (grey) of the corresponding frame shown in (a).

compared to approaches of Trumble and Malleson. We informed the authors about this inconsistency. In addition, we suspect that the MPJPE may have been calculated differently.

In order to validate that modeling heading angles and sensor-to-bone misalignments originating from an inaccurate initial pose are crucial and also accurately estimated by VIP we report tracking results of VIP-IT and VIP-vanilla in Table 4.4. VIP-vanilla is similar to VIP but does not optimize heading angles and initial pose. During optimization this causes inconsistencies between IMU related and video related energy terms leading to a deterioration in tracking accuracy to an MPJPE of 45.8 mm and a MPJAE of 18.0°. This validates the importance of modeling these components.

Table 4.4: Tracking accuracy on TotalCapture for tracker variants with different heading and initial pose settings.

Approach	MPJPE [mm]	MPJAE [°]
IT	55.0	17.5
VIP-vanilla	45.8	18.0
VIP-IT	28.2	12.2
VIP	26.0	12.3

VIP-IT is identical to IT, but it uses the estimated heading angles and initial pose obtained by running the original VIP method in advance. Hence VIP-IT itself does not utilize any visual information during optimization. The tracking results of VIP-IT are almost en-par with VIP. This validates the accuracy of estimated heading errors and initial pose. Also, this shows that VIP can accurately reconstruct the pose even if the video information is not available all the time. This could be the case, if a person is occluded or a 2D pose candidate has not been selected during the assignment step. In the following paragraphs various experiments are presented to further investigate the influence of VIP parameters and components.

Initial Pose Optimization

We assess the influence of estimating sensor-to-bone-misalignments in isolation by disabling the optimization of heading parameters. We call this tracker variant VIP-noHeadParam. On TotalCapture it achieves a MPJPE of 25.9mm and a MPJAE of 12.3°. Hence it clearly outperforms VIP-vanilla, which is almost identical but does not optimize the initial pose. Hence, optimizing the initial pose is crucial to this method. VIP-noHeadParam is also marginally better than VIP, even without modeling heading errors. However, the method is based on gradient descent and the differences can be explained from landing in different local minimums. Also, during the recordings of TotalCapture, the IMU reference coordinate frames are re-calibrated frequently. Consequently, the heading errors are rather small. The tracking accuracy of VIP-noHeadParam, VIP-vanilla and VIP are summarized in Table 4.5.

Table 4.5: Tracking accuracy on TotalCapture for different VIP variants demonstrating the influence of initial pose optimization.

Approach	MPJPE [mm]	MPJAE [°]
VIP-noHeadParam	25.9	12.3
VIP-vanilla	45.8	18.0
VIP	26.0	12.3

Heading Drift Estimation

In this section we evaluate the influence of modeling heading drift. In order to investigate this model parameter in isolation, we utilize ground-truth initial poses (indicated by gtip) and held it fixed during optimization. Otherwise heading errors might also be compensated to some extend by adjusting bone rotations about the vertical axis in the initial frame.

First, we compare the tracking accuracy of VIP-gtip and its corresponding variant VIP-gtip-noHead, which does not estimate heading drift. According to Table 4.6, VIP-gtip performs better than VIP-gtip-noHead but the difference in accuracy is rather small. This is probably due to frequent re-calibration of heading angles during dataset recordings. However, without time-consuming frequent re-calibration, the heading error is unbounded.

Table 4.6: Tracking accuracy on TotalCapture for tracker variants with different heading settings using ground-truth initial pose (gtip).

Approach	MPJPE $[mm]$	MPJAE $[°]$
VIP-gtip	25.3	12.4
VIP-gtip-noHeadPar	26.5	13.1
VIP-gtip-25	25.3	12.4
VIP-gtip-45	25.2	12.5
VIP-gtip-25-noHeadPar	34.5	15.8
VIP-gtip-45-noHeadPar	50.3	20.2
VIP	26.0	12.3

For VIP we use the IMU placement as a proxy to know how local sensor axes are aligned to the body. This gives a rough estimate of the sensor to bone offset, which we use to compute initial heading angles. In order to investigate how this approximation affects tracking accuracy we report tracking results of VIP-syntheticHeading25 and VIP-syntheticHeading45. For both variants the IMU heading angles are synthetically distorted by a rotation angle sampled from a uniform distribution within the interval of $[-25, 25]$ and $[-45, 45]$ degrees, respectively. These ranges are chosen to model uncertainty in the IMU placements and hence uncertainty in the initial heading estimates. The lower range of $[-25, 25]$ degrees represents careful IMU positioning and the higher range of $[-45, 45]$ degrees represents an imprecise IMU attachment. For each range, we repeat the experiment ten times and report the accuracy metrics averaged over all runs in Table 4.6. The synthetic heading error has a large impact on tracking accuracy if heading errors are not considered in the model: For VIP-gtip-25-noHeadPar the MPJPE increases from $25.3\,mm$ to $34.5\,mm$ and the MPJAE from $12.4°$ to $15.8°$. For VIP-gtip-45-noHeadPar the MPJPE even increases to $50.3\,mm$ and the MPJAE to $20.2°$. In contrast, the tracker variants with heading optimization

enabled, VIP-gtip-25 and VIP-gtip-45, still achieve the same tracking metrics as for the undistorted IMU signals. This validates the importance of modeling heading errors and shows, that VIP can cope with imprecise sensor placements.

Ground-truth 2D poses

In order to investigate how much VIP can improve if 2D pose estimation methods keep improving, we report tracking accuracy of VIP-2D in Table 4.7. VIP-2D is identical to VIP, but utilizes ground-truth 2D poses obtained by projecting ground-truth joint positions to the images. VIP-2D achieves a MPJPE of 15.1 mm and a MPJAE of 10.2°, which indicates how much VIP can improve if 2D pose estimation methods keep improving.

Table 4.7: Tracking accuracy on TotalCapture using ground-truth joint detections.

Approach	MPJPE [mm]	MPJAE [°]
VIP-2D	15.1	10.2
VIP	26.0	12.3

Ground-truth camera

VIP works with a hand-held camera and estimating camera pose is an important part of the method to reliably fuse visual and inertial information. In order to evaluate how much accuracy improves with a known camera pose, we report tracking accuracy of VIP-Cam in Table 4.8. VIP-Cam is identical to VIP but uses ground-truth camera pose instead of estimating it. The MPJPE of VIP-Cam is 25.3 mm, which is only 0.7 mm better compared to VIP. Hence, even though we estimate the camera pose from a monocular view, VIP is capable to accurately reconstruct the relative pose between actor and camera.

Table 4.8: Tracking accuracy on TotalCapture using ground-truth camera pose.

Approach	MPJPE [mm]	MPJAE [°]
VIP-Cam	25.3	12.2
VIP	26.0	12.3

Sparse IMU setup

In Chapter 3, we present the Sparse Inertial Poser which recovers 3D pose from only a sparse set of inertial sensors. In this section, we compare SIP to VIP and

evaluate the influence of incorporating additional information from the camera view. In particular, we report tracking results of

- SIP-gtInitPose: This tracker corresponds to the SIP approach, presented in Chapter 3. It uses 6 IMUs and ground-truth initial pose provided in TotalCapture.
- SIP-zeroInitPose: Identical to SIP, but uses an initial pose with all parameters set to zero. This corresponds to the standard setting for VIP experiments, which is more realistic in a natural scenario.
- VIP-6IMU: Identical to VIP, but uses the same sparse set of IMUs as the SIP tracker variants.

The tracking results are summarized in Table 4.9. By comparing SIP-gtInitPose and SIP-zeroInitPose we see that an inaccurate initial pose leads to a deterioration in tracking accuracy for SIP. The joint position error increases from 52.2 mm to 68.7 mm. However, in TotalCapture subjects start in a T-pose, which permits a lot of variation in chest, shoulder and arm regions. This adds a lot of variation to the initial pose, which can be avoided by selecting another more reproducible pose. Alternatively, the tracking results of VIP-6IMU indicate, that adding visual information from a single hand-held camera suffices to compensate much of the initial pose uncertainty. VIP-6IMU achieves a MPJPE of 35.8 mm and a MPJAE of 14.7°. This is not as accurate as VIP with 13 IMUs, but it still remarkably accurate for the sparse sensory input.

Table 4.9: Tracking accuracy on TotalCapture using sparse inertial sensor inputs.

Approach	MPJPE [mm]	MPJAE [°]
SIP-gtInitPose	52.2	14.6
SIP-zeroInitPose	68.7	18.6
VIP-6-zeroInitPose	35.8	14.7
VIP	26.0	12.3

This quantitative evaluation demonstrates the accuracy of VIP. Ideally, we would evaluate VIP quantitatively also in challenging scenes, like the ones in 3DPW. However, there exists no dataset with a comparable setting with ground-truth, which was one of the main motivations of this work.

4.4.3 Evaluation on 3DPW

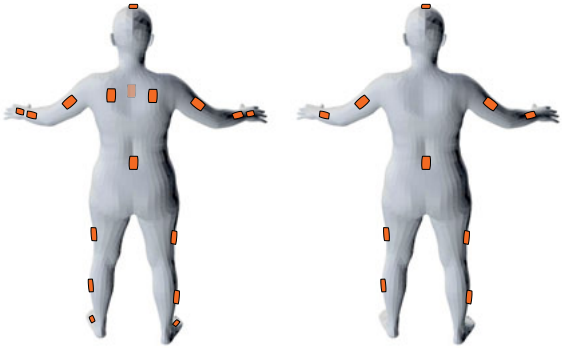
In comparison to TotalCapture, the additional challenges in 3DPW originate from multiple people in the scene. Hence, we assess the accuracy of our automatic

assignment of 2D poses to 3D poses using manually labelled 2D pose candidate IDs.

3DPW Dataset

The 3DPW dataset contains synchronized streams from a hand-held smartphone camera and one or two IMU-equipped actors performing various activities such as shopping, doing sports, hugging, discussing, capturing selfies, riding bus, playing guitar, relaxing. In total, the dataset includes 60 sequences, more than 51000 frames and 7 actors. It also provides non-rigidly fitted SMPL models of the actors similar to Zhang *et al.* [94] and Pons-Moll *et al.* [95].

Due to a limited number of IMU devices, different sensor setups have been used for single or multi-person tracking, see Figure 4.11. For single subject tracking, we attached 17 IMUs to all major bone segments. We used $9 - 10$ IMUs per person to simultaneously track up to 2 subjects. During all recordings one additional IMU was attached to the smartphone.



(a) (b)

Figure 4.11: IMU placement in the 3DPW dataset. In total, 20 IMUs were available for recordings. One IMU was always attached to the camera and the rest was distributed to the recorded subjects, depending on the number of participants. (a) for single person recordings the sensors are strapped to feet, lower legs, upper legs, waist, shoulders, sternum, head, upper arms, lower arms and hands. (b) for two person recordings sensors are strapped to lower legs, upper legs, waist, upper arms and lower arms. One additional IMU was attached to the head of one out of the two subjects.

Video and inertial data was automatically synchronized by a clapping motion at the beginning of a sequence as proposed by Pons-Moll *et al.* [41]. For every sequence,

the subjects were asked to start in an upright pose with closed arms.

In Figure 4.12 we show qualitative tracking results obtained with VIP, illustrating the 3D model alignment with the images. Figure 4.13 shows more tracking results, where we animated the 3D models with the reconstructed poses.

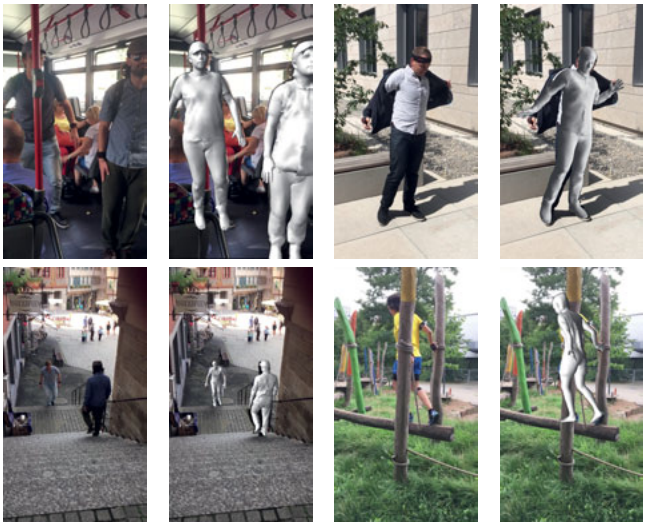


Figure 4.12: Illustration of the tracking performance of VIP for some challenging activities. With VIP we get accurate 3D poses aligning well with the images using the estimated camera poses.

Assignment Accuracy

We use the 3DPW dataset to examine pose candidate assignment accuracy of VIP under challenging real world conditions. For this purpose, we manually created ground-truth assignments for each frame. Unfortunately, 2D pose candidates are not perfect, making the labeling task non-trivial. Frequently, some landmarks of a pose candidate are off, see Figure 4.4. In these situations, we create a ground-truth assignment, if at least 7 landmarks of major body joints are correctly located on a person. The idea behind this strategy is that VIP is rather robust to occasional landmark errors, since it incorporates information from multiple sensor modalities. Also, visual information is mainly used to estimate long-term parameters such as heading errors and sensor-to-bone-misalignments. Hence, sporadic landmark errors have only a minor impact on the final tracking result. Further, simply rejecting a

pose candidate if it contains only very few false landmarks would remove to much valuable information, especially in situation where people interact closely.

Given the ground-truth assignments, we evaluate the assignment accuracy of VIP in terms of precision and recall rates. Precision refers to the ratio of correctly assigned person labels with respect to all assignments made. Recall or sensitivity refers to the ratio of correct assignments with respect to all ground-truth assignments.

VIP achieves an assignment precision of 99.3% and a recall rate of 92.2% demonstrating the method correctly identifies the tracked persons for the vast majority of frames. This is a strong indication that VIP achieves a 3D pose accuracy on 3DPW comparable to the MPJPE of 26mm reported for TotalCapture.

4.5 Conclusion

By combining body-worn IMUs and a single moving camera, we introduced the first method that can robustly recover pose in challenging scenes. Previous approaches rely on a set of static cameras and incorporate information of a small set of IMUs merely to resolve visual ambiguities. In contrast, we use the visual information of a single hand-held camera to overcome general limitations of IMU-based motion capture. This has the advantage that the approach is portable and only requires a minimal number of exteroceptive sensors.

The proposed method, named VIP, is composed of three steps: initialization, assignment and sensor fusion. In the initialization step, initial 3D poses are obtained by fitting the body models to IMU data. The initial poses are still deteriorated by heading drift and inaccurate sensor-to-bone alignments. Also, relative distances between persons cannot be estimated using IMU data only. Hence, in a second step we associate 2D joint detections in the images to the initial 3D poses using a discrete graph labeling formulation. This formulation outputs a globally consistent assignment and does not require manual initialization. Finally, in the third step visual and inertial information are combined to estimate 3D pose, heading drift, camera pose, relative person distances and sensor-to-bone misalignments. Given the estimated error parameters and camera pose, association and sensor fusion are repeated once to further refine the results.

We evaluate VIP on TotalCapture, which is a dataset containing IMU data and ground-truth from a marker-based MoCap system. VIP achieves an average 3D joint position error of 26 mm and an average angular error of 12.3°, which is a clear improvement to the initial 3D poses having an error of 55 mm and 17.5°, respectively. We validate that modeling heading errors and optimizing sensor-to-bone misalignments are crucial and further investigate the influence of both aspects in isolation. We show that VIP can improve up to an average joint positions error of 15.1 mm if 2D pose annotations are perfect and demonstrate that camera pose is

accurately estimated. In addition, we evaluate the accuracy of VIP if only 6 IMUs are used. In this case, the average joint position error is $35.8\,mm$ which is only slightly worse compared to the full IMU setup.

Using VIP we created a new 3DPW dataset consisting of 1-2 person captures in challenging scenes performing various activities, such as shopping in a very crowded pedestrian zone, riding bus, doing sports, hugging, etc. 3DPW is publicly available for research purposes. In total, it contains 60 video sequences ($51,000$ frames or 1700 seconds of video captured with a phone at 30Hz), IMU data, 3D scans and 3D people models with 18 clothing variations, and the accurate 3D pose reconstruction results of VIP in all sequences. We anticipate that the dataset will stimulate novel research by providing a platform to quantitatively evaluate and compare methods for 3D human pose estimation.

We also used 3DPW to investigate the accuracy of the association step of VIP. Even for the challenging scenes in the dataset, VIP achieves an assignment precision of 99.3% and a recall rate of 92.2% validating that the method can correctly associate visual and inertial information for the vast majority of frames.

A major limitation of the proposed method is that it requires a sequence to be recorded before inference can take place. Hence, it is not applicable to situations that process poses in real-time. However, it would be possible to estimate heading errors and sensor-to-bone misalignments at the beginning of a recording and run an IMU-based approach online afterwards. In the experiments we show, that this produces very accurate results. Another limitation is that VIP does not exploit visual information to full extend. In the proposed method only the pixel coordinates of detected 2D poses are used. Visual appearance of people could be incorporated to simplify and improve the association of 2D to 3D poses. Also, geometric information about the real world geometry could be used to improve 3D poses and to estimate the camera pose more accurately.

Figure 4.13: Illustrations of several example frames of sequences in the 3DPW dataset. The dataset contains large variations in person identity, clothing and activities. For a couple of cases we also show animated, textured SMPL body models.

5 Conclusions

This thesis addresses the MoCap problem, which is to reconstruct the skeletal state of the human body from sensor measurements. Obtaining a numeric representation of the body pose has several applications in the fields of medical diagnosis, biomechanics, computer graphics, human-machine interactions, surveillance and learning approaches. Consequently, human motion capture has actively researched for decades and there exist various approaches to solve the MoCap problem.

Marker-based systems reconstruct the body pose by tracking the position of body-worn markers using several calibrated cameras. While this is accepted as the gold-standard in human motion capture, such systems are restricted to very controlled environments and wearing plenty markers on the body is intrusive. Marker-less methods extract and associate features in images that are used as virtual markers to estimate their pose. While this permits to wear regular clothing, it still requires a lot of static cameras to cope with depth-ambiguities and self-occlusions generated by the articulated structure of the human body. Motion capture with inertial sensors does not suffer from these limitations. The sensors are body-worn, hence no external equipment is required and the body pose is reconstructed in terms of the sensed sensor orientations. However, inertial sensors drift over time and wearing a lot of them is intrusive.

In this work two novel methods are presented, which work with a sparse sensory setup addressing several limitations of current MoCap approaches. In particular, a novel global optimization formulation is developed to incorporate measurements over a large temporal horizon. Together with anthropometric constraints imposed by a body model this enables to resolve ambiguities caused by the sparsity of input signals.

Sparse Inertial Poser

In the first part of this thesis this strategy is applied to reconstruct the full body pose from only 6 inertial sensors. We call this method Sparse Inertial Poser or SIP.

Using only 6 sensors instead of 10-17 introduces ambiguities. Previous approaches solve this by matching the incomplete sensor signals to the full set of measurements within a pre-recorded motion database. This is only works satisfactorily, if the query motions are part of the database. Unseen motions cannot be reconstructed.

Instead, in SIP the ambiguities are resolved with a generative model. We define a global energy term, that incorporates all available information of a complete recording sequence and maximize consistency between the model poses and sparse sensor measurements. A key observation is that the kinematic constraints imposed by the statistically learned skeletal model of SMPL reduce the search space in such a way, that acceleration data can be utilized to resolve ambiguities. Before, incorporating acceleration data in a sparse sensor setup was not possible due to the inherent drift caused by implicit double integration.

We evaluate SIP on two benchmark datasets, TNT15 and TotalCapture, as well as in challenging outdoor recordings. In the experiments we show, that the method can faithfully reconstruct the body pose from the sparse IMU inputs and that a statistically learned body model is superior to hand-crafted ones.

The reduced sensory effort of SIP can be crucial in applications where setup times have to be minimized or during long-term recordings, where it is simply more comfortable to wear fewer sensors. However, we also show that a certain level of motion is required to disambiguate the sparse sensor information. If acceleration of body parts are low accuracy degrades, since there is an uncertainty associated with static poses. Further, similar to all approaches relying only on inertial data, orientation data is affected by heading drift and global translation cannot be reconstructed. The latter limitation makes it impossible to capture interactions between people. This requires accurate knowledge of relative distances.

Video Inertial Poser

In the second part of this thesis we address the limitations of IMU-based motion capture by incorporating visual information of a single hand-held camera. We call this method the Video Inertial Poser or VIP. Visual and inertial information has already been combined in previous works. However, these methods address tracking problems, in which the body pose is estimated frame by frame. In addition, inertial information is mainly used to resolve visual orientation ambiguities in a static multi-camera setup. In contrast, with VIP we apply a global formulation and use visual information primarily to improve accuracy and remove restrictions of IMU-based motion capture. Further, using a single hand-held camera has the advantage that it preserves the portability of IMU-based motion capture.

In order to combine visual and inertial information, we apply a CNN to obtain 2D poses in form of pixel coordinates of major body joints detected in the camera

images. Since we make no assumptions about the number of persons visible in the scene, a single-frame approach is prone to tracking failures. The projection of the 3D worlds onto the 2D image sensor creates ambiguities and usually multiple 2D poses match to the poses of IMU-equipped persons. Also, in close interactions 2D poses are often erroneous, making it even more difficult to reliably incorporate visual information. In contrast, with VIP we apply a global matching strategy by formulating a graph labeling problem. The graph consists of all detected 2D poses of a recording sequence and we find a globally consistent matching to corresponding 3D poses obtained from the IMUs. After solving this discrete optimization problem, we apply a similar continuous global optimization method as for SIP. However, the visual information enables to estimate sensor heading drift, relative distances between people and to correct for IMU-to-bone misalignments originated from an inaccurate initial pose. Unfortunately, there exists no dataset which contains IMU and ground-truth data in outdoor scenarios to evaluate VIP in the target scenario. Hence we evaluate the method in two steps.

First, tracking accuracy of VIP is investigated on the TotalCapture dataset. By incorporating visual information from a single camera, VIP reduces the mean joint position error from $55\,mm$ to $26\,mm$ in comparison to the IMU only approach. In several experiments we investigate various aspects of VIP and demonstrate that estimating heading drift and correcting for IMU-to-bone misalignments is crucial.

Second, we evaluate assignment accuracy of VIP using the new 3D poses in the Wild dataset (3DPW). Since TotalCapture is captured in an indoor environment and does not contain multiple people recordings, we recorded 1-2 persons during everyday activities, such as shopping in a crowded pedestrian zone or during a bus ride. Even in very crowded scenes and close interactions, the graph labeling formulation of VIP achieves an assignment precision of 99.3% and a recall rate of 92.2%. We also use 3DPW to demonstrate the performance of VIP quantitatively. In summary, VIP enables practicable human motion capture of multiple people in natural environments. To the best of our knowledge, it is the first method that fully combines the advantages of camera-based and IMU-based motion capture: It is accurate, portable, reconstructs the poses of multiple people, can cope with temporal occlusions and works in natural environments.

Future Work

In this work we present methods that facilitate human motion capture with sparse sensor configurations, which are more practicable than existing approaches. However, there are still limitations which might be addressed in future work.

A major contribution of this work is the formulation of a global optimization scheme which incorporates all measurements of a recording sequence. Consequently, this requires to wait until all measurements are available. Such a method is not applicable

to situations that require real-time capabilities. An obvious solution to this is to apply sliding window techniques. Actually, there are already follow-up works to SIP applying this strategy. Huang *et al.* [96] train a long short-term memory network (LSTM) to reconstruct the full-body pose with 6 IMUs close to real-time and with competitive accuracy. Since the method relies on deep learning, they called the tracker Deep Inertial Poser.

The generative methods developed in this work require person-specific body models, which have been created using laser scanners. This is expensive and impracticable. In Chapter 3.4.2 we show that SIP also works with body models obtained from person height, weight and word ratings. Another solution has been presented by Alldieck *et al.* [97], who create accurate SMPL body models with a single hand-held camera. Interestingly, this would ideally suit to the VIP setup.

In the proposed VIP method, visual cues are transformed into 2D body poses using a CNN. This completely disregards geometric information about the background. Structure-from-motion approaches might be incorporated to further stabilize camera pose or to explicitly model interactions between the environment and persons. Another point related to geometric reasoning which is not considered in this work are self-intersections or intersections between body models. Modeling interpenetration cost terms into the objective function could further improve accuracy and realism of reconstructed poses.

Bibliography

[1] Bodo Rosenhahn, Reinhard Klette, and Dimitris Metaxas. *Human Motion*. Springer, 2008.

[2] Animals and hunter on the stone wall of the cave. Digital image, accessed 29 July 2019, <https://www.shutterstock.com/de/image-photo/figure-animals-hunter-on-stone-wall-527231944>.

[3] Vitruvian man. Digital image, accessed 29 July 2019, <https://www.wga.hu/art/l/leonardo/10anatom/1vitruviu.jpg>.

[4] Muybridge photo sequence of a running man. Digital image, accessed 29 July 2019, <https://photos.com/featured/1-muybridge-photo-sequence-of-a-running-man-eadweard-muybridge-collection-kingston-museumscience-photo-library.html>.

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

[6] Nikolaos Sarafianos, Bogdan Boteanu, Bogdan Ionescu, and Ioannis A Kakadiaris. 3D human pose estimation: A review of the literature and analysis of covariates. *Computer Vision and Image Understanding*, 2016.

[7] Thomas B Moeslund, Adrian Hilton, and Volker Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer vision and image understanding*, 104(2-3):90–126, 2006.

[8] Irvin Hussein López-Nava and Angelica Munoz-Melendez. Wearable inertial sensors for human motion analysis: A review. *Sensors Journal (IEEE)*, 16(22):7821–7834, 2016.

[9] Vicon. Accessed 16 August 2019, <http://www.vicon.com>.

[10] Qualisys. Accessed 16 August 2019, <https://www.qualisys.com/>.

[11] Gerard Pons-Moll and Bodo Rosenhahn. *Model-Based Pose Estimation*, pages 139–170. Springer, 2011.

[12] The captury. Accessed 29 August 2019, <https://thecaptury.com/>.

[13] Simi Reality Motion Systems GmbH. Accessed 16 August 2019, <http://www.simi.com>.

[14] 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 *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1297–1304. IEEE, 2011.

[15] J Taylor, J. Shotton, T. Sharp, and A. Fitzgibbon. The vitruvian manifold: Inferring dense correspondences for one-shot human pose estimation. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 103–110. IEEE, 2012.

[16] G. Pons-Moll, J. Taylor, J. Shotton, A. Hertzmann, and A. Fitzgibbon. Metric regression forests for human pose estimation. In *Proc. of British Machine Vision Conference (BMVC)*. BMVA Press 2013, 2013.

[17] Lulu Chen, Hong Wei, and James Ferryman. A survey of human motion analysis using depth imagery. *Pattern Recogn. Lett. (Elsevier Science Inc.*, 34(15):1995–2006, November 2013.

[18] Julieta Martinez, Rayat Hossain, Javier Romero, and James J. Little. A simple yet effective baseline for 3d human pose estimation. In *Proc. of International Conference on Computer Vision (ICCV)*. IEEE, 2017.

[19] Helge Rhodin, Jörg Spörri, Isinsu Katircioglu, Victor Constantin, Frédéric Meyer, Erich Müller, Mathieu Salzmann, and Pascal Fua. Learning monocular 3d human pose estimation from multi-view images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2018.

[20] Cristian Sminchisescu and Bill Triggs. Kinematic jump processes for monocular 3d human tracking. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2003.

[21] Ehsan Jahangiri and Alan L. Yuille. Generating multiple diverse hypotheses for human 3d pose consistent with 2d joint detections. In *Proc. of International Conference on Computer Vision Workshops (ICCVW)*. IEEE, 2017.

[22] Gerard Pons-Moll, David J. Fleet, and Bodo Rosenhahn. Posebits for monocular human pose estimation. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2345–2352, Columbus, Ohio, USA, 2014. IEEE.

[23] Xiaowei Zhou, Spyridon Leonardos, Xiaoyan Hu, and Kostas Daniilidis. 3D shape estimation from 2D landmarks: A convex relaxation approach. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4447–4455. IEEE, 2015.

[24] Edgar Simo-Serra, Ariadna Quattoni, Carme Torras, and Francesc Moreno-Noguer. A joint model for 2d and 3d pose estimation from a single image. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3634–3641. IEEE, 2013.

[25] Feng Zhou and Fernando De la Torre. Spatio-temporal matching for human detection in video. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 62–77. IEEE, 2014.

[26] Chunyu Wang, Yizhou Wang, Zhouchen Lin, Alan L Yuille, and Wen Gao. Robust estimation of 3d human poses from a single image. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2361–2368. IEEE, 2014.

[27] Edgar Simo-Serra, Arnau Ramisa, Guillem Alenyà, Carme Torras, and Francesc Moreno-Noguer. Single image 3d human pose estimation from noisy observations. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2673–2680. IEEE, 2012.

[28] Sijin Li, Weichen Zhang, and Antoni B Chan. Maximum-margin structured learning with deep networks for 3d human pose estimation. In *Proc. of International Conference on Computer Vision (ICCV)*, pages 2848–2856. IEEE, 2015.

[29] Bastian Wandt, Hanno Ackermann, and Bodo Rosenhahn. 3d reconstruction of human motion from monocular image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 38(8):1505–1516, 2016.

[30] Petrissa Zell, Bastian Wandt, and Bodo Rosenhahn. Joint 3d human motion capture and physical analysis from monocular videos. In *Proc. of the Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2017.

[31] Daniel Roetenberg, Henk Luinge, and Per Slycke. Moven: Full 6dof human motion tracking using miniature inertial sensors. *Xsens Technologies, December*, 2007.

[32] Daniel Vlasic, Rolf Adelsberger, Giovanni Vannucci, John Barnwell, Markus Gross, Wojciech Matusik, and Jovan Popović. Practical motion capture in everyday surroundings. *ACM Transactions on Graphics (TOG)*, 26(3):35, 2007.

[33] XSens. Accessed 12 October 2016, <https://www.xsens.com/products/>.

[34] Shimmer. Accessed 16 August 2019, <http://www.shimmersensing.com>.

[35] Notch Interfaces Inc. Accessed 16 August 2019, <https://wearnotch.com/>.

[36] R. Slyper and J. Hodgins. Action capture with accelerometers. In *Proceedings of the 2008 Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, 2008.

[37] Jochen Tautges, Arno Zinke, Björn Krüger, Jan Baumann, Andreas Weber, Thomas Helten, Meinard Müller, Hans-Peter Seidel, and Bernd Eberhardt. Motion reconstruction using sparse accelerometer data. *ACM Transactions on Graphics (TOG)*, 30:18, 2011.

[38] Huajun Liu, Xiaolin Wei, Jinxiang Chai, Inwoo Ha, and Taehyun Rhee. Realtime human motion control with a small number of inertial sensors. In *Symposium on Interactive 3D Graphics and Games*, pages 133–140. ACM, 2011.

[39] L. Schwarz, D. Mateus, and N. Navab. Discriminative human full-body pose estimation from wearable inertial sensor data. *Modelling the Physiological Human (Springer)*, pages 159–172, 2009.

[40] Gerard Pons-Moll, Andreas Baak, Thomas Helten, Meinard Müller, Hans-Peter Seidel, and Bodo Rosenhahn. Multisensor-fusion for 3d full-body human motion capture. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 663–670. IEEE, 2010.

[41] Gerard Pons-Moll, Andreas Baak, Juergen Gall, Laura Leal-Taixé, Meinard Muller, Hans-Peter Seidel, and Bodo Rosenhahn. Outdoor human motion capture using inverse kinematics and von mises-fisher sampling. In *Proceedings of the 2011 International Conference on Computer Vision (ICCV)*, pages 1243–1250. IEEE, 2011.

[42] Matthew Trumble, Andrew Gilbert, Charles Malleson, Adrian Hilton, and John Collomosse. Total capture: 3d human pose estimation fusing video and inertial sensors. In *Proceedings of 28th British Machine Vision Conference*, pages 1–13, 2017.

[43] Charles Malleson, Marco Volino, Andrew Gilbert, Matthew Trumble, John Collomosse, and Adrian Hilton. Real-time full-body motion capture from video and imus. In *2017 Fifth International Conference on 3D Vision (3DV)*, 2017.

[44] Sheldon Andrews, Ivan Huerta, Taku Komura, Leonid Sigal, and Kenny Mitchell. Real-time physics-based motion capture with sparse sensors. In *Proceedings of the 13th European Conference on Visual Media Production (CVMP 2016)*, page 5. ACM, 2016.

[45] Zerong Zheng, Tao Yu, Hao Li, Kaiwen Guo, Quionghai Dai, Lu Fang, and Yebin Liu. Hybridfusion: Real-time performance capture using a single depth sensor and sparse imus. In *Proc. of European Conference on Computer Vision (ECCV)*. IEEE, 2018.

[46] Thomas Helten, Andreas Baak, Gaurav Bharaj, Meinard Muller, Hans-Peter Seidel, and Christian Theobalt. Personalization and evaluation of a real-time depth-based full body tracker. In *2013 International Conference on 3D Vision (3DV)*, 2013.

[47] Gypsy 7 Motion Capture System. Digital image, accessed 15 August 2019, <https://metamotion.com/gypsy/gypsy-motion-capture-system.htm>.

[48] Electromagnetic Tracking 6DOF: Reliable Data, Repeatable Results. Digital image, accessed 15 August 2019, <https://polhemus.com/applications/electromagnetics/>.

[49] Takafumi Taketomi, Hideaki Uchiyama, and Sei Ikeda. Visual slam algorithms: A survey from 2010 to 2016. *IPSJ Transactions on Computer Vision and Applications*, 9(1):16, 2017.

[50] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment—a modern synthesis. In *International workshop on vision algorithms*, pages 298–372. Springer, 1999.

[51] Timo von Marcard, Gerard Pons-Moll, and Bodo Rosenhahn. Human pose estimation from video and IMUs. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 38(8):1533–1547, 2016.

[52] Timo von Marcard, Bodo Rosenhahn, Michael Black, and Gerard Pons-Moll. Sparse inertial poser: Automatic 3d human pose estimation from sparse imus. *Computer Graphics Forum 36(2), Proceedings of the 38th Annual Conference of the European Association for Computer Graphics (Eurographics)*, pages 349–360, 2017.

[53] Timo von Marcard, Roberto Henschel, Michael J. Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *Proc. of European Conference on Computer Vision (ECCV)*. IEEE, 2018.

[54] Roberto Henschel, Timo von Marcard, and Bodo Rosenhahn. Simultaneous identification and tracking of multiple people using video and imus. In *Proc. of Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2019.

[55] Multimodal motion capture dataset tnt15 - documentation. Accessed 16 August 2019, <http://www.tnt.uni-hannover.de/project/TNT15/TNT15_documentation.pdf>.

[56] 3d poses in the wild dataset. Accessed 16 August 2019, <http://virtualhumans.mpi-inf.mpg.de/3DPW/>.

[57] Richard M. Murray, Zexiang Li, and S. Shankar Sastry. *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[58] Saladin Dr. Kenneth. *Human Anatomy*. McGraw-Hill Education, 2013.

[59] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, October 2015.

[60] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, 2nd edition, 2006.

[61] Ethan Eade. Gauss-newton/levenberg-marquardt optimization. *Tech. Rep.*, 2013.

[62] Manon Kok, Jeroen D Hol, and Thomas B Schön. Using inertial sensors for position and orientation estimation. *Foundations and Trends in Signal Processing*, 11(1-2):1–153, 2017.

[63] XSens MTw development kit. Accessed 12 October 2016, <https://www.xsens.com/products/mtw-development-kit/>.

[64] Leonid Sigal, Alexandru O Balan, and Michael J Black. Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *International Journal of Computer Vision (IJCV)*, 87(1-2), 2010.

[65] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 36(7):1325–1339, 2014.

[66] CMU motion capture database. Accessed 12 October 2016, <http://mocap.cs.cmu.edu/>.

[67] Moritz Tenorth, Jan Bandouch, and Michael Beetz. The TUM kitchen data set of everyday manipulation activities for motion tracking and action recognition. In *Proc. of 12th International Conference on Computer Vision Workshops, ICCV Workshops*, pages 1089–1096. IEEE, 2009.

[68] Mpi08 database. Accessed 22 May 2019, <http://www.tnt.uni-hannover.de/project/MPI08_Database/>.

[69] Andreas Baak, Thomas Helten, Meinard Müller, Gerard Pons-Moll, Bodo Rosenhahn, and Hans-Peter Seidel. Analyzing and evaluating markerless motion tracking using inertial sensors. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 139–152. IEEE, 2010.

[70] Ilya Baran and Jovan Popović. Automatic rigging and animation of 3d characters. In *ACM Transactions on graphics (TOG)*, volume 26, page 72. ACM, 2007.

[71] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[72] Matthew M. Loper, Naureen Mahmood, and Michael J. Black. MoSh: Motion and shape capture from sparse markers. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 33(6):220:1–220:13, 2014.

[73] Gerard Pons-Moll. *Human Pose Estimation from Video and Inertial Sensors*. PhD thesis, Leibniz Universität Hannover, 2014.

[74] James G Richards. The measurement of human motion: A comparison of commercially available systems. *Human movement science (Elsevier)*, 18(5):589–602, 1999.

[75] Pierre Merriaux, Yohan Dupuis, Rémi Boutteau, Pascal Vasseur, and Xavier Savatier. A study of vicon system positioning performance. *Sensors (Multidisciplinary Digital Publishing Institute)*, 17(7), 2017.

[76] Mixamo. Accessed 12 October 2016, <http://www.mixamo.com/>.

[77] House of moves. Accessed 12 October 2016, <http://moves.com/>.

[78] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *Proc. of European Conference on Computer Vision (ECCV)*. IEEE, 2016.

[79] Howard E. Haber. Notes on the matrix exponential and logarithm. http://scipp.ucsc.edu/~haber/webpage/MatrixExpLog.pdf, 2019. (accessed 2019-08-28).

[80] Stephan Streuber, M. Alejandra Quiros-Ramirez, Matthew Q. Hill, Carina A. Hahn, Silvia Zuffi, Alice O'Toole, and Michael J. Black. Body Talk: Crowdshaping realistic 3D avatars with words. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 35(4), 2016.

[81] Daniel Vlasic, Ilya Baran, Wojciech Matusik, and Jovan Popović. Articulated mesh animation from multi-view silhouettes. In *ACM Transactions on Graphics (TOG)*, volume 27, page 97. ACM, 2008.

[82] Juergen Gall, Carsten Stoll, Edilson De Aguiar, Christian Theobalt, Bodo Rosenhahn, and Hans-Peter Seidel. Motion capture using joint skeleton tracking and surface estimation. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pages 1746–1753. IEEE, 2009.

[83] Helge Rhodin, Christian Richardt, Dan Casas, Eldar Insafutdinov, Mohammad Shafiei, Hans-Peter Seidel, Bernt Schiele, and Christian Theobalt. EgoCap: egocentric marker-less motion capture with two fisheye cameras. *ACM Transactions on Graphics (TOG)*, 35(6):162, 2016.

[84] Takaaki Shiratori, Hyun Soo Park, Leonid Sigal, Yaser Sheikh, and Jessica K Hodgins. Motion capture from body-mounted cameras. In *ACM Transactions on Graphics (TOG)*, volume 30, page 31. ACM, 2011.

[85] Ijaz Akhter and Michael J Black. Pose-conditioned joint angle limits for 3D human pose reconstruction. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1446–1455. IEEE, 2015.

[86] Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter Gehler, and Bernt Schiele. DeepCut: Joint subset partition and labeling for multi person pose estimation. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016.

[87] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

[88] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM Transactions on Graphics (TOG)*, 36(4):44, 2017.

[89] Roberto Henschel, Laura Leal-Taixé, Daniel Cremers, and Bodo Rosenhahn. Fusion of head and full-body detectors for multi-object tracking. In *Proc. of the Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, 2018.

[90] Evgeny Levinkov, Jonas Uhrig, Siyu Tang, Mohamed Omran, Eldar Insafutdinov, Alexander Kirillov, Carsten Rother, Thomas Brox, Bernt Schiele, and Bjoern Andres. Joint graph decomposition & node labeling: Problem, algorithms, applications. In *Proc. of the Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, volume 7. IEEE, 2017.

[91] Gurobi Optimization, Inc. Accessed 16 August 2019, <https://www.gurobi.com/>.

[92] Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, and Bernt Schiele. Subgraph decomposition for multi-target tracking. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5033–5041. IEEE, 2015.

[93] Adam D Bull. Convergence rates of efficient global optimization algorithms. *Journal of Machine Learning Research*, 12(Oct):2879–2904, 2011.

[94] Chao Zhang, Sergi Pujades, Michael Black, and Gerard Pons-Moll. Detailed, accurate, human shape estimation from clothed 3D scan sequences. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017.

[95] Gerard Pons-Moll, Sergi Pujades, Sonny Hu, and Michael Black. ClothCap: Seamless 4D clothing capture and retargeting. *ACM Transactions on Graphics, (Proc. SIGGRAPH)*, 36(4), 2017.

[96] Yinghao Huang, Manuel Kaufmann, Emre Aksan, Michael J Black, Otmar Hilliges, and Gerard Pons-Moll. Deep inertial poser: learning to reconstruct human pose from sparse inertial measurements in real time. In *SIGGRAPH Asia 2018 Technical Papers*. ACM, 2018.

[97] Thiemo Alldieck, Marcus Magnor, Bharat Lal Bhatnagar, Christian Theobalt, and Gerard Pons-Moll. Learning to reconstruct people in clothing from a single RGB camera. In *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1175–1186. IEEE, Jun 2019.

Curriculum Vitae Timo von Marcard

Date of birth: 31.03.1084
Place of birth: Giessen, Germany

Professional Experience

Nov 2019 -	Research Engineer, Simi Reality Motion Systems GmbH, Hannover, Germany
Nov 2013 - Oct 2019	Research associate, Institut für Informationsverarbeitung, Leibniz University Hannover, Germany
May 2018 - Oct 2019	Consulting and software development for Otto Bock Healthcare GmbH, Duderstadt, Germany
Sep 2010 - Oct 2013	Developer for embedded software and algorithms, Otto Bock Healthcare GmbH, Duderstadt, Germany

Education

Nov 2013 - Oct 2019	Phd candidate, Institut für Informationsverarbeitung, Leibniz University Hannover, Germany
Sep 2008 - Jun 2010	Masters degree, "Systems, Control and Mechatronics", Chalmers University, Sweden
Oct 2003 - Feb 2008	Dipl.-Ing. (FH), "Mechatronics", Hochschule für Wirtschaft und Technik Karlsruhe, Germany

Awards

Apr 2017	Best Paper Award at Eurographics 2017, Lyon, France
Oct 2008	VDI Award for for outstanding academic achievements, Karlsruhe, Germany

Die Reihen der Fortschritt-Berichte VDI:

1 Konstruktionstechnik/Maschinenelemente
2 Fertigungstechnik
3 Verfahrenstechnik
4 Bauingenieurwesen
5 Grund- und Werkstoffe/Kunststoffe
6 Energietechnik
7 Strömungstechnik
8 Mess-, Steuerungs- und Regelungstechnik
9 Elektronik/Mikro- und Nanotechnik
10 Informatik/Kommunikation
11 Schwingungstechnik
12 Verkehrstechnik/Fahrzeugtechnik
13 Fördertechnik/Logistik
14 Landtechnik/Lebensmitteltechnik
15 Umwelttechnik
16 Technik und Wirtschaft
17 Biotechnik/Medizintechnik
18 Mechanik/Bruchmechanik
19 Wärmetechnik/Kältetechnik
20 Rechnerunterstützte Verfahren (CAD, CAM, CAE CAQ, CIM ...)
21 Elektrotechnik
22 Mensch-Maschine-Systeme
23 Technische Gebäudeausrüstung

ISBN 978-3-18-3**86610**-6